SHIPBUILDING
ICCAS
Cambridge USA • 1999

10TH INTERNATIONAL CONFERENCE ON COMPUTER APPLICATIONS IN SHIPBUILDING

7–11 JUNE, 1999

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, USA

VOLUME 1

EDITED BY CHRYSSOSTOMIDIS AND JOHANSSON

10th International Conference on Computer Applications in Shipbuilding

Volume 1

About ICCAS

ICCAS is short for International Conference on Computer Applications in Shipbuilding. The ICCAS conferences are initiated by IFIP WG5.6. They are held every 2 to 3 years. Previous conferences were held in:

Tokyo 1973	Trieste 1985
Gothenburg 1976	Shanghai 1988
Glasgow 1979	Rio de Janiero 1991
Annapolis 1982	Bremen 1994

Yokahama 1997

The ICCAS '99 Conference

ICCAS '99 was organized by the Massachusetts Institute of Technology Sea Grant College Program and the Department of Ocean Engineering at MIT. The conference was held at MIT in Cambridge, Massachusetts, USA on June 7-11, 1999.

International Program Committee

Chairman:
Kaj Johansson, Kockums Computer Systems AB, Sweden

Members:
Torben Andersen, Odense Steel Shipyard, Denmark
Joachim Brodda, Balance Technology Consulting, Germany
Yoo-Sang Choo, National University of Singapore, Singapore
John Duncan, Ministry of Defense, UK
Soon-Hung Han, KAIST, Korea
Bill Hills, EDC / Newcastle University, UK
Ken Ito, Mitsubishi Heavy Industries, Japan
Shen Jin-fang, Marine Design and Research Institute, China
Takeo Koyama, University of Tokyo, Japan
Olivier Richard, Chantiers de l'Atlantique, France
Richard Lee Storch, University of Washington, USA
Svein Waagboe, Marintek, Norway

International Program Committee
Chairman:
Kaj Johansson, Kockums Computer Systems AB, Sweden
Members:
Torben Andersen, Odense Steel Shipyard, Denmark
Joachim Brodda, Balance Technology Consulting, Germany
Yoo-Sang Choo, National University of Singapore, Singapore
John Duncan, Ministry of Defense, UK
Soon-Hung Han, KAIST, Korea
Bill Hills, EDC / Newcastle University, UK
Ken Ito, Mitsubishi Heavy Industries, Japan
Shen Jin-fang, Marine Design and Research Institute, China
Takeo Koyama, University of Tokyo, Japan
Olivier Richard, Chantiers de l'Atlantique, France
Richard Lee Storch, University of Washington, USA
Svein Waagboe, Marintek, Norway

Local Organizing Committee
Chairman:
Chryssostomos Chryssostomidis, Massachusetts Institute of Technology
Members:
James Boudreaux, Electric Boat
P. Thomas Diamant, J.J. McMullen Associates
Tom Eagar, Massachusetts Institute of Technology
Sotiris Emmanuel, Massachusetts Heavy Industries
Neil Gallagher, Webb Institute
Otto Jons, Nichols Advanced Marine
Robert G. Keane, Naval Surface Warfare Center
Hauke Kite-Powell, Woods Hole Oceanographic Institution
Tom Lamb, University of Michigan
Don Liu, American Bureau of Shipping
Takashi Maekawa, Massachusetts Institute of Technology
Dennis Mahoney, Massachusetts Institute of Technology
Hank Marcus, Massachusetts Institute of Technology
Koichi Masubuchi, Massachusetts Institute of Technology
Bryan Miller, Tribon
Jerry Myers, Avondale Industries
Mike Parsons, University of Michigan
Nick Patrikalakis, Massachusetts Institute of Technology
Mike Powell, Newport News Shipbuilding
Nick Sapidis, Massachusetts Institute of Technology
Robert Schaffran, Consultant
Bill Slowik, ONR
Richard Lee Storch, University of Washington
Matt Tedesco, NASSCO
Richard Thorpe, Kvaerner Masa Marine Inc.
William Vorus, University of New Orleans
Stan Weliever, Bath Iron Works
David Whitfield, Mississippi State University
Ronald Wood, Ingalls Shipbuilding

Local Organizing Committee

Chairman:
Chryssostomos Chryssostomidis, Massachusetts Institute of Technology

Members:
James Boudreaux, Electric Boat
P. Thomas Diamant, J.J. McMullen Associates
Tom Eagar, Massachusetts Institute of Technology
Sotiris Emmanuel, Massachusetts Heavy Industries
Neil Gallagher, Webb Institute
Otto Jons, Nichols Advanced Marine
Robert G. Keane, Naval Surface Warfare Center
Hauke Kite-Powell, Newport News Shipbuilding
Tom Lamb, University of Michigan
Don Liu, American Bureau of Shipping
Takashi Maekawa, Massachusetts Institute of Technology
Dennis Mahoney, Massachusetts Institute of Technology
Hank Marcus, Massachusetts Institute of Technology
Koichi Masubuchi, Massachusetts Institute of Technology
Bryan Miller, Tribon
Jerry Myers, Avondale Industries
Mike Parsons, University of Michigan
Nick Patrikalakis, Massachusetts Institute of Technology
Nick Sapidis, Massachusetts Institute of Technology
Robert Schaffran, Consultant
Richard Lee Storch, University of Washington
Matt Tedesco, NASSCO
Richard Thorpe, Kvaerner Masa Marine Inc.
William Vorus, University of New Orleans
Stan Weliever, Bath Iron Works
David Whitfield, Mississippi State University
Ronald Wood, Ingalls Shipbuilding

Sponsors
The conference was sponsored by the MIT Sea Grant College Program and the Department of Ocean Engineering at MIT. Additional support was provided by the Office of Naval Research.

Editors
C. Chryssostomidis, Massachusetts Institute of Technology
K. Johansson, Kockums Computer Systems AB

Published by:
Massachusetts Institute of Technology
MIT Sea Grant College Program
292 Main Street, MIT Room E38-300
Cambridge, MA 02139, USA

MIT Sea Grant Publication 99-1

ISBN 1-56172-024-0

Acknowledgement

Office of Naval Research

through mutual cooperation, has helped organize the conference by providing necessary funds. We are deeply grateful to them.

10th International Conference on Computer Applications in Shipbuilding

ICCAS '99

June 7-11, 1999 at the Massachusetts Institute of Technology in Cambridge, Massachusetts, USA

Volume 1

Edited by C. Chryssostomidis
Massachusetts Institute of Technology

and

K. Johansson
Kockums Computer Systems AB

PREFACE

Advances in information technology have revolutionized aspects of shipbuilding, from preliminary design to assembly and shipyard management. This technology will continue to be an important factor in future productivity and performance. ICCAS '99 at the Massachusetts Institute of Technology brought together a broad cross section of the international academic and industrial community to address these issues. The papers presented ranged the full spectrum, from reviews of operational experience with existing computer applications to discussions of emerging advances in information technologies destined to become the basis for the next generation of shipyard computer systems.

Papers were grouped into the following areas:

CAD/CAM/CIM Systems
Operation and Management
Product Modelling
Emerging Information Technologies
Information Technology Infrastructure
Detailed and Production Design
Preliminary Design
Assembly and Construction

In all, 86 papers were presented at ICCAS '99, and they are all published in these three volumes.

We deeply appreciate the efforts of the committee members, contributors and participants, all of whom have helped this conference to be a success. We also thank the Office of Naval Research for their generous support.

C. Chryssostomidis
K. Johansson

June 1999

Contents

CONTENTS

Invited Paper: For See: MIT-R-99-001

Design and Manufacturing in a Distributed Computer Environment
Nicholas M. Patrikalakis, Chryssostomos Chryssostomidis and Konstantinos P. Mihanetzis
published in a separate volume

Contents of Volume 1

CAD/CAM/CIM Systems

Operation and Management

Operation and Management

Product Modelling

Emerging Information Technologies

Contents of Volume 2

Preliminary Design

Assembly and Construction

CAD/CAM/CIM Systems

COMPASS SMART PRODUCT MODELING ENVIRONMENT FOR SHIPBUILDING

Dominique Payannet, Intergraph Corporation, Huntsville, Alabama, U.S.

Introduction

The Problem

The DARPA/MARITECH Commercial Object Model of Products/Processes for an Advanced Shipbuilding System (COMPASS) Program completed research and performed work in developing a Smart Product Modeling (SPM) Environment for shipbuilding. Some of the problems faced in this program included:

- **Scalability** -- Applications need to be same, whether used by one or one hundred designers.
- **Reusable design components** -- From a structural bracket to a complete bulkhead, the user needs the ability to "cut and paste" elements of the design, similar to a Word document.
- **Managing change over the design process** -- Ships can have up to 50 million components that need to be tracked throughout the entire life cycle.
- **Multiple orthogonal views from a single logical object model** -- A component can belong to a system, design, zone, construction unit, and a sub-assembly at the same time. The user must be able to "see" that component in any view, change the view on the fly, and redefine what views the component belongs to as the ship design matures.
- **Open information model** -- Shipyards have multiple systems that must interact with ship design data on a daily basis as the design evolves. It is no longer acceptable to get data "downloads" or translations to business, planning, and production systems. There must be an open, extensible repository of ship information.
- **Concurrent engineering** – Since shipbuilding is low-volume manufacturing of highly complex products, the industry needs a concurrent, multi-user engineering design and production environment where engineering and manufacturing activities happen simultaneously.

The Solution

The COMPASS SPM Environment provides support to build applications without worrying about the underlying technology. This platform provides a three-tier Windows NT Microsoft Distributed interNet Architecture (DNA)-based architecture. Each task is developed independently from other tasks, which takes advantage of:

- Business rules supported by binary relations
- Relations described in UML and editable by UML visual modeler
- A transaction context allowing short and long transactions
- Concurrency and locking
- A versioning capability
- Independence from the data store
- A set of CAD objects (Active Xs, controls, scriptlets, beans)
- All commands are just COM objects that can be written in any language and plugged in at run time

This paper will review the research and show how the problems were solved in developing this COMPASS platform for the shipbuilding industry.

Current CAD Systems Limitations

Generally, traditional CAD systems are not designed to handle large projects such as those found in the shipbuilding industry. This evolution stems from mechanical-based file systems. Those systems that were applied to shipbuilding projects were hampered by lack of specific shipbuilding functionality, limited in regard to external database structures and management, and lacked the necessary technology that allowed concurrency to be applied to the process. The ones that did meet the functional requirements were monolithic entities that made usage cumbersome.

Data Problems

Mechanical applications are "model oriented." Geometry is the critical data and other data associated with the project (production, maintenance, price, and history) are added when required. The workflow is driven by the modeling capability of the CAD system, while data management is an externalized workflow. Consequently, the data management product serving production and maintenance duplicates some modeling data in order to provide query and versioning capabilities. Traditional CAD files are used as objects referenced by the data management system.

Initially, CAD systems did not provide complex relationships; the data models were static geometry. Model changes invariably meant a remodeling of the complete part. Many mechanical systems eventually added parametric relationships that allowed individual parts to be modified through 'simple' parameter edits. Nevertheless, since these systems are file based, the boundaries of the files imposed severe limitations within the relationship tree. Although it is possible to manage relationships within a single file for a specific part or a set of parts, the management of relationships across thousands of files is nothing short of a pipe dream.

Mechanical systems were first designed to create and edit parts. It is only during the past few years that CAD vendors have became interested in assemblies. Progress in this field was again hampered by file boundaries limiting the size of the assemblies. A ship requiring millions of parts is not suited to file-based assembly processes.

Traditional CAD systems are evolutionary. This evolution turned these systems into monolithic applications because of the demand for more specific requirements and "how-to" interjections. In order to provide a complete solution amid an evolutionary system, the re-programming of existing standard components (printing, text, GUI) had to be considered within each framework. The result of these changes produced complex applications that were difficult to learn, use, and customize.

Through the evolution of geometric data, vendors required their own data format in order to enhance their tools. Too much freedom in the evolutionary process of a CAD system can generate integration problems at the data extraction level when dealing with external processes such as data management and reporting. To alleviate these problems, vendors and customers agreed to formalize initiatives for data standards like STEP[1]. Although this may resolve the fundamental data exchange problems, the complete schema remains very much hidden in proprietary data formats of individual CAD systems. These systems, having been handicapped over the years by proprietary formats, can only expose public domain schemes through translation. This equates to loss of information and duplication of data.

[1] STEP "Standard for the Exchange of Product Model Data," http://www.nist.gov/sc4/www/stepdocs.htm.

Concurrency Problem

Working in a concurrent environment is restricted with file-based applications, since this allows only long transaction processes into the database. (i.e., only one user working on one file). The ensuing dilemma involves two problems. First, if the CAD system is able to handle assemblies, then only a handful of users will access and manipulate the "higher level assemblies." Second, if the files have "part" granularity, the relationship across parts is very limited and re-computation of other files becomes necessary, thus returning to the old problem. With file-based systems, access control cannot be managed at smaller granularities than a single file, again inducing the problem of relationships across parts and concurrent access.

Evolution

CAD systems have evolved for many years and have been ported to different systems (mainframes, Unix, NT). Backward compatibility was achieved only by providing a proprietary framework that shielded all the compatibility problems. Guaranties for upward compatibility meant that the learning curve for developers, as well as for end users, became prohibitive. The development of applications on these proprietary systems necessitated costly training and high consultancy fees.

Scalability also became an issue. Customers requiring a single workstation for a small specific project would be offered a lower-level CAD system that would be cost effective, but not necessarily technically productive. At the same time, the vendor would offer that customer who required a much larger number of workstations a higher performance system at a higher premium. These disparate systems could possibly work by exchanging data via translation and data duplication. However, the scalability of a system is a key factor in its successful acceptance in the market. More importantly, it addresses directly the concerns that the industry has for many years had to contend with.

COMPASS Architecture

Notwithstanding the problems described above, the DARPA/MARITECH COMPASS research program was launched to develop an SPM environment for shipbuilding. The thrust of this research project was to ultimately develop a set of tools that encompasses past lessons learned, incorporates the latest technologies, and is built on a Windows® framework and tool set. The bottom line is to create major productivity inducements within the shipbuilding engineering design.

The first issue was to remove proprietary frameworks in order to limit the learning curve of application developers. Scalability and component re-use prompted the move towards the standard DNA[2] from Microsoft.

DNA Architecture

The DNA architecture is based on scalability and binary object re-use. The design of applications using DNA provides seamless scalability from a single workstation to a fully distributed network of computers by making use of all the binary components available on the market. All applications are split into three logical tiers (See Figure 1):

- Server tier that handles the data access.

[2] Windows DNA "Building Windows Applications for the Internet Age," Stephen Rauch, Platform Strategy and Architecture, PDC, September '98, Denver, USA.

- Middle tier that handles all the business objects and rules, and ensures data consistency before committing any transaction to the server tier.
- Client tier that contains the Graphics User Interface (GUI) and all the commands accessing the middle tier.

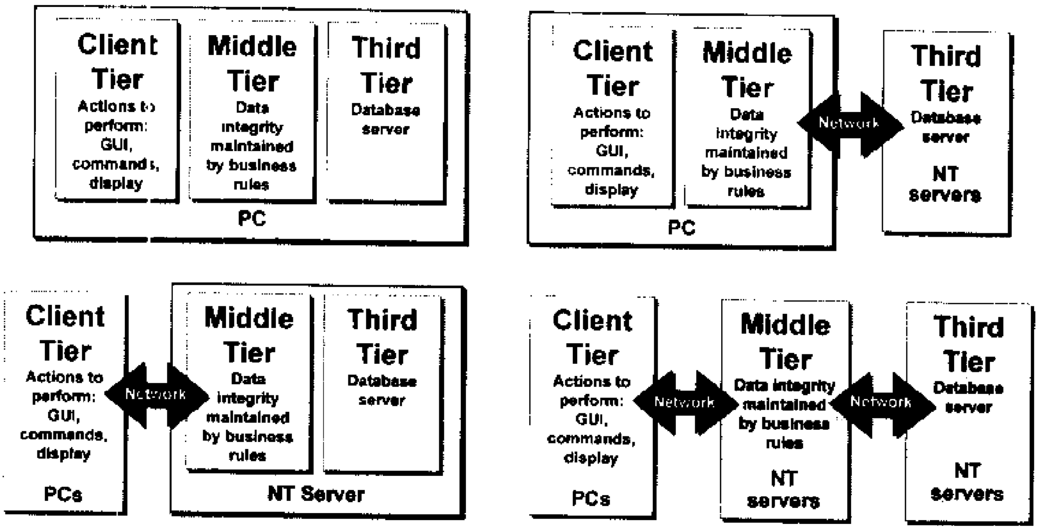


Figure 1. Three-tier Architecture

Each of these tiers can be distributed over several machines depending on the configuration. It can run on a single machine as well as with X clients driving M middle tier machines, accessing P servers. Microsoft tools provide load-balancing software as well as object caching and resource pooling for the middle tier machines. In this way, the development framework becomes the standard DNA model and the development tool is Microsoft Visual Studio. Within this framework, the COMPASS SPM toolkit becomes just a toolbox containing a set of binary re-usable components (as well as external components) that can be plugged in to build applications that handle large data sets.

Moreover, as the DNA model is Component Object Model (COM[3])-based, the components offer another degree of scalability. The interface granularity provides a way to downsize applications. For example, a single application not requiring any access control does not contain components that do not call on the access control interfaces in the middle tier, hence skipping the relevant algorithms One other major asset of this technology is the variety of client types ('rich' or 'thin') that can be plugged in without impacting any existing application.

Component Model

In order to avoid the 'monolithic' application problems (where all the environments have to be released at the same time by the same provider, deciding the eventual schema evolution), the COMPASS Program selected the more successful component model. With this schema, complex shipbuilding applications are made of numerous environments such as Catalogs, Structural, Outfitting, Planning, Preliminary Design, Drawing, or Analysis, and can be acquired individually and

[3] COM "Common Object Model," http://www.microsoft.com/com.

released separately. For the first time, customers can easily create and integrate their own applications into this single environment.

In this manner, a shipbuilding application is made up of independent environments that can be plugged in or removed at any time. Each environment can also be completely independent from the others and have its own GUI, schema, and data store. Each independent environment can reference other environments in a Direct Acyclic Graph (DAG) fashion. (See Figure 2.) Hence the COMPASS platform handles transactions across several data stores.

In order to provide independent design for each environment, there are no imposed constraints. This allows each environment to have: 1) a specific data store that corresponds to its requirements, and 2) its own schema and evolution. This enables users to write their own integrated environments without impacting the existing data model.

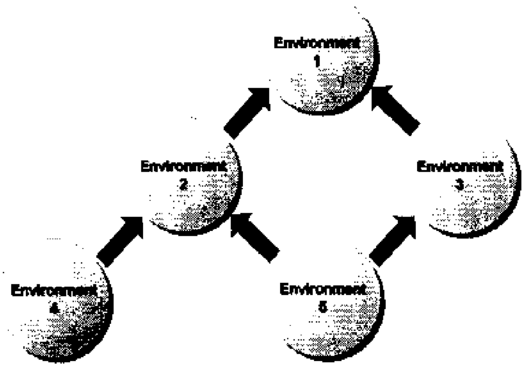


Figure 2. DAG Dependencies Between Environments

Server Tier

As stated earlier, the server tier is responsible for data access. COMPASS adds an insulation layer between the business objects and the persistence schema by traversing the layer. In this way, the data store can be anything as long as there is an available driver for it. The following data stores can be used in the COMPASS platform:

- Object databases: ODI, Versant, Objectivity.
- All OLEDB providers: SQL7.0, Oracle, flat files.

Each data store provides functionality for persistence, query, transaction support, and locking mechanism. Since it is a COM model, some data stores might not support all of this functionality. However, the system will downgrade seamlessly. For example, a data store that does not support query will prevent this functionality from being exposed.

Middle Tier

The middle tier is the heart of the environment as it exposes the object model (logical schema) to all the possible clients. (See Figure 3.) The middle tier's role is to:

- Provide a transacted model for editing data.
- Insure data consistency of business objects particularly relevant with models containing large numbers of business rules.
- Be responsible for keeping object or server connections cache.

The data stores running on the server tier are connected as standard resources for MSDTC (Microsoft Distributed Transaction Coordinator).

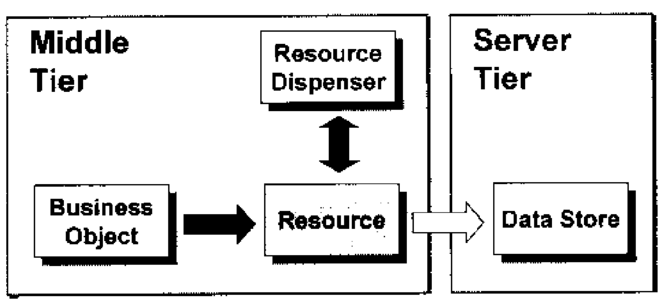


Figure 3. Middle Tier Overview

They all expose a set of interfaces insulating the data access from the implementation of the data stores. Each business object in the middle tier only accesses this set of interfaces, keeping the business object's logic completely independent from the data stores.

Data Consistency

As the environments use separate data stores, the middle tier has an active short transaction in which each data store part of the data edition is automatically enrolled. This is performed by the standard MSDTC, which implicitly enrolls all the resources (data stores) in the current transaction. When there are multiple independent data stores involved in the application, there is automatically a function of multiple schemas. The "active" schema is the union of all the schemas exposed for a specific operation. Since the environment can only have DAG dependencies, the corresponding schema will also have DAG included.

Data Models

There are two types of schemas for applications: 1) the physical model corresponding to the physical implementation for the data store, and 2) the logical model corresponding to the COM model. Middle and client tier objects access the latter since the business rules are only enforced by the business rules stored in the middle tier. Nevertheless, for read-only operations (i.e. reporting) clients might be able to access the data store model directly. The logical model is a public UML[4] schema entered with a visual modeler and stored in Microsoft Repository[5]. The business logic refers to the Repository's public cache to create and maintain the business logic.

Business Rules

Binary relationships[6] enforce all business rules across objects. These can handle direct dependencies or non-linear constraints. Each relationship can have semantics for every complex operation such as evaluation or copying. Every time a business object is edited and an update is requested, the graph of dependencies described by the relationship is captured and ordered. Then the corresponding semantics (e.g., evaluation semantics) are triggered and modify the dependencies. If all the semantics succeed during the update process, then the current transaction can be committed. Otherwise the current transaction is aborted.

The relationship and its semantics are described in the UML model. Changing a semantic can be done by just editing the schema -- instead of requiring a new release of software, as is common for traditional CAD systems. The mechanism is also impacted by the access control delaying some tasks across access control boundaries.

Transaction

The CCMPASS platform supports two types of transactions: short and long. Long transactions are simple to handle since data is insulated from concurrency before editing is executed. Only short transactions make sense for concurrent environments, since the data is locked for very small periods (milliseconds). Short transactions are driven by commands, but their duration is not dependent on the GUI. An end user can take several minutes to perform a complex command while the data might be locked only for a few milliseconds. Combining optimistic locking during the GUI operations with pessimistic locking (during the last data consistency check) provides this capability.

---

[4] UML "Unified Modeling Language Resource Center," http://www.rational.com/uml.

[5] Bernstein, P.A., and U. Dayal, "An Overview of Repository Technology," International Conference on Very Large Data Bases, Morgan Kaufmann Publishers, San Francisco, 1994, pp. 705-713.

[6] Patent US5692184: Object relationship management system.

No objects remain between transactions; the only possible data left are object caches or connection pools.

Access Control

Workflow and security must be controlled on large projects. Each business object is subject to access control rules before finding its way into the middle tier. This is done by a two rule-based system that can be altered at run time. The first rule set assigns the objects according to logical groups; these rules can be as simple as the notion of an 'active group' to rules based on any information exposed in the UML schema (i.e. type, interface, attribute value). In the latter case, the rules are re-evaluated before committing the edited data.

The second set of rules grants users, groups, or roles certain operations such as Create, Read, Update, or Delete on certain logical groups. These pre-compiled, optimized rules are run every time a business object is instantiated in the middle tier (creating or loading business objects). The business logic takes into account the access control to delay or prevent the editing of data. This provides full flexibility to organize a dynamic data partition (static in the case of files).

Query

Business objects do not have geometric constraints. Geometry is treated as any other attribute for the data management such as provider name, creation date, and price. Query becomes the main selection mechanism to work with data (i.e., "locate" in traditional CAD systems). Query is based on the Active Data Object (ADO) model. It is performed by SQL statements against the logical schema exposed through the UML model and returns a list of business objects. (See Figure 5.)

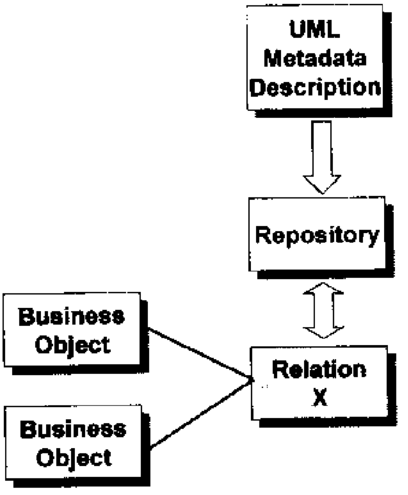


Figure 5. Open UML Schema

COMPASS Business Objects

Typically, application writers build their business objects, but the COMPASS toolbox provides several generic objects that can be re-used as is. These include:

- Relations (allow the creation of any business rule).
- References (to external 3rd party objects supporting OLE for Design & Modeling[7]).
- Symbols (allow sharing of parametric definitions of parts).
- Proxies (allow creation of relationships across data stores).
- Basic 3D geometry (usually aggregated by applications business objects).

Versioning

There are a number of reasons why a shipbuilding CAD system must efficiently handle versioning of objects. Two of the most notable include supporting what-if trade-off studies in the early stages of design, and allowing multiple ships to be built from a single class design without having to duplicate and separately maintain copies of the database for each ship.

Traditional CAD systems do not deal with multiple versions of many objects. They depend on an external PDM system to perform this function. This requires copying portions of the product model into working files used by the CAD system, and then copying the data back into the PDM

[7] OLE for Design and Modeling Applications, Microsoft, January 24, 1995, Redmond, USA.

database when the work was completed. This approach works for items such as automobiles, with a moderate number of objects, but does not scale well to the ship design problem where the database may house millions of objects and a user will reference tens of thousands of parts in a typical design session.

The COMPASS design has a very clean solution to this problem that takes advantage of the N-tier distributed object architecture. In this approach, the physical model on the disk stores multiple versions of objects, while the logical business object model is kept simple by ignoring versions. A thin layer of software "filters" through the physical versions and delivers the single appropriate version of each object to the in-memory business object tier.

Figure 6 shows how this approach provides each user with immediate access to the shared product model while presenting a logical view of a particular configuration.

The COMPASS design also supports many relationships between objects in the logical and physical models. These relationships may also have multiple versions in the physical model. The software layer that filters the objects also filters the relationships. This insures that the appropriate version of each object presented to the in-memory business objects and their interrelationships are consistent with the view of the physical model selected by the user.

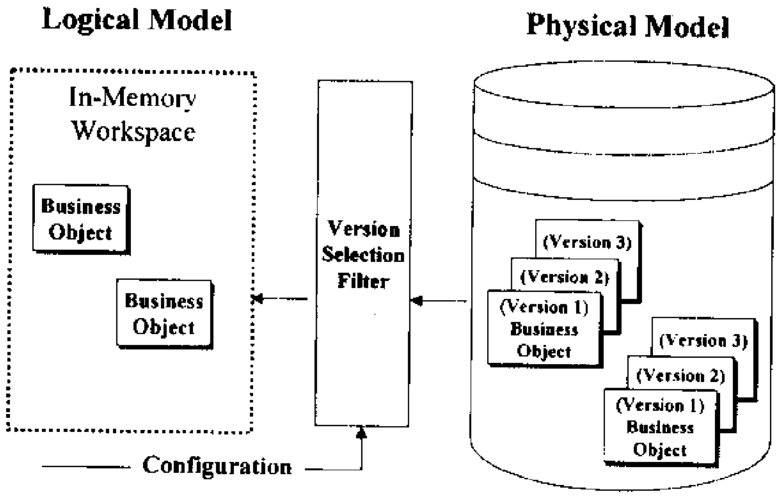


Figure 6. Logical Model is Consistent with the Physical Model

Client Tier

The client tier contains all the GUI and commands specific to applications and environments. It accesses the business objects. For read-only operations, the connections can be accessed through the standard ADO model. Applications can support many types of different clients, from thin clients (that might be Web-centric reporting tools) to rich clients providing real time 3D-model manipulation. Thin clients can be written with market-available tools such as DHTML, controls, ActiveX™, applets, scripts, and through the eventual re-use of the components that form part of the COMPASS toolbox. Although the market provides many standard components (tree controls, GRIDs), there are still few re-usable binary objects for building rich 3D clients (3D views, command control).

The COMPASS toolbox provides the most useful components for a 3D CAD system: view sets, 3D views, locators, highlighters, menus, working sets, transaction manager, command manager, units of measure, session manager, trader, view printers, element lists, select sets, project management, status bar, style manager, and MDI control. From these components, only the working set and the transaction manager access the middle tier.

With the traditional framework, there is a rigid model made of hardwired subsystems like views and documents. Developers override as much as they can of the standard presentation model in order to customize their specific applications. This requires a thorough understanding of the full model as well as all the hard-coded behaviors.

The COMPASS platform provides a unique way to design environments without this drawback. Instead of trying to re-shape a fixed framework, each environment describes the set of components that it requires, including the layout with an XML[8] page. Once the environment is selected, its XML page is parsed, loading into memory the objects described in the page. The layout information is used for GUI initialization. All the loaded components are accessed through a trader object that becomes the root of a flat automation model, that is the list of components described in the XML page. If the components have interdependencies, they are loosely connected once the framework is fully compiled. (See Figure 7.)

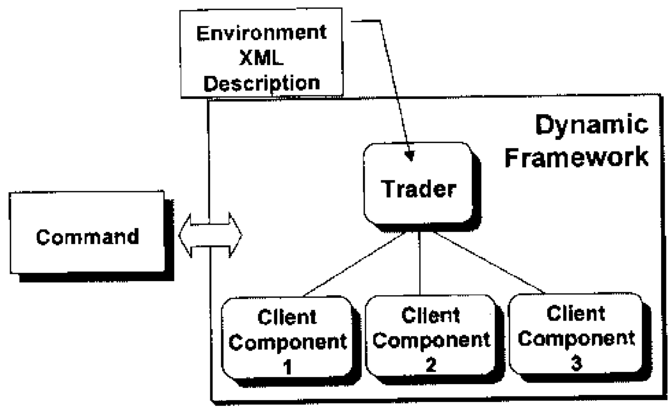


Figure 7. Dynamic Component Framework

This unique technique allows people to dynamically create environments from any COMPASS component, as well as re-usable components from third-party vendors (ActiveX, controls, applets). (See Figure 8.) The environments load only the necessary components and do not require detailed knowledge of any proprietary framework. Note that each environment can be persisted in sessions files (i.e., saved in the client space) retaining the status of the components when a change in the environment is executed if the session is saved.

In this schema, commands are just standard ActiveX DLLs running against the flat automation model described by the XML page. Commands can be written easily with any language supporting COM objects (applications example delivered with the toolbox shows command examples in VB, J++, and C++). In the same way, all the GUI components, toolbars, ribbon bars, and forms, are just standard controls that can be created with Visual Studio.

Large applications such as shipbuilding can be split into separate environments. The screen is uncluttered with full sets of GUI tools, showing only the specific environment commands. There is also full re-use of components since commands (being components themselves) can be shared across environments. When the end user switches between environments, the corresponding XML pages are compared and the common elements are kept in memory, allowing a smooth change over of the dynamic framework. The COMPASS toolbox also contains 2D controls that are able to parametrically edit 2D profiles. In this way, any 3D application allows the editing of 2D shapes (i.e.,

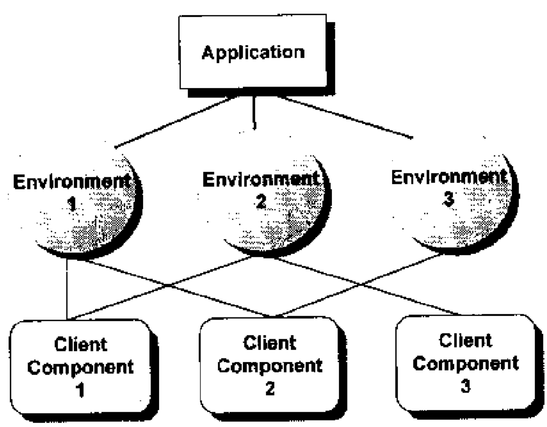


Figure 8. Re-use of Components

[8] XML "Extensible Markup Language," http://www.w3.org/TR/PR-xml.html.

profiles for extrusions, revolutions, drawing generation).

Conclusion

Having exposed the problems that traditional CAD systems have in handling complex applications such as shipbuilding, the COMPASS Program demonstrated the following research conclusions:

- Moving from 3D geometry file-based applications to general applications storing data in databases is key to handling large projects efficiently.
- The DNA open architecture is key to controlling and reducing development costs, learning curve, and scalability.
- An open, logical schema is critical for third-party vendors in order to interface their existing applications into this architecture.
- Formalizing the relationships for business rules ensures a full data consistency without burden for the application writers.
- The notion of independent environments simplifies the GUI and facilitates the integration of multiple modules as building blocks.

References

[1] STEP "Standard for the Exchange of Product Model Data," http://www.nist.gov/sc4/www/stepdocs.htm.

[2] Windows DNA "Building Windows Applications for the Internet Age," Stephen Rauch, Platform Strategy and Architecture, PDC, September '98, Denver, USA.

[3] COM "Common Object Model," http://www.microsoft.com/com.

[4] UML "Unified Modeling Language Resource Center," http://www.rational.com/uml.

[5] Bernstein, P.A., and U. Dayal, "An Overview of Repository Technology," International Conference on Very Large Data Bases, Morgan Kaufmann Publishers, San Francisco, 1994, pp. 705-713.

[6] Patent US5692184: Object relationship management system.

[7] OLE for Design and Modeling Applications, Microsoft, January 24, 1995, Redmond, USA.

[8] XML "Extensible Markup Language," http://www.w3.org/TR/PR-xml.html.

The Microsoft Repository, Philip A. Bernstein, Paul Sanders, Brian Harry, David Shutt, Jason Zander, Proceedings of the 23rd VLDB (Very Large Data Bases) Conference, Athens, Greece, 1997.

OVERVIEW OF ADVANCED CIM FOR SHIPBUILDING PROJECT

– VISION AND REFERENCE ARCHITECTURE

Ken Ito, Masahiro Sonda , Atsushi Fujita, Satoru Endo,
Ship & Ocean Foundation, Tokyo, Japan

Abstract

The Advanced Computer Integrated Manufacturing (ACIM) for shipbuilding project sponsored by the Nippon Foundation and participated by seven major shipbuilders in Japan, was launched in 1997 with a planned duration of three years. The goal of ACIM project is to enable highly effective, efficient and flexible collaborative engineering for shipbuilding based on knowledge sharing among multiple disciplines by developing practical product/process model and promoting network based software integration environments. To achieve the goal, we have developed a CORBA-based reference architecture that enables integration and management of distributed objects and systems including newly developed applications and existing legacy systems. The aspect of the construction of the future virtual enterprises has also been considered in the development of the reference architecture. In this paper, we provide an overview of the ACIM project and present the ACIM reference architecture.

Introduction

Since 1989, a series of research and development projects on the CIM for shipbuilding has been carried out by Ship and Ocean Foundation together with seven major shipyards in Japan, aiming at drastic increase of productivity in shipbuilding industries. Figure 1 shows the history of CIM projects. The concept of CIM for shipbuilding, which is rather different from mass productive industries such as automobile industries, was established in the first project. The importance of ship product model was recognized and has since been emphasized throughout the CIM related projects. The specification of product model was established in the second project. Object oriented technologies have been widely applied for the system analyses as well as system implementation since the first project. In the GPME project, the preceding project of ACIM, the general product model environment was developed for the efficient development of product model. The results generated from these R&D activities have been transferred to the Japanese major shipyards. All seven shipyards have established their own CIM systems based on the results of the CIM research projects.

The ACIM project was launched in the spring of 1997 with a planned duration of three years. The vision of ACIM project is to enable highly effective, efficient and flexible collaborative engineering environment for shipbuilding based on knowledge sharing among engineers from multiple disciplines. Developing product and process models is one of the major R&D tasks in this project since they are the basis for knowledge sharing. Another important task is to establish a distributed object and system environment based on CORBA (Common Object Request Broker Architecture). Other emerging information technologies such as agent technology is also introduced for better engineering work support and information exchange through network.

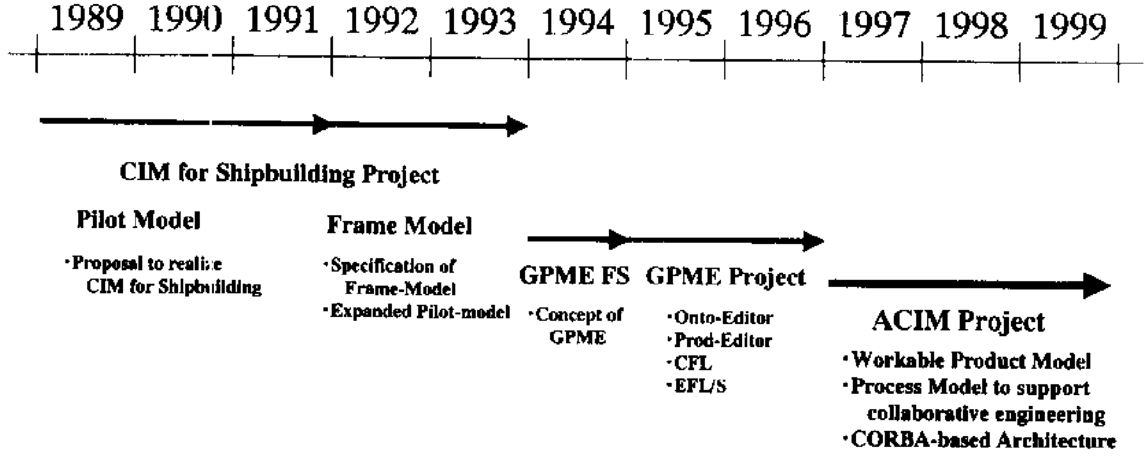


Figure 1. History of SOF CIM Project

Goal and Objectives

As stated above, the overall goal of ACIM project is to develop a system framework to support the collaboration among engineers in design and production in shipbuilding.

The specific goals of the project include:

1. To develop a CORBA-based ACIM reference architecture that enables integration and management of distributed objects/models in the network and supports concurrent engineering of shipbuilding
2. To construct a practical product model through the extension of the Frame Library developed in the preceding GPME project
3. To define a process-driven collaborative engineering work paradigm
4. To develop a process model for engineering support that can adapt dynamically changing actual engineering process
5. To develop intelligent agent based collaboration support mechanism
6. To test the system infrastructure using real shipbuilding case scenarios

The deliverables generated by this project, such as product model and applications, are not prototype systems but are practical application systems and will be used in actual ship design and production. "Practical",—i.e., real—is one of the key words in this project. Therefore, the issues related to deployment of the deliverables to the shipyard are also included in the project scope.

Needs for Knowledge Sharing

Shipbuilding is a typical example of concurrent engineering where the information is exchanged intensively between engineers at different stages and continuously updated by multiple disciplinary engineers. The main goal of ACIM project is to develop a system infrastructure to support this collaboration by using advanced information technologies and to achieve information and knowledge sharing for high efficiency.

To share the knowledge and information about shipbuilding among multiple engineers and furthermore between engineers and computers, one needs both product and process models. Sharing engineering information through a common product model has been proven to be an effective way for engineering team members to identify inconsistencies and generate timely information flows. By accessing a logically centralized product model database, an engineer can retrieve the latest engineering information generated by other engineers and pass his or her task results to others through the database.

While shared product model can be used to facilitate information flows among engineers, it contains only the results generated by engineers. The information of the process through which the results were generated is not part of a product model. Another important aspect of engineering knowledge is about engineering processes. Engineering tasks are not carried out in a random way. Rather, they are well planned and managed based on the planned processes. An engineering process is composed of a set of interrelated activities that collectively realize certain engineering objectives.

Furthermore, there are many applications prepared for the engineering support for design and production. These applications are developed based on shipbuilding specific domain knowledge. It can be said that the specific knowledge of design and production of shipbuilding is expressed by product model, process model and applications that are distributed in the network.

The information technology about distributive objects is emerging nowadays. This technology secures interoperability and adaptability between software systems. Interoperability means how easily a software can be integrated into various other systems and adaptability denotes how one system can be upgraded following the latest technology. This information technology of distributed objects enables us to construct a network-based collaborative environment for ACIM.

Common Object request Broker Architecture

OMG is a software standardization organization that promotes Object-Oriented technologies for integrating heterogeneous and distributed computer systems. OMG's system building concept is described as OMA (Object Management Architecture), the architecture aiming at global network oriented software linking environment. The standard defined by OMG is called CORBA (Common Object request Broker Architecture). It is natural way to adopt CORBA standard to realize above mentioned ACIM collaboration support environment. The specification of CORBA defines how to make interface which is compliant to OMA's architecture. CORBA enables OMA concept in the following four mechanisms.

- OMG IDL (Interface Definition Language) is used to define software interface grammatically.
- ORB (Object Request Broker) is incorporated to simplify the transaction in heterogeneous and distributed computer system environment.
- OMG IDL written interface facilities called Basic Object Service and Common Facility are defined.
- CORBA Object-Oriented approach enables interoperability and adaptability.

CORBA basic object services are implemented by software venders as CORBA product. Software components with interoperability and adaptability can be developed by use of CORBA product. CORBA can mask the discrepancy of computer systems (i.e. hardware platforms, software language, data access mechanisms, compiler versions, component module interfaces, network protocol etc.) and there are several wrapping methods to accomplish it.

ACIM Reference Architecture

As described above, CORBA, the distributed object technology, is an important information technology to build the system infrastructure for ACIM since we want to achieve knowledge and information sharing through the integration of applications and data without worrying about hardware or software platforms.

This CORBA-based system environment enables to continuous extension of ACIM through the addition of the new software components and the integration with existing systems. In order to construct the ACIM based on distributed objects environment, it is required to define reference architecture as the system development framework. The architecture is basically designed by referring to the OMA proposed by OMG. The specific requirements that comes from the concept of the advanced CIM for shipbuilding system has been taken into consideration for construction of reference architecture.

1. Product model that represents knowledge and information about ship product and process model that represents knowledge and information about ship engineering process are two core models for knowledge description.
2. The effective services/facilities to support collaboration of engineers, which can be commonly used in ACIM environment, have to be provided.
3. It should be described in the architecture that the ACIM can be enhanced according to the preparation for the domain specific services concerning ship structural design and pipe design.
4. From the practical view point, special consideration must be paid to the integration with legacy systems used in the shipyards.

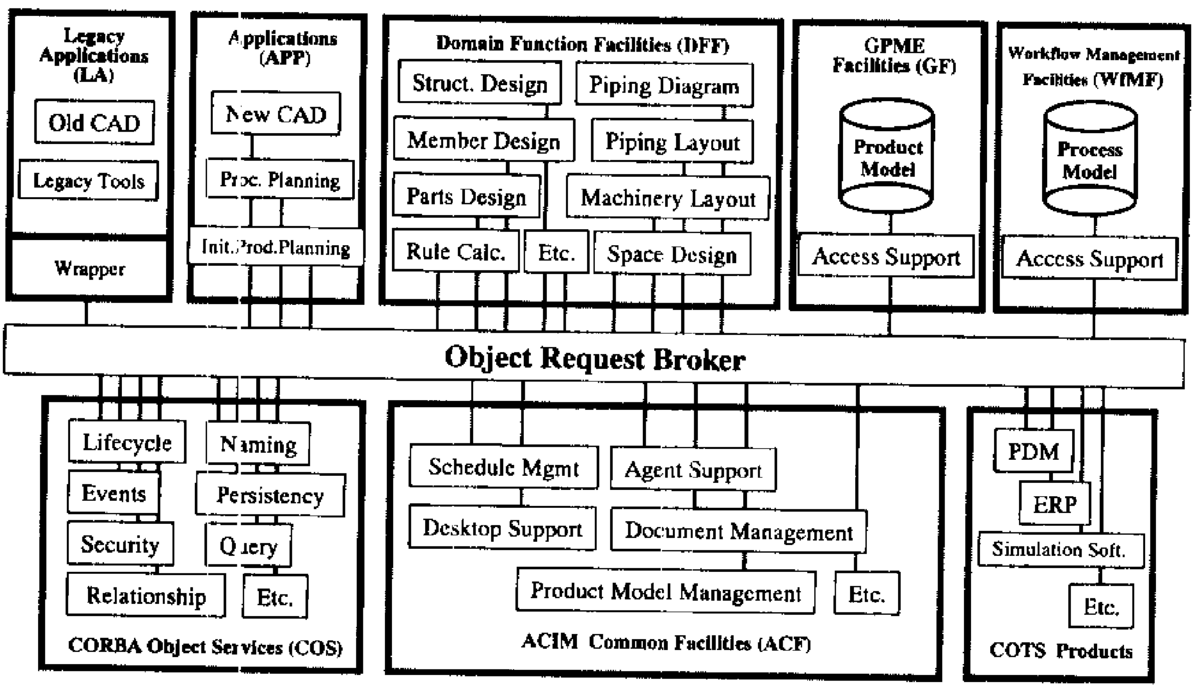


Figure 2. ACIM Reference Architecture

The ACIM reference architecture is shown in Figure 2. In this architecture, eight (8) interface categories are defined according to the functionality of software components which provide necessary facilities and services to support concurrent engineering of engineers. ORB plays an important role to associate these software components included in these categories.

In the following, we describe the functionality of each category and implementation of these categories.

CORBA Object Services : COS

This category is corresponds to the Basic Object Services, one of architectural components of OMA described in previous section. These services are the basic services to construct distributed object system. The specifications on functionality of each service are defined by OMG and software venders implements its functionality based on these specifications. There are several commercial CORBA products available nowadays. The specification of object services has been already defined by OMG and published. At the moment, there are sixteen (16) services defined as specification, such as "Naming service" for linking object's name and real object, "Event service" for asynchronized communication. In the ACIM project, we adopted "Orbix" from IONA Technology as the CORBA product.

GPME Facilities : GF

GPME facilities provides the access facilities to the product model which is implemented in the previous GPME project and extended to the practical level in the ACIM project. The versatile access to the product model is available through open natured IDL interfaces. The instantiation and retrieval of instances from product model are performed through this services. The important point is that we have adopted the late binding method for implementation, so all the class methods which have been already implemented in Frame Library and also new methods which may be added in future are available as a CORBA objects in ACIM environment through this access facilities. The late binding enables to specify dynamically the functions or method when application is executed. This means it is not necessary to define the IDL interfaces and implement it one by one. Even in the case that extension or modification of certain method is occurred, only the recompilation of source program has to be done.

Workflow Management Facilities : WfMF

As was described above, we regard the process model as one of core model to express knowledge about shipbuilding process and to realize collaborative engineering environment. In ACIM project, we propose a process-driven work paradigm for collaboration support. The paradigm will be briefly described in the succeeding section and the details are presented in another paper submitted to ICCAS.

This category provides the necessary functions to realize process-driven work paradigm. Two important servers have been identified as system components of work support system. The components are Process Server and Enterprise Resource Server. The functions of Process Servers include capturing process models and managing process execution based on the process models. For process execution management, Process Server generates activation signals, control the real process by reasoning based on the given rules. On the other hand, Enterprise Resource Server centralizes resource information logically. To manage engineering processes, resource information is very important. It has also process templates and journal information about the results of engineer's operation or communication. The services corresponding these functions are provided by WfMF facilities.

Domain Function Facilities : DFF

DFF is the category which includes shipbuilding specific functions necessary to develop applications. The DFF facilities provide the services based on the shipbuilding ontology. The applications are developed mainly on the basis of these DFF facilities. The functions are dependent on the kinds and functions of applications to be developed. Theses DFF facilities are sometimes used for several applications, and furthermore one DFF function may be used by other DFF facilities as the primitive services. The DFF facilities provide functions for application development by combination of GF services such as assessor and methods which GF provides.

As mentioned before, the services provided by GF are implemented by late binding method. This means all the methods provided by product model are available as the CORBA facilities. The methods included in the GF category contains rather higher level functions such as calculating the weld length and weight calculation about intermediate assembly products. Due to the late binding method, these functions are provided by the GF facilities. The examples of DFF are shown in the another paper submitted to ICCAS.

Applications : APP

In this reference architecture, applications are defined as the new applications used in shipbuilding design and production which has been developed on the basis of the ACIM reference architecture and adapted to the CORBA-based ACIM open environment, i.e. distributed object environment. In ACIM environment, applications are developed efficiently combining the services and facilities provided by software components included in other categories of ACIM reference architecture. In this project, two (2) applications for production planning are developed in a practical level for the purpose of the verification of ACIM product model.

Legacy Applications : LA

This category contains all the applications, such as CAD systems and other domain specific applications to support design and production work, currently used in the shipyard. These applications does not have the adaptability to the CORBA environment. So it is impossible to communicate with other applications through CORBA. However, from the practical view point, it is very important to exchange information between these legacy applications and software object developed under the ACIM open environment. So these applications have to be adapted to the ACIM environment. CORBA enables to make these legacy applications adapt to the ACIM open environment. The software added to the legacy applications in order to make it adaptable to CORBA environment is called wrapper. Through the wrapper, legacy applications can communicate with new applications or COTS products adaptable to CORBA environment. On the contrary, it is also possible to make the functional components included in the legacy applications as a software component that can be used by other applications. CORBA provided the great advantage that makes the legacy application adaptable to distributed object environment. This means that the a part of legacy application can survive as a software component for a long time.

ACIM Common Facilities : ACF

Software components in this category provides the general services commonly used in other software components. These components are independent from the domain specific ontology, so these are widely used from many applications. Commercial software components not compliant with CORBA environment can be adapted to the ACIM environment by adding a wrapper and added as the new service. In this way, the new software components can be added easily to ACIM services.

On the other hand, the native services are developed and prepared as the ACF services in this project. These are agent facilities and product model management facilities. As theses facilities are independent from shipbuilding domain ontology, services are categorized as ACF.

Example of services provided are schedule management, agent supported functions that may be used in the process-driven ACIM collaborative environment, product model management facilities that is indispensable for operational phase in a shipyard. Furthermore, ACIM desktop, work environment for ACIM users, is developed in this project. Services necessary to develop ACIM desktop are provided as ACF.

COTS (Commercial-Off-The-Shelf) Products

This category includes the COTS products which provide CORBA compliant services. The concept of integration of distributed systems based on CORBA is widely accepted. Software venders such as PDM(Product Data Management) and ERP(Enterprise Resource Management) are now moving to adapt their software products to CORBA. Furthermore, the consortium of PDM venders are trying to define the standardized IDL interfaces aiming at the information exchange between different PDM products. Also it is expected that the software venders which are specialized in development of software component will emerge in near future according to the progress of the distributed object technology progress. By using best-breed COTS and/or combining effectively commercial software components, the application development environment will be much improved.

On the other hand, in the world of Microsoft Windows, distributed object linking mechanism DCOM(Distributed Common Object Model) has been formalized. Many software component products are available nowadays. For example, Microsoft Transaction Server can be easily customized by selecting and combining software components implemented as the DCOM objects. As will be explained later, it has already been confirmed through the prototype test that the information exchange between CORBA and DCOM can be realized with ease by use of CORBA/DCOM bridge provided by CORBA commercial products. This means that the libraries compliant for DCOM can be used from CORBA world. Therefore it is expected that increase of commercial software components compliant with DCOM enable efficient application development even in CORBA environment.

As was described above, the Advanced CIM system which enables highly efficient collaborative engineering can be realized on the basis of CORBA-based ACIM reference architecture. The flexible exchange of information and knowledge through network can be realized by CORBA. Services physically distributed in the network can be used regardless of the hardware/software platform. The information exchange between applications become possible over the enterprise.

Verification of CORBA Based System Through Prototyping

In order to clarify the technical issues prior to the actual development we have carried out the verification of CORBA-based system by prototyping. The scenario of prototype system is as follows;

- Firstly, a test application access to the product model to get all parts from specified assembly block
- Secondly, the calculation of weight and welding length of specified assembly block is carried out by using services included
- Display the results to a screen in graphical way by Java applet

The prototype scenario is very simple one, and it includes services of several categories such as CORBA Object Services(COS), Domain Functional Facilities(DFF), GPME Facilities(GF) and

Applications(APP), and it is enough to verify the possibility about the CORBA based systems. Figure 3 shows the prototype test environment.

We have confirmed through this prototype system that there is no specific technical issue for developing the CORBA-based system. The performance overhead is not so big in case that the granularity of service is not so small. We have also carried out another prototype test concerning about CORBA/DCOM linkage. The test scenario is to get information from product model on the EWS and to transfer it to the spread sheet on the PC. The functions provided by commercial CORBA products as a CORBA/DCOM bridge was used in the test. We have confirmed that the data exchange between CORBA object and DCOM object can be realized without any problem. This means that the services implemented as DCOM object can be utilized in ACIM environment.

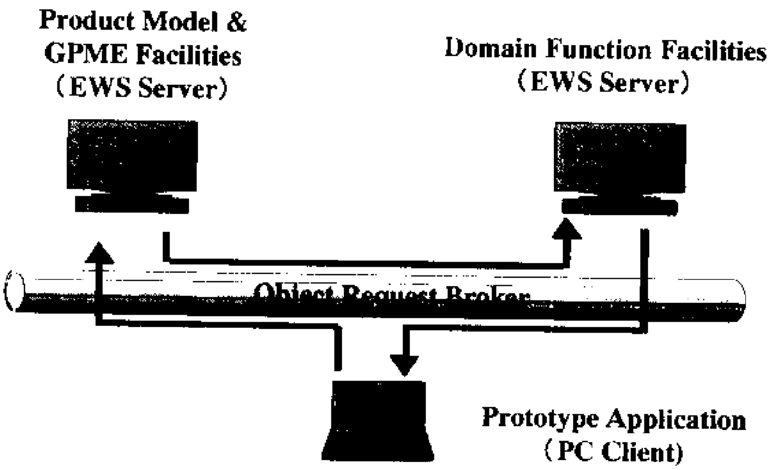


Figure 3. Prototype Test Environment

Development of Practical Product Model for Shipbuilding

Now, we will explain another important issues in the ACIM project. In the ACIM project, it is recognized that product model which expresses knowledge and information concerning ship as a product plays key role for the knowledge sharing. Thus, the product model should have enough ability to express information and knowledge on actual ships and should be practical. One of the important R&D tasks in the project is to establish the practical product model for shipbuilding, which express ontology of shipbuilding. When developing the workable product model, maximum usage is made of GPME (General Product Model environment) which has been developed in the preceding CIM project. In the GPME project, CFL(Common Frame Library), the information model to express common ontology for assembly industries has been extended to EFL/S (Extended Frame Library for Shipbuilding) which includes the ontology for shipbuilding. However, it was a relatively small extension only for verifying the usefulness of GPME as a product model development software tools. In the ACIM project, extension of EFL/S is being carried out to much more practical level. The product model is realized by combined use of CFL and EFL/S and becomes the base for the engineer's collaboration. EFL/S is now being extended in two directions as shown in Figure 4. One is to enhance the product model so as to be rich in the modeling flexibility; this lead to the easy design alteration and simulation. Another is to extend the expression capability so as to express most of the hull structures including details and almost all outfitting of actual ships. The details of the practical product model construction is referred to paper submitted by other authors.

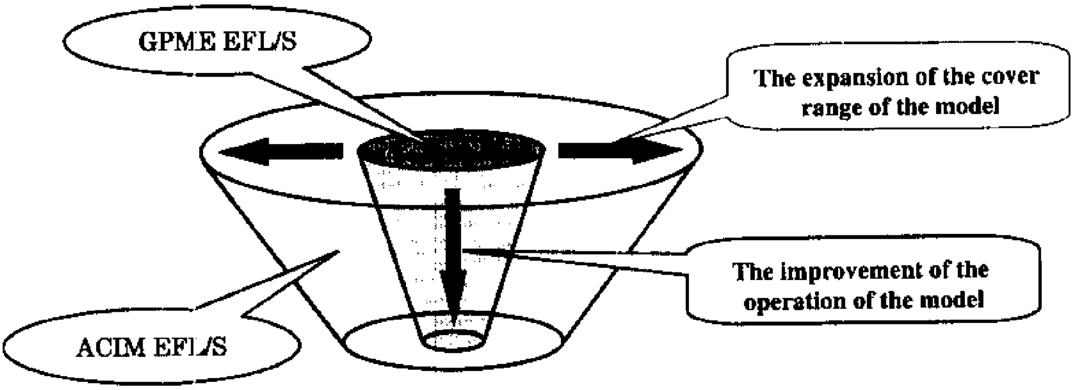


Figure 4. Extension of EFL/S in Two Directions

Development of Process Model To support Collaborative Engineering

While knowledge about engineering products captured by product models is important for engineering support and needs to be shared among distributed engineers and systems, another important aspect of engineering knowledge is about engineering processes. In ACIM project, the process model that represents the knowledge and information about engineering process is one of the core models for collaboration through knowledge sharing. Engineering tasks are not carried out in a random way. Rather, they are well planned and managed based on the planned processes. An engineering process is composed of a set of interrelated activities that collectively realize certain engineering objectives. A process model is a representation of a engineering process in a form which supports automated manipulation or enactment by a process management system. Process model includes the control information that defines how engineering activities should be carried out and how they relate to each other.

In ACIM project, we propose a framework based on a process-driven and agent-supported approach to collaborative engineering support called Active Process. The process model developed in Active Process has a capability to describe 'how engineering tasks should be done' in order of precedence. It is powerful but still not sufficient because engineering task procedures will be often changed according to situation. To meet this problem, agent-supported approach is introduced to Active Process. Each participant in the system has its own agent who is interested in negotiation with others to acquire information.

Figure 5 illustrates how process information represented as process models can be used to support engineering work and collaboration. The details of the proposed framework is presented in another paper submitted by other authors.

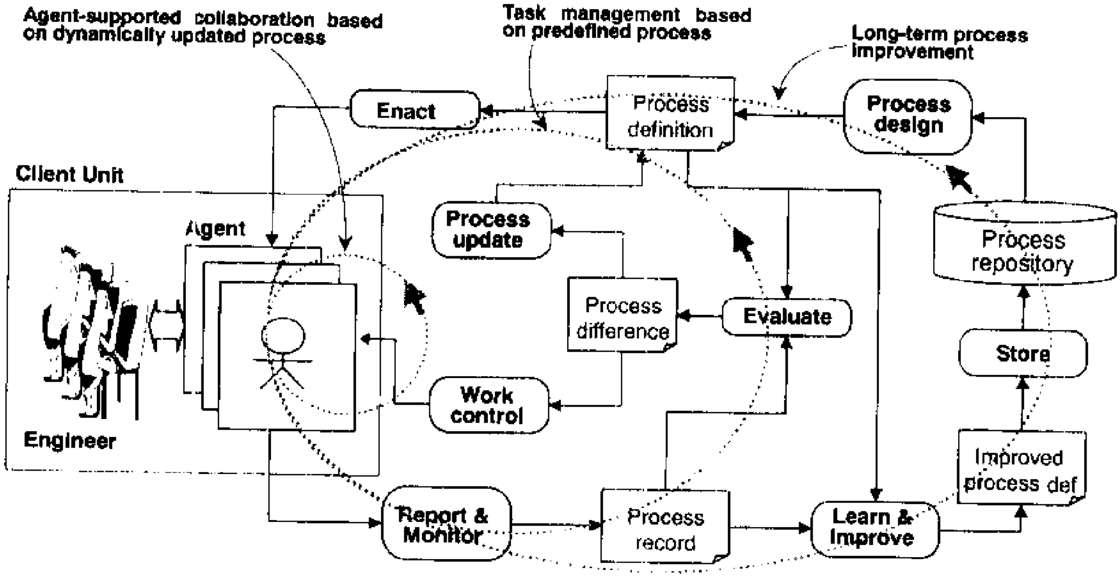


Figure 5. Process Model and its Application for Engineering Support

Verification of Product Model and Collaborative Engineering Environment

Verification of Enhanced and Extended Product Model

As was stated in the previous section, product model (mainly the EFL/S) is now being extended in two directions, i.e. one for enhancement on the functions such as modeling flexibility and the other for the extension of expression capability. We take different verification approach for these two extensions.

As for the verification of enhancement of functions such as modeling flexibility, we take an approach to verify it through the development of applications and their execution for the enhanced product model. The application is being developed by using the class methods provided by enhanced product model. The application must be a practical one, not a prototype, in order to evaluate functionality of product model more correctly. Two applications are now being developed for this purpose. One is an application to support initial production design so-called block breakdown (ACIM IPD : Initial Production Design), while another is an application to support process planning (ACIM CAPP : Computer Aided Process Planning). These applications are developed on the basis of the ACIM reference architecture. The services provided by GF, DFF and COS of ACIM reference architecture are used to develop these applications. Details of the applications are described in the paper submitted by other authors.

As for the verification of the extension of model expression capability, we take an approach to verify it through the exchange of model information for actual ships generated by CAD system being used in the shipyard. As the model data of shipbuilding CAD system now in practical use can express most of the hull structures including details and almost all outfitting of actual ships. Therefore, if these information can be transferred to the ACIM product model and successfully stored, it is confirmed that the extended product model has enough coverage to express actual ship structures and outfitting items.

In order to exchange data between product model and CAD system in shipyards, we are now developing the dual-direction data transmission application. This application is also developed on the basis of the ACIM reference architecture. The data exchange test will be carried out using this software. As the software is so developed as to be able to exchange in dual direction, the results obtained from ACIM IPD and CAPP described above is transferred to the existing CAD system in shipyards and continuously used in the practical design. This means that this data exchange software can be used in a actual ship design process.

Verification of Collaborative Engineering Environment

The goal of ACIM project is to establish the system infrastructure using advancing information technologies which enable efficient and flexible collaborative engineering environment for shipbuilding on the basis of the knowledge sharing. We plan to verify it based on a set of test scenarios. In the test scenarios, all the deliverables obtained from the project such as practical product model, applications of ACIM IPD and CAPP and process model are taken into consideration. In the project, we are planning to propose the ACIM desktop which is newly developed user's work environment with enhanced graphical interface. The process-driven and agent-supported mechanism for collaborative engineering support mentioned in the previously section will work on this ACIM desktop. The verification of collaborative engineering environment will be carried out through this desktop according to the test scenario.

Conclusions

In this paper we presented an overview of our on-going ACIM project, focusing on the ACIM reference architecture. The vision of ACIM project is to enable highly effective, efficient and flexible collaborative engineering environment for shipbuilding based on knowledge sharing between multiple disciplines by developing practical product/process model and promoting network based software integration environments.

In order to achieve our goals, we have developed an OMG CORBA-based reference architecture that enables integration and management of the distributed objects, models, and systems including newly developed applications and existing legacy systems currently used in shipyards.

Our proposed ACIM reference architecture represents a framework of the advanced CIM systems. This architecture is open to the outer world and adaptable for the emerging information technologies. The product model and process model are defined as the core components of knowledge sharing ACIM environment. The practical product model is developed in this project. This product model is used in the shipyard after this project is completed.

We propose a framework based on a process-driven and agent-supported approach to collaborative engineering support called Active Process.

The verification of the enhanced product model and collaborative environment is carried out through the newly developed applications.

The progress of information technology is being accelerated nowadays. The ACIM reference architecture based on the distributed object oriented paradigm is very flexible enough to incorporated the new technologies. We believe that the ACIM reference architecture provides a platform for continuous improvement of CIM for shipbuilding.

Acknowledgements

This project has been carried out under the sponsorship of the Nippon Foundation, and the authors would like to express their cordial thanks to them. They would also like to express their gratitude for the guidance and encouragement received from Prof. Takeo Koyama, University of Tokyo and Prof. Yan Jin, University of Southern California.

References

1. Koga, K., CIM System and Product Model for Shipbuilding, ICCAS 94
2. Nagase, Y. et al., GPME Modeling Methodology and Applications, ICCAS 97
3. Kawazoe, H. et al., GPME: An Information Modeling and System Integration Environment for Shipbuilding, ICCAS 97
4. Nagase, Y. et al., On the Development of ACIM Product Model and its Verification through Practical Applications, ICCAS 99
5. Sonda, M. et al., Using Process Models and Intelligent Agents to Support Collaborative Engineering in Shipbuilding, ICCAS 99
6. http://www.omg.org, OMG's home page
7. Ship & Ocean Foundation, 1997 Report on Research and Development of Advanced CIM for Shipbuilding based on Knowledge Sharing.
8. Ship & Ocean Foundation, 1998 Report on Research and Development of Advanced CIM for Shipbuilding based on Knowledge Sharing.

HIGH LEVEL LANGUAGE FOR SHIP DESIGN

F. Alonso. Sener, Ingeniería y Sistemas, S.A., Tres Cantos, Madrid, Spain
R. M. Tronstad. Sener, Ingeniería y Sistemas, S.A., Tres Cantos, Madrid, Spain
J. Subias. Sener, Ingeniería y Sistemas, S.A., Tres Cantos, Madrid, Spain

INTRODUCTION

The CAD/CAM Systems currently used in shipbuilding are powerful tools, essential nowadays to perform in an efficient and precise way all the design and manufacturing processes of a ship.

The evolution of these Systems over the last 25 years led from an initial situation, covering in a limited way the design and manufacturing processes (basically naval architecture and N.C cutting), up to the present situation, covering most of ship design and production tasks.

There has been also a big change, from the initial applications, basically in batch mode or with limited interaction (mainly alphanumeric), to the present graphic interactive applications, working with 3D models and with a high degree of interaction between the user and the application.

As a result, the applications have become very complex, requiring a high degree of customisation and flexibility in their operation.

All this has generated new user requirements to achieve a greater efficiency and flexibility in the use of the CAD/CAM Systems.

These new user requirements can be summarised as the need for a macro language allowing the users to perform repetitive tasks, to define their own standard constructive solutions, and in general to create "intelligent" programs able to access the functions and commands of the System.

As an answer to this important requirement of the users of its FORAN CAD/CAM System, SENER decided to develop the new FORAN Macro Programming Language (FMPL) as a general tool to be added to all the modules of the System.

The new FORAN Macro Programming Language has been developed using the Tcl/Tk scripting language, extended with the capability of executing commands and other utilities of the System.

OBJECTIVES OF THE PAPER

This paper intends to describe the main characteristics of the new FORAN Macro Programming Language, making special emphasis on the characteristics relevant to a user of ship design applications.

With this objective in view, the paper has been structured in the following parts:

- A short introduction to Tcl/Tk, the basic language used for the development of the FMPL.
- A description of the objectives, main characteristics and structure of the FMPL.
- A presentation of some practical cases showing the application of the FMPL to different areas of ship design.
- An analysis of the future developments of the Tcl/Tk language and the FMPL.
- A summary of the results obtained up to this moment, with an estimation of the savings and advantages that the users can expect from the normal use of this Macro Programming Language in their daily design tasks.

A SHORT INTRODUCTION TO Tcl/Tk

Tcl is one of the most widely known scripting languages freely available through the network. A scripting language, as opposed to a system programming language (such as C or C++), is designed to integrate, connect or "glue" basic components, commands or applications, [JO'98]. Tk is an extension to Tcl, a toolkit to build Graphical User Interfaces.

Tcl/Tk is an interpreted language with many features available in standard procedural languages (variable assignment, procedure calls, flow control structures, etc). The basic data types are strings, lists and arrays. To speed up the process, a compiler is also available with the latest Tcl/Tk version.

Among the many advantages that Tcl/Tk offers, we can note the following:

- It is easy to learn.
- It allows rapid development of applications.
- It is available in many platforms (Windows/NT and UNIX, including LINUX).
- It strongly facilitates the embedding in the language of new functions or commands developed in other languages like C or C++.
- It is free, as well as many of its extensions.

All these advantages, as well as the availability of tools to build Graphic User Interfaces (the toolkit Tk), induced SENER to select Tcl/Tk among other scripting languages (Perl, Visual Basic or Python) as the basic language to develop the FORAN Macro Programming Language.

On the other hand, although Tcl/Tk provides many useful tools and capacities, the power and flexibility of a Macro Programming Language built on top of a scripting language like Tcl will mainly depend on the components, functions or commands of the basic application (in this case FORAN) made available to the users through the scripting language.

THE FORAN MACRO PROGRAMMING LANGUAGE

The FORAN Macro Programming Language is a new macro language build on top of the Tcl/Tk scripting language, to extend and complement the interactive command capabilities of FORAN.

The Macro Language uses all the power of Tcl/Tk extended with the possibility of executing FORAN commands and other utility functions.

The Macro Programming Language has been developed to fulfil different objectives:

- To allow the users to repeat sequences of commands, to facilitate the execution of repetitive tasks during the design activities.
- To generate geometry macros for the 3D definition of geometric models of equipment units, and outfitting and accommodation components.
- To develop more powerful macros accessing all the capabilities of the FORAN command language, and a set of specific FORAN functions for each design discipline.

To achieve the first objective, normal users of the System do not need to know anything about the Macro Programming Language or about programming in general. More advanced users trying to reach the two last objectives need some more knowledge of Tcl/Tk and of the FMPL.

A schema showing the integration of FMPL inside the FORAN modules can be seen in Figure 1.

Macros can be executed in the same way as other interactive commands of FORAN, and also can be added to the standard menus of FORAN as commands or icons.

Macro recorder utility to generate sequences of commands

Using this utility, users can generate macros containing sequences of commands. The utility opens a console to add interactivity and control to the automatically generated macro.

The recorder console permits the user to decide which actions should be interactive when reprocessing the macro. The console has also functions for displaying user messages, adding pauses, storing the current viewing attributes or storing the scene.

For more advanced users, this utility can be used for the creation of skeletons of new macros to be completed or modified later on using the text editor.

Generation of geometry macros

A geometry macro is a user defined program which performs a 3D object layout based on the rules described in the macro.

Geometry macros can be used to define 3D representations of pieces of equipment, piping and ducting components, accommodation elements and outfitting structures (supports, hangers, ladders, etc) making it easy to adopt different shapes and dimensions.

The parameters of the macros can be standard values stored in the database or can be given by the user or directly measured from the ship 3D model.

Macro programming for advanced users

A complete use of the capabilities of the FMPL allows advanced users to combine all the power of Tcl/Tk with all the existing FORAN capabilities.

By extending the script interpreter with new FORAN functions (libraries), the macro language will not only be able to process FORAN commands but also to call library functions that will allow to implement new capabilities not covered by the existing commands.

To facilitate user interaction, specific functions for text prompting and graphical input have been added to the Macro Language.

APPLICATION FIELDS

The Macro Programming Language has not been conceived as a macro language for a specific ship design area, but as an open and complete language for all the design activities covered by the FORAN CAD/CAM System.

As a consequence, the language incorporates general functions that can be used in all disciplines and specific functions for each design discipline.

To illustrate the capabilities of the Macro Programming Language, we have selected some examples of macros in the different areas covered by the System. These macros have been created using a first prototype of the Macro Programming Language.

Two of the examples (Hull form import and Definition of standard loading conditions) use application functions developed in C, called from the Tcl scripts. The other macros call directly different FORAN interactive commands from the Tcl script.

The prototype will evolve in the near future, incorporating more functions and commands that will allow adding more "intelligence" or "expertise" to the macros.

Definition of the working environment

As we have already indicated one of the simplest uses of the Macro Programming Language can be the repetition of sequences of commands (with or without interaction).

One example of this use can be the definition of the working environment when a user starts a new working session.

To establish this environment, the user needs to enter a set of interactive commands to read the whole information of the working area (including hull structure, equipment, ducts, pipes, foundations and cable ways), to change the visual parameters and to define the default technological attributes.

All these commands and other common initial tasks can be replaced by a simple macro performing all these tasks (See the result in Figure 2).

As was mentioned before, an important advantage is that the macros or scripts with sequences of commands can be automatically generated during the interactive sessions of the modules of the System.

Hull Form Import

This example shows the results of a macro to import a complete hull form, defined by means of DXF lines, into the System.

In this case the macro is a Tcl script using the following basic functions:

- Opening a file
- Reading a set of DXF lines
- Creation of a boundary curve
- Creation of a knuckle
- Creation of a grid of waterlines
- Creation of a grid of frames
- Closing a file

The macro can be used to import any hull form defined by means of any set of DXF lines. The results of the use of this macro to import the hull lines of a catamaran can be seen in Figure 3.

Naval Architecture

To create the macro shown in this example the following basic functions have been used:

- Definition of a new loading condition.
- Obtaining compartments with some specific contents.
- Definition of a load in a compartment.
- Storing a loading condition.

Combining these basic functions, a macro for the creation of standard loading conditions can be easily created. The macro and some loading conditions created with the macro are shown in Figure 4. This example shows how easy the preparation of a macro can be. The complexity of the macro (and its power) can be increased according to user requirements.

Hull Structure

This case consists of a macro for the definition of all the shell plates of a central zone of a ship.

The macro makes use of the following individual commands of the FORAN Shell and Deck Plate Development module:

- Butt definition.
- Topological point definition.
- Seam definition.
- Panel definition.
- Plate definition.
- Some measurement commands, like distances in development.

The input parameters for the macro are the following:

- Limits of the zone (they can be given interactively).
- Keel width.
- Plate width.
- Plate length.
- Height of seam above double bottom deck.

The macro also establishes some default parameters required for the plate generation, like the material quality and thickness of the plate (they also could be given interactively) and takes some "decisions" depending on the geometry of the shell.

The result of the application of the macro to a bulkcarrier of 182 metres of length can be seen in Figure 5.

The macro allows the creation of 11 butts, 15 seams, 30 panels and 150 plates, and the corresponding symmetrical elements, with a single command, producing a considerable saving of time.

This macro can be considered of medium/high complexity, requiring not only a good knowledge of Tcl/Tk but also of the FORAN command language.

The result of the application of the same macro to a different ship (in this case a chemical tanker of 133 metres of length) can be seen in Figure 6.

Piping

To show the possibilities of the Macro Language in piping, a macro for the automatic generation of different layout alternatives, following the main ship directions between two points, has been created.

The macro makes use of some standard available piping layout commands, including clash detection between the new pipe and the other elements. The macro takes into account some constructive constraints (e.g. bending constraints). The macro also inserts flanges at the ends of the pipe.
The macro builds consecutive layouts in a sequential way and the user may select interactively the most adequate layout.

Starting with the working environment shown in Figure 1, a first piping layout alternative built by the macro can be shown in Figure 7. This alternative presents a clash that is also visualised in the same figure.

Another alternative piping layout built by the macro and without any clash is shown in Figure 8.

Piping supports

Piping supports can be generated using geometry macros with standardized values of the parameters that can be hard coded in the same macro.

The piping support generated by one of these macros is shown in Figure 9, including the parameters of the macro.

Macros can also be used to position the piping supports in piping lines, as can be seen in Figure 10. The nominal diameter will be taken directly from the piping line.

Accommodation

Accommodation is another example of the use of macros. The result of a very simple macro for the generation of sets of accommodation components is shown in Figure 11. The only parameter in this case is the number of chairs.

FUTURE EVOLUTION OF THE FORAN MACRO PROGRAMMING LANGUAGE

The future of the FMPL will depend, firstly, on the evolution of the scripting language on top of which it has been built.

The future of Tcl/Tk

Tcl/Tk is becoming one of the most popular scripting languages available through the web. According to the latest estimations there are about 50,000 new users each month.

One key factor in this dramatic increase of Tcl users has been the constitution of a new company by the Tcl creator, J. Ousterhout. The aims of this new company are to develop the free Tcl core and to give commercial support to Tcl/Tk, creating new development tools, like debuggers, compilers and source code checkers, and promoting the development of new Tcl extensions [Tcl'99].

Among these extensions, because of their interest for future developments of the FMPL, we can point out the following:

- [incr Tcl], [incr Tk] and [incr Widgets], which are object oriented programming extensions.
- OraTcl, to provide access to ORACLE databases.
- Tcl Browser Plugin, to run Tcl/Tk scripts inside web Browser pages.
- TclDP, to provide distributed programming features, including socket access.

New capabilities of the FORAN Macro Programming Language

The FMPL has to follow a continuous process of development, incorporating new functions as a result of the requirements imposed by the users of the System.

In addition to this, two important developments are foreseen for the near future:

- The extension of the language with a set of functions to generate tailor made reports, including text, tabular information and graphics. Functions to extract information from the product model, including SQL calls will be available.

- A new Programming Environment to create new modules that will use all the functionality of the FMPL.

CONCLUSIONS

As has been shown in the practical cases presented in this paper, the use of a Macro Programming Language integrated in a shipbuilding CAD/CAM System can produce significant savings in many ship design tasks, not only for detail design, but also for ship conceptual and basic design.

The main advantages offered by a Macro Programming Language of this type built on top of Tcl/Tk can be summarised in two points:

- ♦ Simplicity of use.
- ♦ Easy access to the commands and functions of a powerful ship design and production system like FORAN.

However, the savings and advantages that can be expected from this new FORAN Macro Programming Language will be determined to a large extent by the degree of compliance of the following goals:

- High adaptability of the Macro Programming Language to the real requirements of the ship design users
- Availability and accessibility from the Macro Programming Language to most of the basic functions and commands of the CAD/CAM System

To achieve these goals, a joint project between ASTILLEROS ESPAÑOLES and SENER has been established. The aims of this project are to identify the design tasks performed with the CAD/CAM System that can be improved by means of the introduction of macros and to specify the corresponding additional functions to be included in the language. The project started the fourth quarter of 1998 and it is foreseen to finish at the end of 1999.

REFERENCES

[BW'97] Brent B. Welch, "Practical Programming in Tcl and Tk", 1997.

[FM'99] "The FORAN Macro Programming Language. User Guide", SENER, Ingeniería y Sistemas. 1999.

[JO'94] John K. Osterhout, "Tcl and the TK toolkit", Addison-Wesley, 1994.

[JO'98] John K. Osterhout, "Scripting: Higher Level Programming for the 21^{st} century", IEEE Computing. March 1998.

[TC'99] Some interesting Tcl web directions: The Scriptics Corporation: www.scriptics.com, The Tcl/Tk consortium: www.tclconsortium.org.

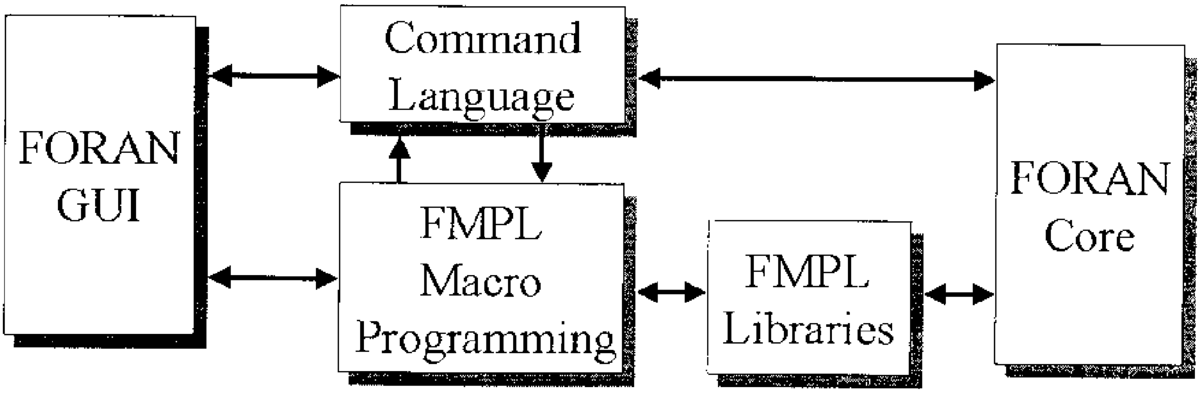


Figure 1 - FORAN Macro Programming Language Schema

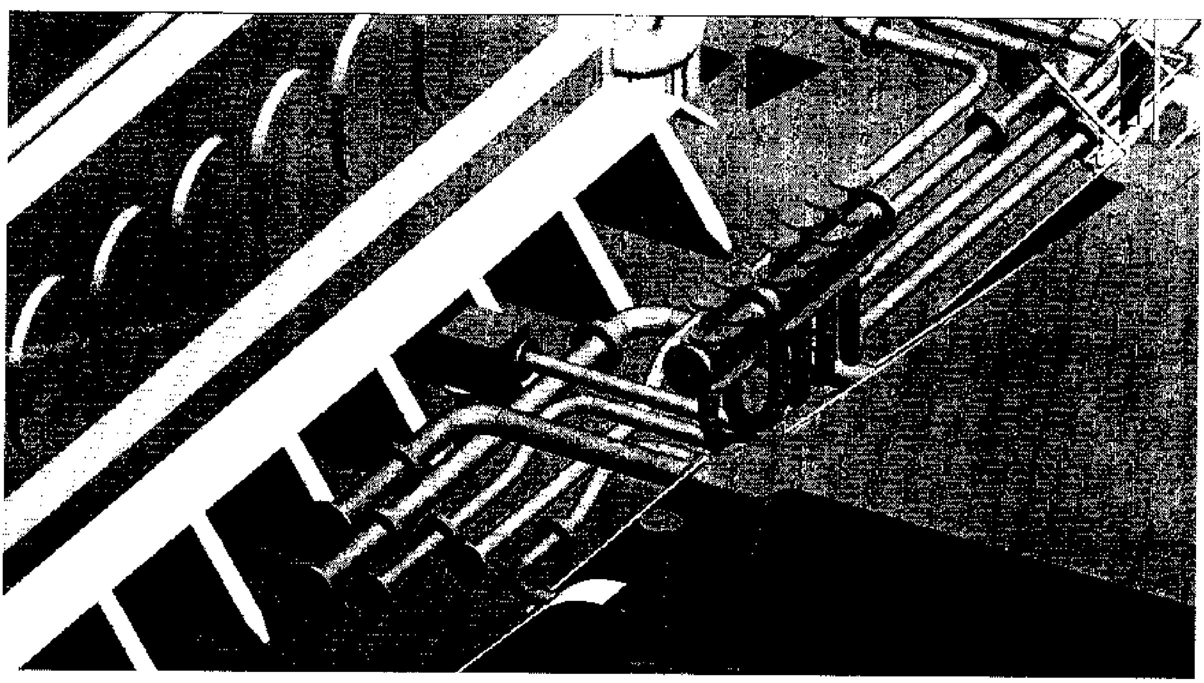


Figure 2 - Working environment defined with a single macro

```
set ballast 1
set fueoil 3
set dieoil 4
set luboil 5
set fwater 6
set load 10
# Assign filling percentage (pfill)
proc fillspaces {content pfill} {
set Lspaces ""
set laux ""
set laux [ ReadLisSpaces -cn $content]
set Lspaces [lsort $laux]
for {set i 0} {$i < [llength $Lspaces]} {incr i} {
  set idsp [string range [lindex $Lspaces $i] 0 3]
#       Sets filling percentage
  LoadInSP -name $idsp -pf $pfill
  }
}
# Main program Definition of Loading Conditions
DefLcond -name BLDE -desc "Ballast departure"
# Selection of spaces with content
fillspaces $ballast 100
fillspaces $fueoil  100
fillspaces $dieoil  100
fillspaces $fwater  100
# Transfer the information
StLoadsInSP
# Store the loading condition
StorLcond
DefLcond -name BLAR -desc "Ballast arrival"
fillspaces $ballast 100
fillspaces $fueoil  10
fillspaces $dieoil  10
fillspaces $fwater  100
StLoadsInSP
StorLcond
```

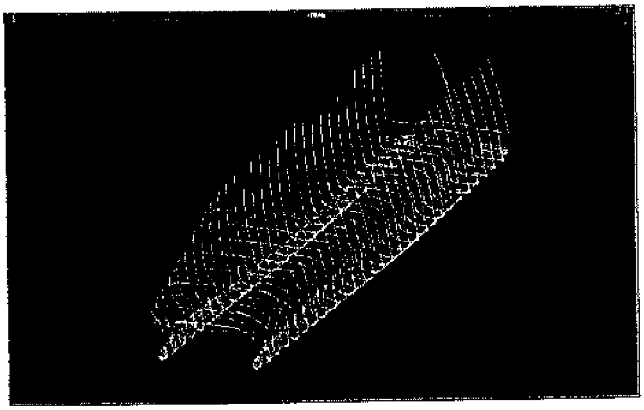


Fig. 3. DXF/3D imported Hull Form

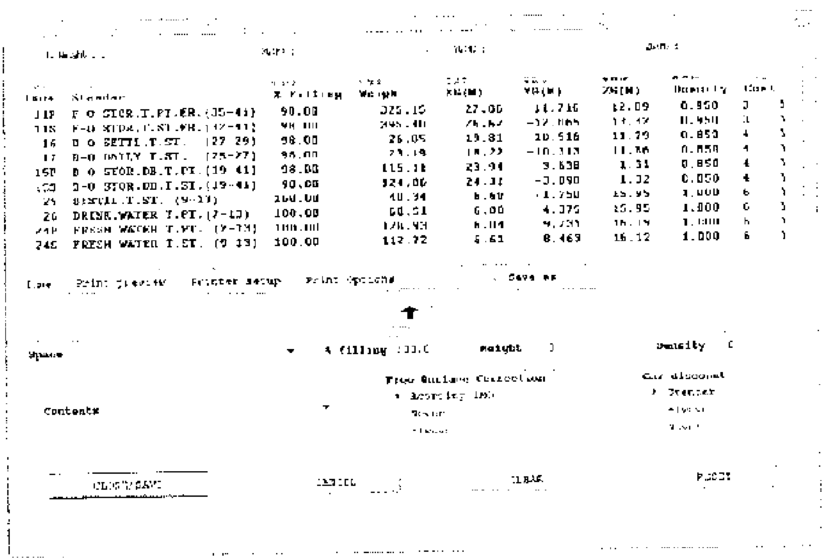


Fig. 4. Macro for definition of standard loading conditions

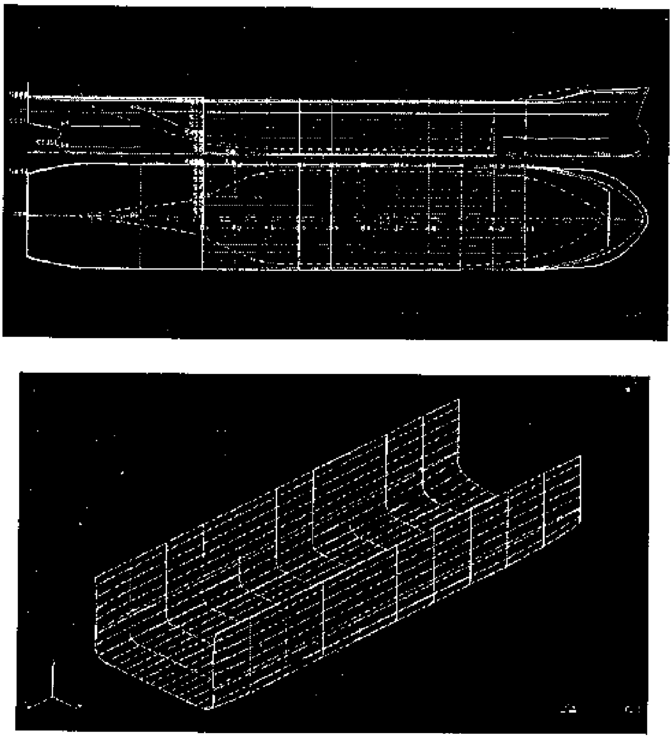


Figure 5. Panel and plate definition. Bulkcarrier

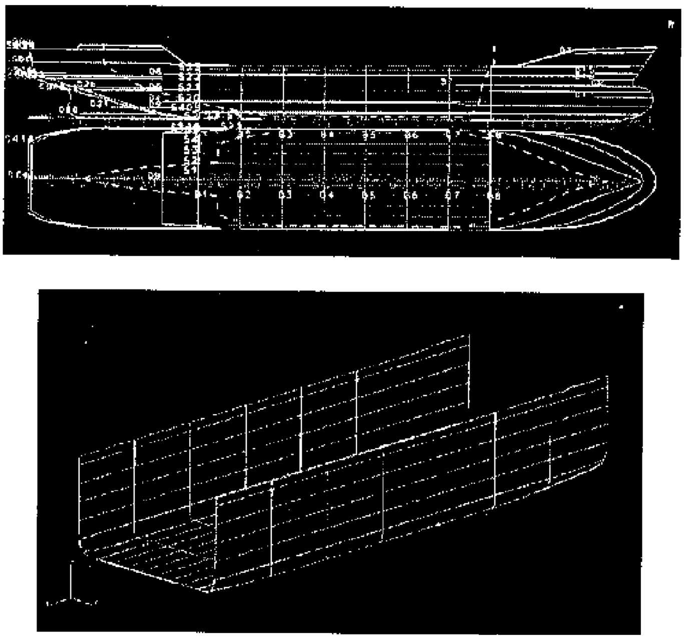


Figure 6. Panel and plate definition. Chemical carrier

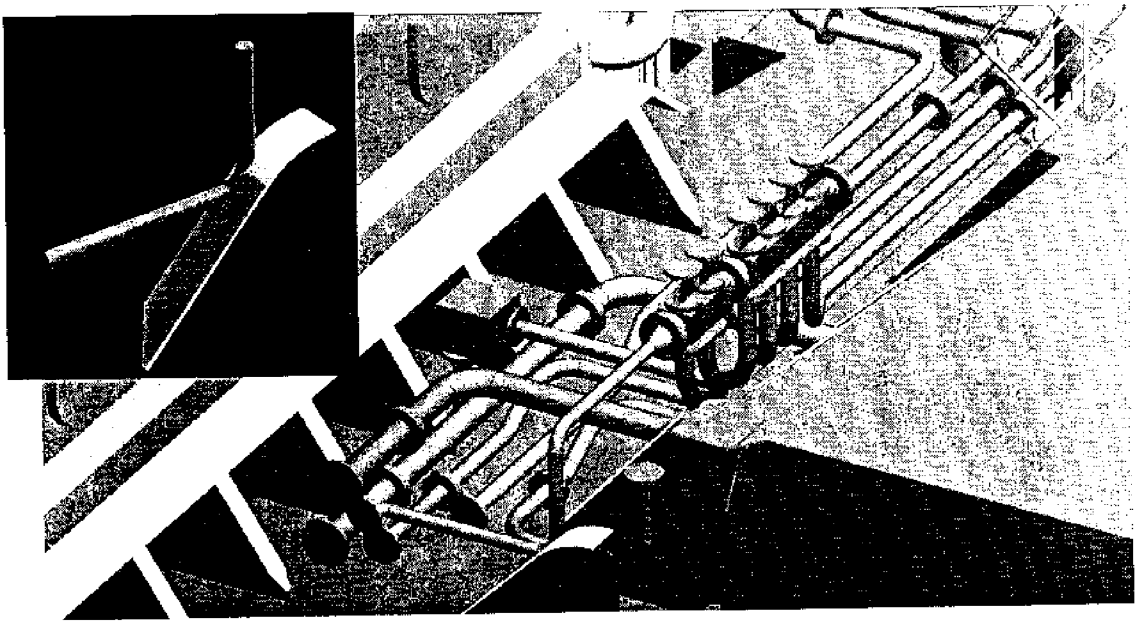


Fig. 7 - First piping layout alternative. Detail of interference

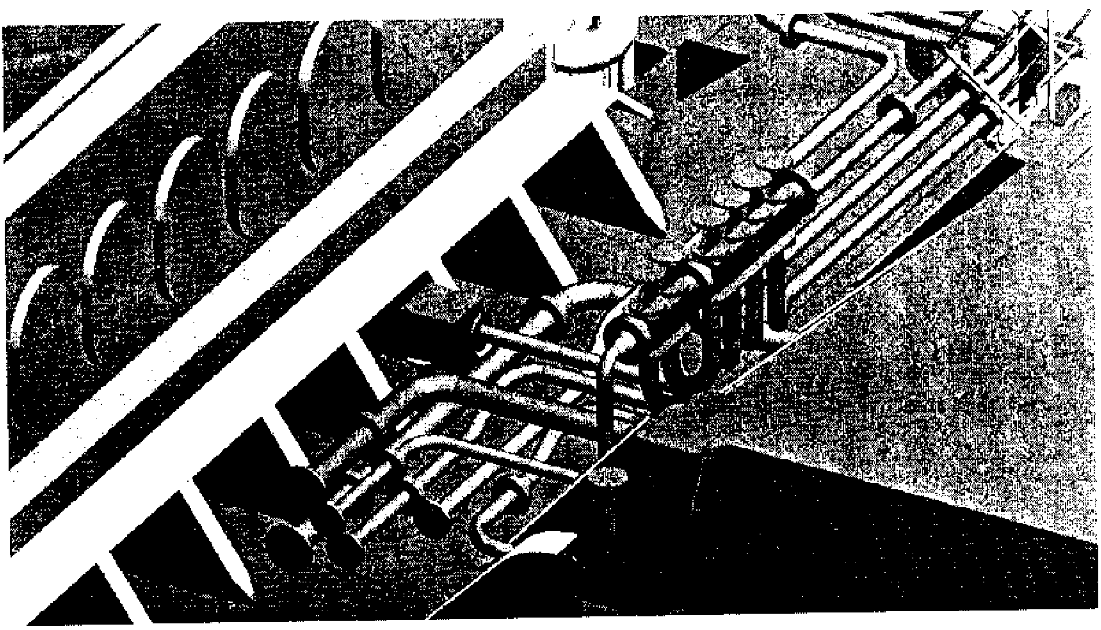


Figure 8 - A different piping layout alternative

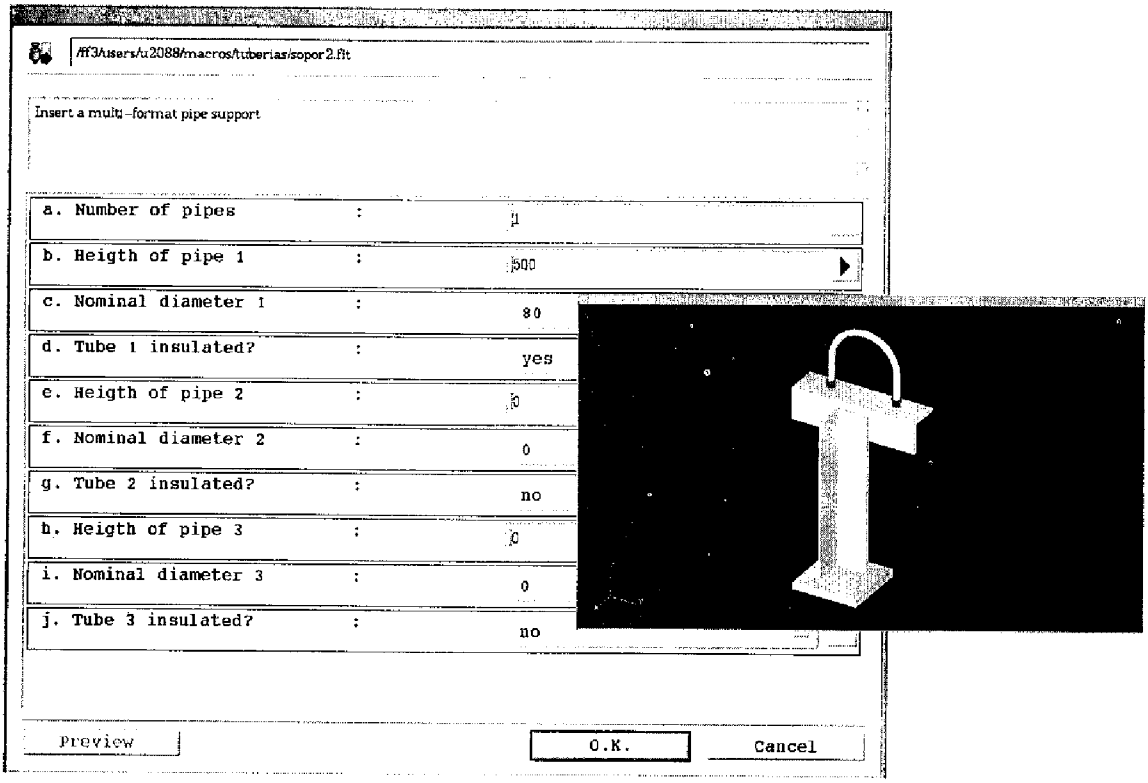


Figure 9 - Piping support defined with a macro

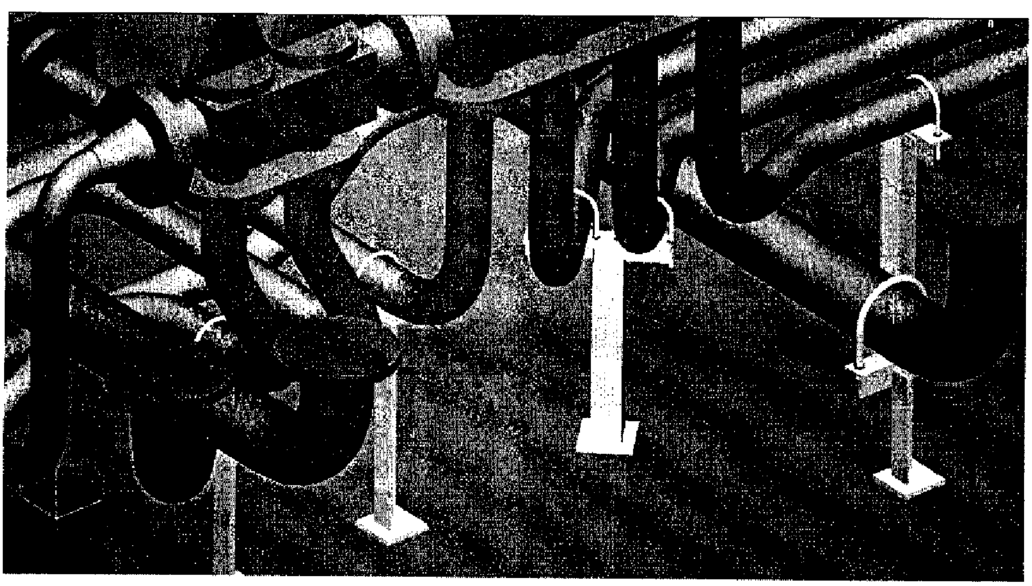


Figure 10 - Positioning of supports

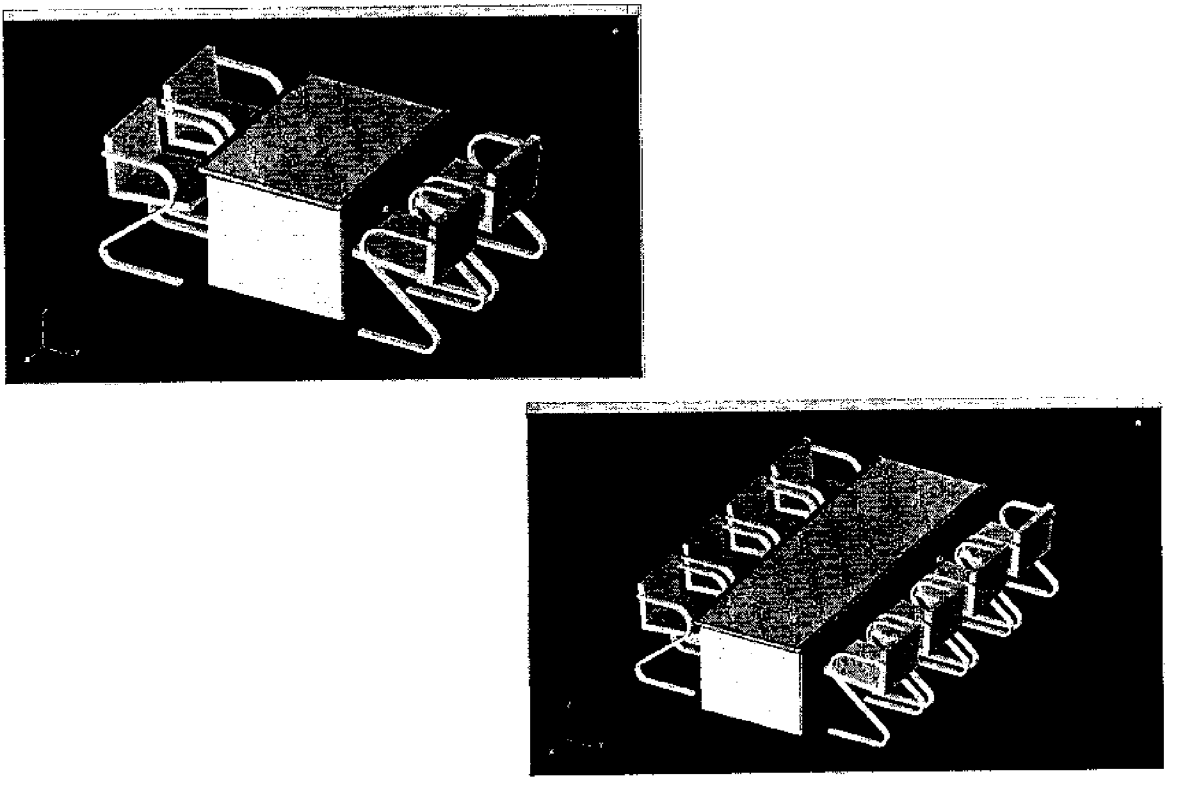


Figure 11 - Definition of accommodation elements by means of macros

COMPACT PRODUCT MODELS AS A VEHICLE FOR EFFICIENT DESIGN MANAGEMENT

Dr. Mats Westenius, Kockums Computer Systems AB, Malmö, SWEDEN

Abstract

A Ship Product Model is constituted by complex and integrated sets of information. To be used productively in the Shipbuilding Process Management, it must be concurrently accessible to all relevant parts of the organization, throughout the design, manufacturing and assembly phases. Furthermore, it must contain numerous types of data whereof the resulting geometry information is an essential but limited part.

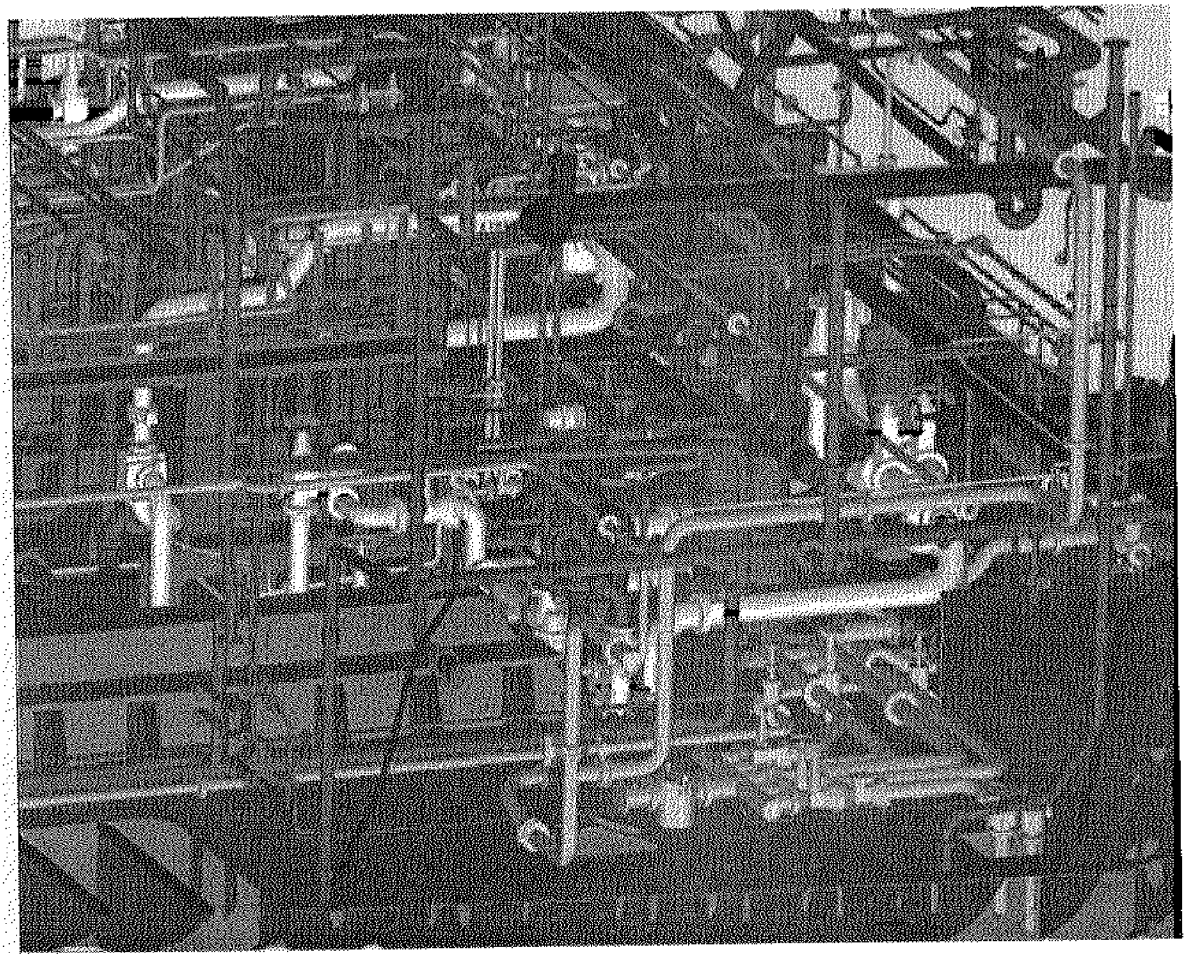


Figure 1. Part of a typical arrangement with several related objects in TRIBON. Courtesy of van der Giessen-de Noord.

The time aspects of the development of the Ship Product Model defines a clear need for efficient, rapid and continuos, in context, design review. During such a review, the navigation in the Ship Product Model can be performed through the structural information inherent in the model. Another way is through the 3D graphical representation itself, irrespective of whether the emphasis of the review is on the geometrical implementation of the individual parts or not.

An efficient design review requires direct access to the, at the specified time, complete Ship Product Model with an adequate response time. Traditional approaches have been focussing on the display of the 3D graphical representation of the large assembly often involving pre-processing of the original Ship Product Model information into formats optimized for graphics. However, to maintain the direct contact with the Ship Product Model, the graphics information must be generated completely transparently and with high speed, directly from the model and thereafter be treated only as cached information.

Advances in computer technologies have been tremendous regarding CPU power and graphics performance as well as the performance of local networks. However, for complete Ship Product Models, the total bandwidth will continue to be a bottleneck, especially when the Internet is considered. Although the individual Product Model objects must contain complete information they still need to be very compact to reduce the total data transfer. This can be achieved by avoiding redundant geometry data in the Ship Product Model objects, keeping only enough for an able client to create the complete, full-blown 3D geometrical information on demand.

This paper discusses a method for Design Management, using the TRIBON shipbuilding system and its implementation of the Ship Product Model.

Product Model Contents

Although the Product Model concept itself sets no limits, the primary purpose of the Ship Product Model, so far, has been to be one of the main sources of information during the definition, construction and assembly phases of a shipbuilding project [1]. Typical examples of its use are e.g. cross-discipline design planning, space management, materials requirements definition, and as a source for all parts manufacturing and assembly information. Accordingly, the content of the Ship Product Model is very often sub-divided into the corresponding areas.

Structural Information

This is the core of the Ship Product Model, which results in geometrical items that can be modeled and visualized on a computer screen. The structural information includes the shell with its plating and seam definitions, all the decks, bulkheads, web-frames, girders and stringers sub-divided into design panels with stiffening, brackets, flanges, cut-outs, notches, holes and other features. A very important fact is that it contains the topological and associative information which e.g. through the definitions of typed connections manifests itself in the instantiation of specific profile end-cuts and bevels.

In order to make a clear connection between the virtual world of the Ship Product Model and the real world in which the ship is to be built, all physical features like steel grade, profile types etc. have to be accounted for.

Finally, the hierarchical sub-division of the structural information into design units is an integral part of the Ship Product Model. At the top level of the Ship Product Model, the different disciplines form individual groups in which further sub-divisions, often in the form: geographical area – functional unit – parts, are defined (Figure 2).

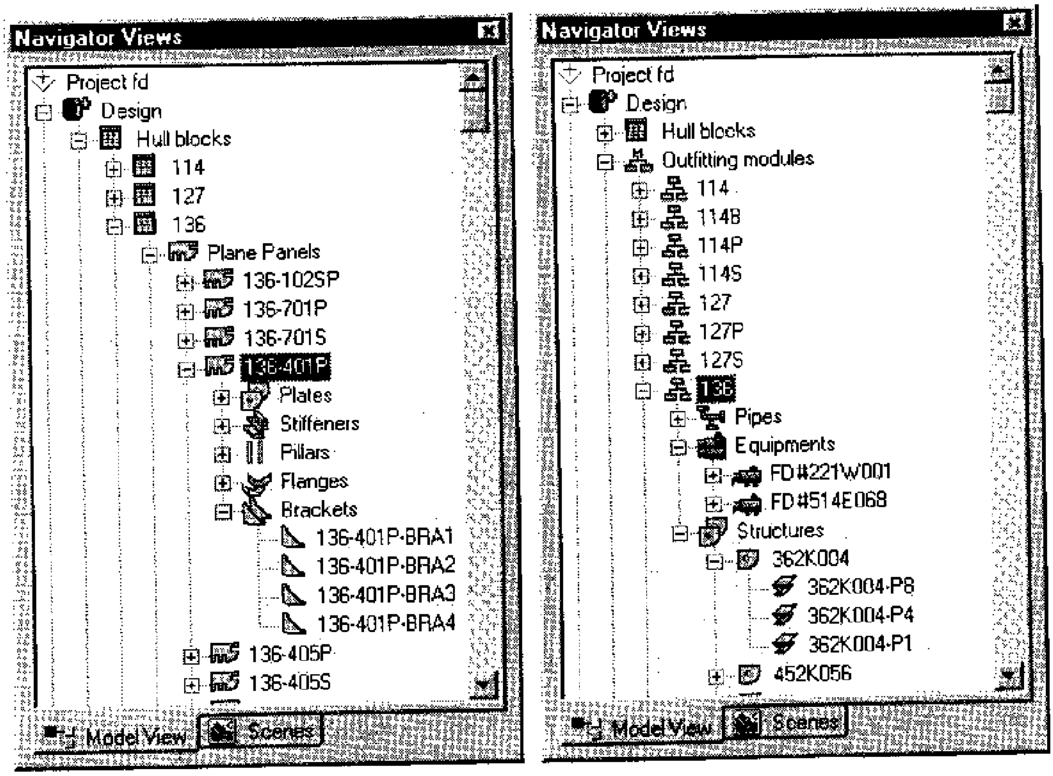


Figure 2. Typical examples of hierarchies in the TRIBON Hull and Outfitting disciplines.

Naturally, all outfitting items like pipes, pipe-hangers, valves, cables, cable-trays, pumps and other equipment as well as their topological and associative connections to each other and to the hull structural members needs to be included in the total picture.

Manufacturing Information

The manufacturing information is, of course, based on the previously discussed structural information. However, the focus is on production or manufacturing properties. This means that the geometry of the parts has been compensated for weld shrinkage and excess material, bevels and surface treatment have been defined etc. Curved plates have been developed and bending templates and jigs have been defined accordingly. Plate and profile nesting results in marking and cutting information for cutting-machines, panel lines and robots as well as profile sketches etc. Again, the corresponding information for the outfitting disciplines should not be forgotten. Production dates, status and workshop routing are just a few examples of additional data.

Assembly Information

The assembly phase defines the cross-discipline information needed to support the complete erection of the ship. Hull and outfitting parts are collected into sub-assemblies, which in turn are arranged into larger assemblies. Typical examples of information accessed through these hierarchies are assembly drawings and parts-lists, calculations of weights and centers of gravity and assembly sequences. Work content calculations, welding definitions and welding sequences, with corresponding information for assembly welding robots are other examples.

Product Model Access Scenarios

All in all, the Ship Product Model represents an enormous source of information, which is accessed for different purposes in different situations. From an application development's point of view, it has to be realized that these different tasks define quite different access scenarios regarding client/server capabilities and abilities as well as requirements on data transfer rates and total response time. Although a traditional set-up is concentrated on communication between computers all connected in a high-speed network, the cooperation with design agents and the emerging of virtual, distributed shipyards, the Internet as an alternative information carrier has to be taken into account.

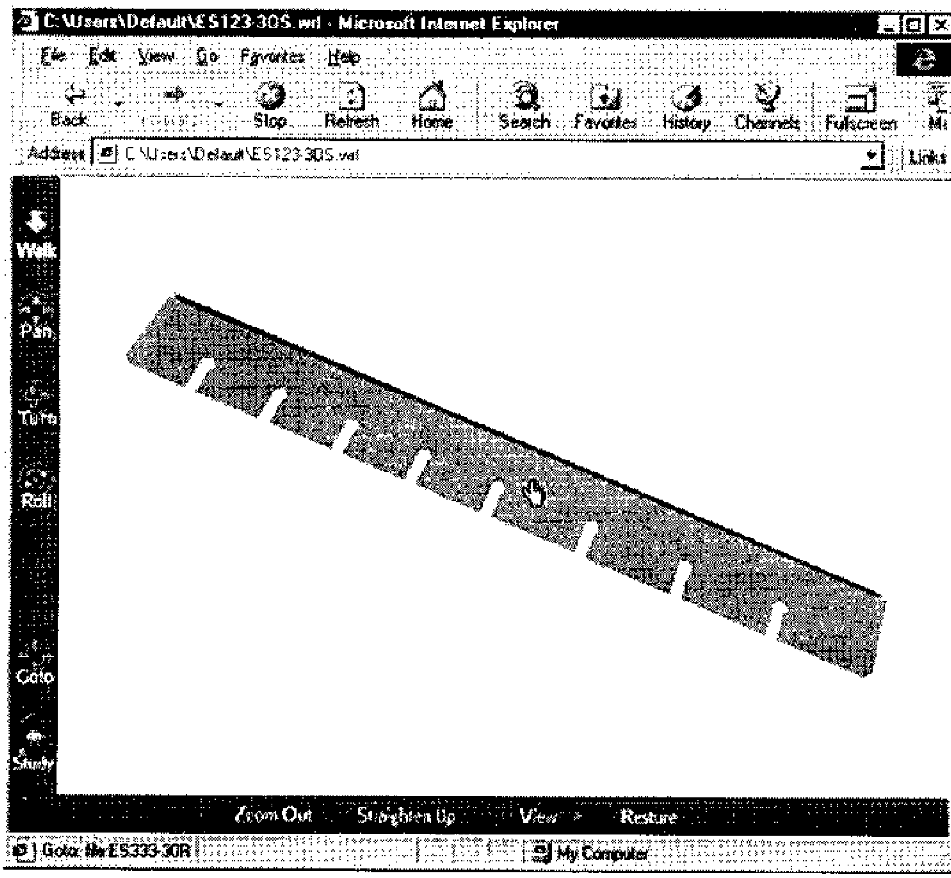


Figure 3. Small panel from TRIBON Hull viewed as a VRML object.

Single Part Access

Since most modeling applications no longer are to be considered as part programming tools, but rather define larger structures from which the parts are automatically derived, single part access is insufficient in the Ship Product Model definition phase. However, in the manufacturing phase, or more specifically, in the workshop it may still be productive to access individual parts to examine their geometrical features and to extract manufacturing information.

The total data-set for one single part is fairly limited which makes it possible to transform its geometrical definition into formats e.g. like the Virtual Reality Modeling Language (VRML) which can be handled by generalized viewers. A thin client application with very limited functionality e.g. like a Web browser with an embedded VRML viewer could, in this case, very well serve its purpose (Figure 3).

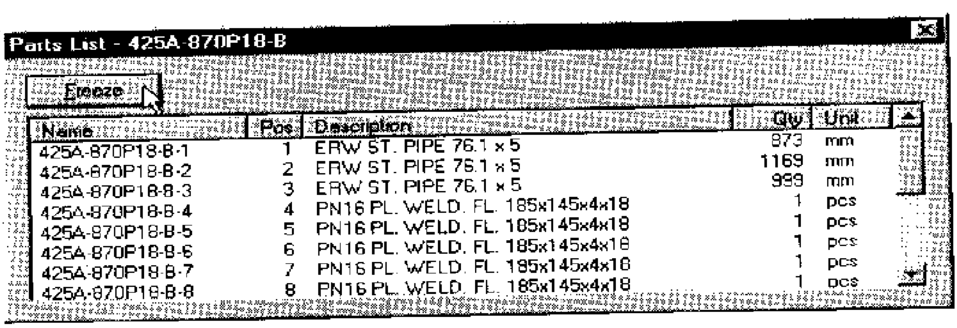
In Context Modeling

During the Ship Product Model definition phase, the context in which the modeling is done sets the rules and requirements for the tools [2]. In other words, the structures being modeled depend heavily on the surroundings i.e. other structures already in place. Highly specialized tools may even require information not directly visible in the current context in order to fulfil the complete set of modeling rules. One example of such a case would be a rule that prevents a pipe from being modeled in a fashion, which makes it impossible to manufacture in the defined pipe-bending machine in the workshop.

In this case, the clients need to have the ability to interpret and define Ship Product Model information according to the rules and requirements set by the current context. The size of the context may vary significantly but the data sets are generally much larger than in the case of single part access. The number of designers, concurrently working on the same project may be several hundreds, which means that the total workload should be distributed over the clients rather than concentrated to the server. The demand for good performance and short response times is high which requires the data sets to be slim in order not to bog down the server. This scenario favors a set-up with a thin and fast server and thick and able clients.

Specified Product Model Queries

This is a type of access that might include very large portions of the Ship Product Model but for which the result still represents a limited set of data. Typical examples could be to summarize the total amount of steel with a specified material quality, generate a parts-list for a named sub-assembly.



Name	Pos	Description	Qty	Unit
425A-870P18-B-1	1	ERW ST. PIPE 76.1 x 5	873	mm
425A-870P18-B-2	2	ERW ST. PIPE 76.1 x 5	1169	mm
425A-870P18-B-3	3	ERW ST. PIPE 76.1 x 5	999	mm
425A-870P18-B-4	4	PN16 PL. WELD. FL. 185x145x4x18	1	pcs
425A-870P18-B-5	5	PN16 PL. WELD. FL. 185x145x4x18	1	pcs
425A-870P18-B-6	6	PN16 PL. WELD. FL. 185x145x4x18	1	pcs
425A-870P18-B-7	7	PN16 PL. WELD. FL. 185x145x4x18	1	pcs
425A-870P18-B-8	8	PN16 PL. WELD. FL. 185x145x4x18	1	pcs

Figure 4. Parts list example.

Typically, this sort of access could be a simple data extraction or involve more or less computing on the server side, but the result is always in the form of listed textual information. The requirements on the client are simply that it should be able to define the query and to handle the resulting list of information. A fairly thin client working with an able server would be the set-up of choice.

Overall and General Access

In a situation where the overall features of the complete Ship Product Model are to be examined, there are no obvious restrictions in the access scenario that might reduce the requirements. This means that the set-up has to be tuned to handle the transfer of massive amounts of Ship Product Model information, both geometrical and non-geometrical and with adequate response times.

Having in mind that one of the major remaining bottlenecks in a computer system environment is the total available bandwidth, it becomes very important to minimize the total amount of data needed for each of the Ship Product Model part definitions. When this is pulled to its extreme, the trade-off is that the client must even be able to create geometrical representations on the fly from topological and associative information.

This is truly a scenario where the optimum set-up is with a thin and fast server and a thick, very able client.

Compact Product Model

It is clear that the storing capacities of modern computer discs and databases are not a limiting factor for a Ship Product Model. However, as seen from the last access scenario, reasons still remain to keep the Ship Product Model size to a minimum.

The obvious way is to avoid redundant data but the data that can be considered as redundant is largely defined by the ability of the client. For a physical item the design intents and the non-geometrical attributes have to be stored. However, its geometrical representation may be generated by the client application from the design intents and the defined set of applicable design rules.

As an example, endpoint associations to connecting objects and the profile type sufficiently defines the main geometry of a stiffener. The connection types and their associated rules define the geometry of the end-cuts and cut-out types for intersecting stiffeners define the inner features of the stiffener geometry. Another example could be the geometry of a pipe, which is given by its centerline definition and references to the components from which the pipe is to be built.

Another important issue is to have an appropriate container object definition. Too small objects result in to many database accesses where each round-trip consumes overhead time. However, if the container objects are too big, the ratio between requested data and the total amount of data transferred becomes unfavorably small. Recommendations for applications in general are hard to give since this depends significantly on the implementation client/server configuration but for the TRIBON Hull application the panel object has turned out to be very well balanced. Such an object define a structural element comprising one to a few plates with holes, flanges, cut-outs, clips and notches, the complete stiffening and all associated brackets. The size of a panel object is typically 5-10 kilobytes and very rarely more than 100 kilobytes. Figure 5 shows one panel with 141 parts and a smaller one with 13 parts.

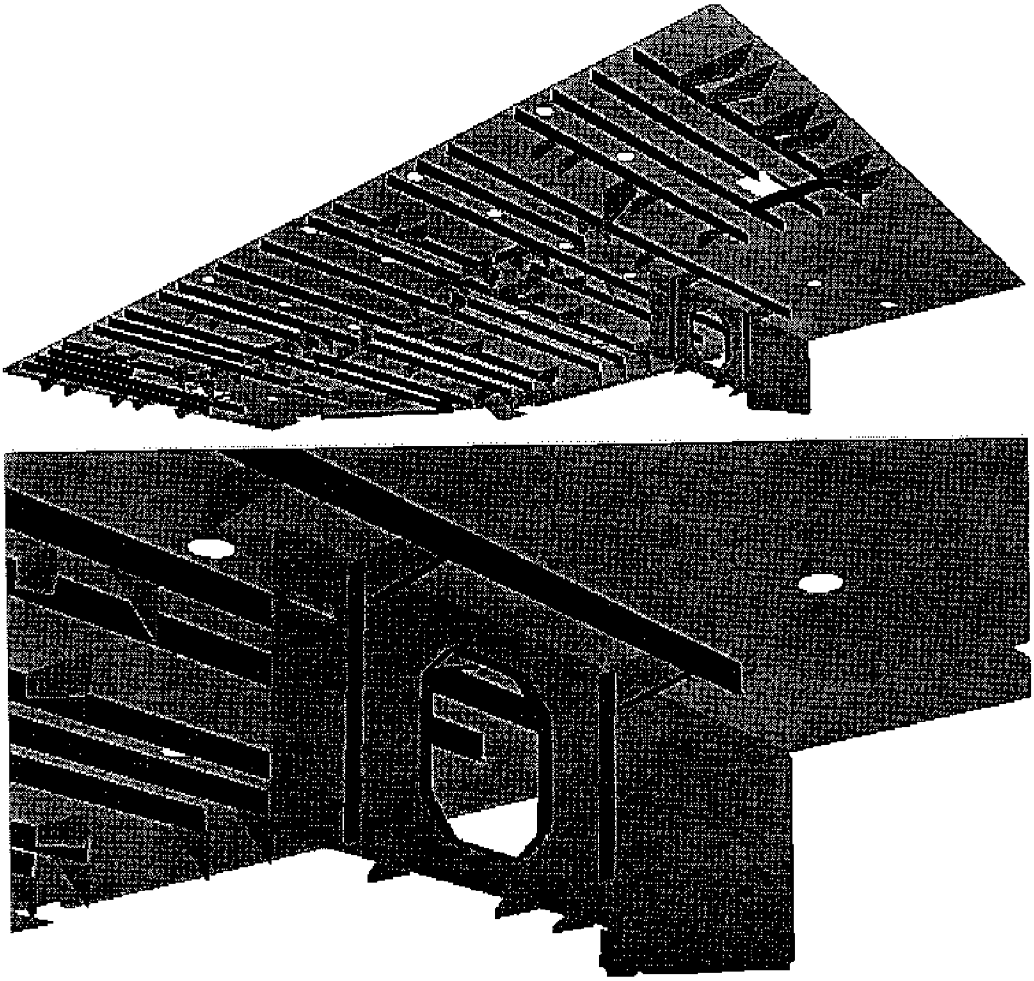


Figure 5. Two TRIBON Hull panels, 66 and 4 Kbytes big (top view) and blow-up of the 4 Kbytes big panel (bottom view). Courtesy of van der Giessen-de Noord.

Design Management

It may seem that very little has been said about Design Management in this paper so far. However, Design Management, as defined here, means the ability to access all parts of the Ship Product Model in an overall way and thus includes most of what has been described above. The task is to monitor the evolution of the complete design, to follow the achievements and to resolve design conflicts, both from the overall perspective and in minute detail. The context is thus neither restricted by discipline nor by zone but it is likely that it is bigger than for any other task, even in the normal case.

An essential part is the navigational possibilities in all Ship Product Model structures both graphically and through defined names and identifiers. This means that, in the Design Management case, the first access to the Ship Product Model always has to obtain the complete information about the hierarchical breakdown structures, if for no other reason, then at least for navigational purposes.

A typical working session could look like the following.

- Start the application and connect to a project i.e. choose Design to be examined.
- Select hierarchical breakdown for the navigation.
- Select a number of items, on any level in the hierarchy, for display.

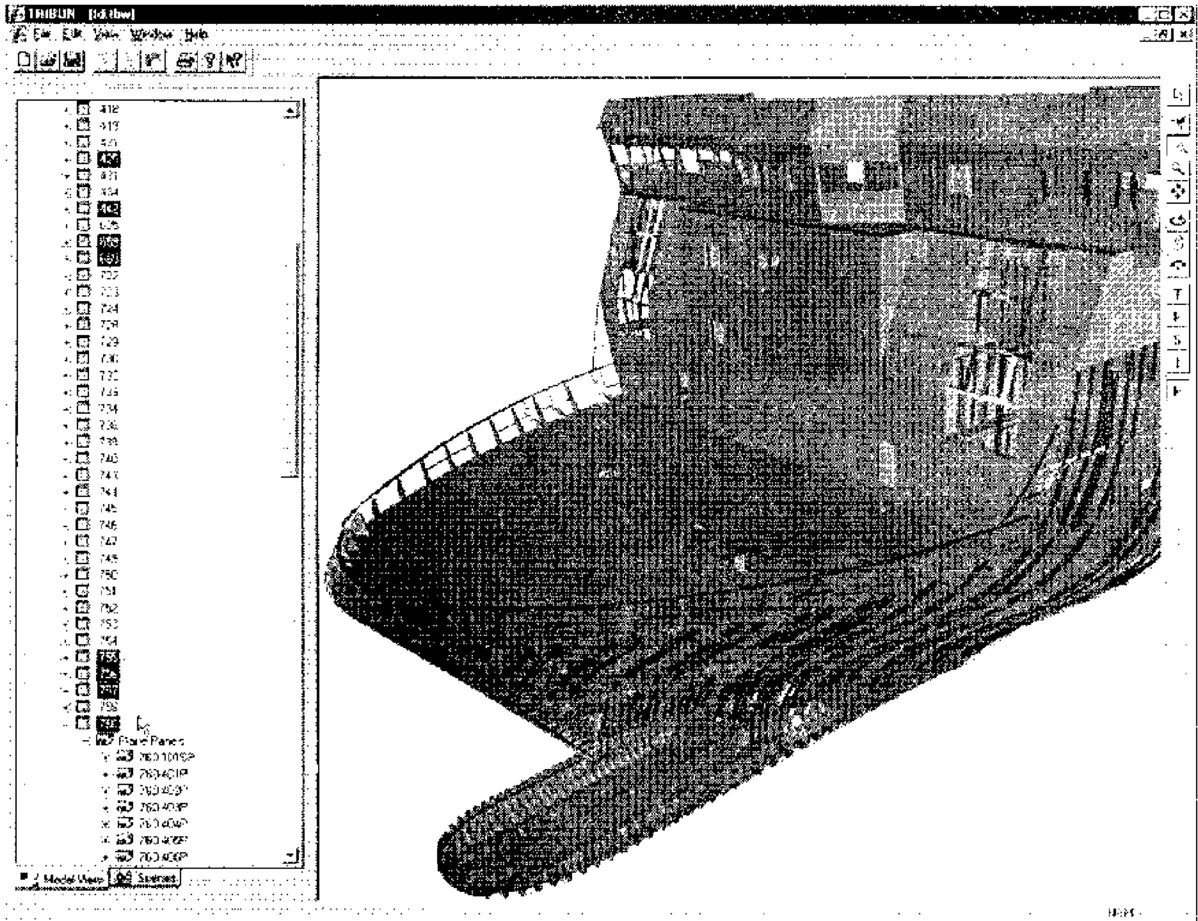


Figure 6. Overview of the fore end of a ship in TRIBON. Courtesy of van der Giessen-de Noord.

- Navigate in the 3D graphical representation or in one of the tree views, to the item in question.
- Request information about the specified item, which has been selected in the graphical or the tree view.

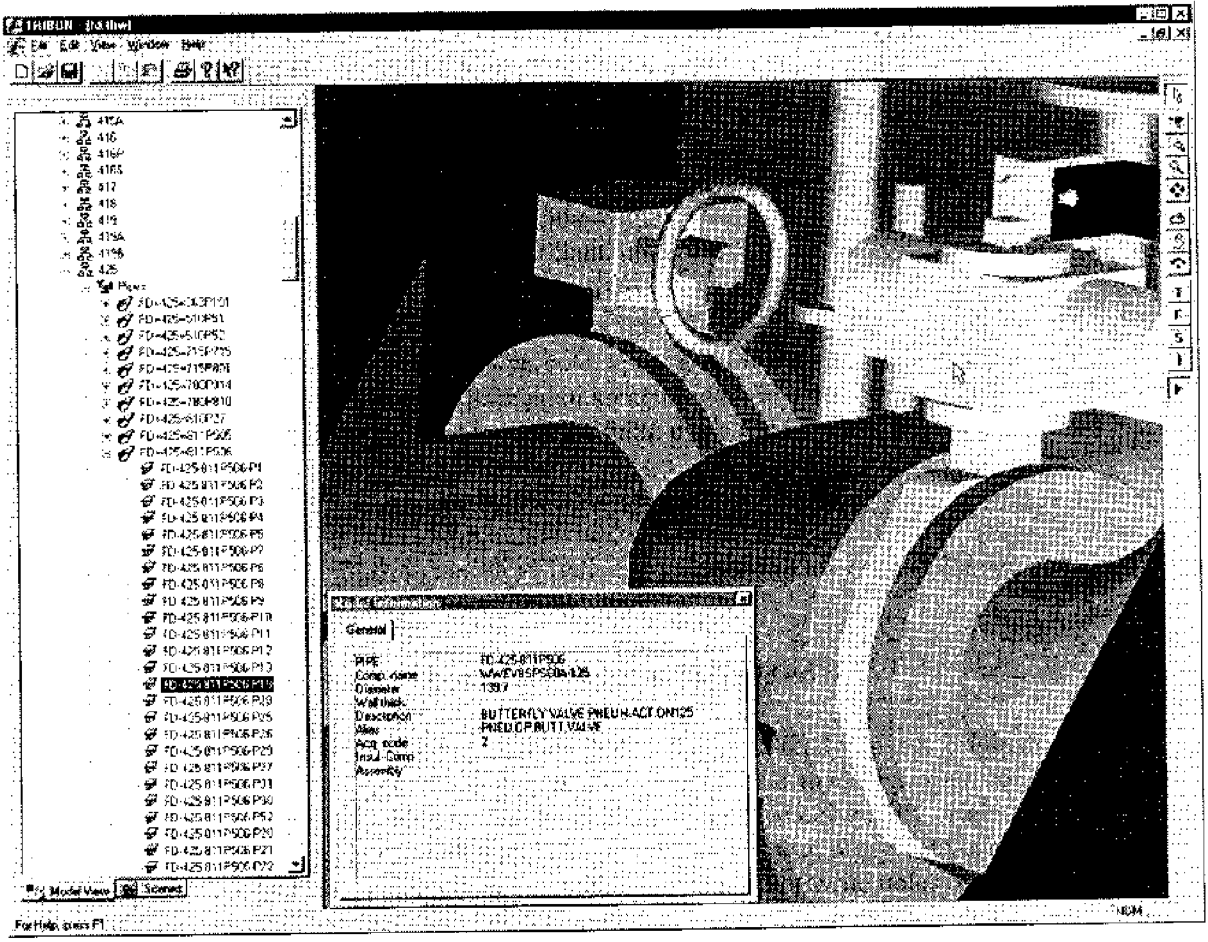


Figure 7. Zoom-in on a butterfly valve in a TRIBON Ship Product Model. Courtesy of van der Giessen-de Noord.

Conclusion

For a developer of shipbuilding applications, Design Management represents a real challenge. It has to connect directly to the Ship Product Model and not through pre-processed, optimized graphics data, otherwise it will be impossible to show the current state of the Ship Product Model at each instance. The client must be able to develop the individual parts and structures from the design intents and to create and handle the facetted representation of very large assemblies with adequate graphics performance [3]. All design data must be available for examination through a mouse-click. Finally it must be realized that the Design Management may take place in a location geographically distant from where the actual design work is done. This means that it should be not only technically possible, but also feasible to connect to Ship Product Models over low bandwidth networks like dial-up connections or the Internet.

The requirements are hard but with a consciously developed basic software architecture regarding a compact Ship Product Model implementation and taking advantage of the rapid evolution of computer speed and graphics performance, there is no doubt, it is possible.

References

1. Tanabe, T. et al. (1997), Introduction and Speedy Practical Development of an Integrated 3D CAD/CAM System for Hull and Outfitting, Proc. of the 9:th International conference ICCAS '97, Yokohama.
2. Nilsson, P-O. (1998), In Context Modeling, KCS Newsletter No. 36, December 1998.
3. Nilsson, P-O. (1998), High Performance Large Assembly Techniques, KCS Newsletter No 35, September 1998.

APPLICATION OF COMPUTER TECHNOLOGY AT MHI SHIPBUILDING, LLC

Sotiris G. Emmanuel, MHI Shipbuilding, LLC, Quincy, MA
Ki-Hee Kim, MHI Shipbuilding, LLC, Quincy, MA

1. Introduction

MHI Shipbuilding (MHI) was formed for the purpose of reactivating the Fore River Quincy Shipyard (inactive since 1987). The objective was to develop a workforce and shipbuilding infrastructure capable of competing to build commercial vessels on the world market and to do so without subsidies. The infrastructure for adequate dry docks, cranes, outfitting piers and land existed when it was purchased. However, the existing facilities and implied processes for producing steel and pipe, were outdated and incapable of achieving world market cost levels. All equipment in these areas was disposed of, allowing a "clean sheet of paper" approach to the design of process, facility and computer applications.

However, in order to achieve a leadership position for producing commercial vessels in the United States, MHI Shipbuilding has adopted highly advanced computer aided design and computer integrated production system.

MHI also has equipped new state-of-art production equipment and facilities so that it will be capable of providing the best quality vessels to its customers and establish an undisputed competitive position in the international marketplace with skilled and experienced workers.

We knew that other advanced shipyards have experienced many difficulties in the beginning of the yard operation even though they had experienced workers. However, having no experience and no skilled workers in building commercial vessels, MHI must avoid the trial-and-error situations as other advanced shipyards have experienced when they began operations. In order to minimize the risk and problems, it is better to maintain close relationship and have support from Halla Engineering & Heavy Industries, who has previous been involved with constructing various kinds of vessels, has similar modernized shipbuilding equipment and facilities as MHI.

Furthermore, MHI has a plan to dispatch a core group of MHI workers including foreman to Halla to obtain hand-on-experience training for the various fields of commercial shipbuilding and constructing sections of its own vessels, in order to have a successful business and yard operating in the future.

This paper will describe MHI's company history, shipyard modernization tasks and goals, layout of the shipyard and shipbuilding process as a new born shipyard. Further it will show application of Tribon and implementation of automated Robotic Production System.

2. ***Company History (Fore River Shipyard)***

1) 1884 ~ 1914: Fore River Shipyard
   - First all-steel and Navy ships in 1900
   - Submarines for Japan, UK, Canada and Spain

2) 1914 ~ 1964: Bethlehem Steel
   - First turbine drive battleship
   - 350 Navy ships, 80 submarines
   - Navy's first nuclear surface ship
   - Passenger liners, Tankers

3) 1964 ~ 1986: General Dynamics
   - Warships and nuclear submarines
   - Lash Ships
   - LNG Carriers
   - Maritime Prepositioning Ships
   - 1200 Ton Goliath Crane

4) 1986 ~ 1997: Mass Water Resources

5) 1997 ~ : MHI Shipbuilding, LLC
   - Shipyard modernization in 1998~1999
   - Shipbuilding started in 1999

3. ***Shipyard Modernization***

1) Modernization Tasks
   - Environmental cleanup
   - Facility cleanup and demolition
   - Build new modernized shops
   - Renovate old shops, dry docks, office, electric, water and gases
   - Equip new state-of-the-art production equipment and facilities
   - Automate / Robotize the production lines & process
   - Adopt highly advanced computer aid design and computer integrated production system

2) Overall Shipyard Modernization Goals
   - World-class shipbuilding facilities featuring
   - State-of-the-art production technology
   - High production efficiency modeling, most productive world class shipyard
   - Adopt Tribon 4.0 software for computer aid design and advanced integrated production system
   - Build ships faster, safer, better and lower cost than other U.S. shipyards

4. *Shipyard Layout*

Overview of shipyard is show on the Figure 1.

1) Concept of yard operation
   - Integration of design, production and MIS
   - Computerized Automation System for indoor production
   - Escalation of pre-outfitting
   - Flexible Dock Operation
   - Optimize the production process
   - Developing production design from well proven licensed design in Tribon, to suit MHI's production process

2) Production Facility
   - Total Area : 6,000,000 ft^2 (557,400 m^2)
   - Fabrication Shop : 600,000 ft^2 (55,740 m^2), capable of processing 4.5x20m plates
   - Pipe Shop : 200,000 ft^2 (18,580 m^2)
   - Five Dry-docks with size ranges 870ft ~ 950ft (265m ~ 290m) long and 132ft ~ 150ft (40.2m ~ 45.7m) wide
   - Goliath Crane : 1200 Tons

3) Shipbuilding Process: MHI adopt modernized shipbuilding process as shown on Figure 2.

   All steel received will be shot-blasted and coated with weldable primer. The fabrication process creates an assembly-line sequence complete steel blocks that are in turn blasted, painted and pre-outfitted. Completed blocks are then pre-staged for all additional pro-outfitting and partial or full system testing prior to erection. Blocks will then be moved into a pre-erection site to create "grand blocks" for subsequent Goliath crane lift into position.

5. *Relationship of MHI and Halla Engineering & Heavy Industries Ltd.*

5.1.1 Basic Agreement
   - Transfer of Design, Engineering, Production Experience and Technology for Commercial Shipbuilding
   - Fine-tune shipyard modernization
   - Partnership agreement for 5 years
   - Share know-how so Halla design is built properly, on time and within budget
   - On site work training in Korea and U.S. for key persons

5.1.2 Migration of Tribon database and drawings between Halla and MHI

Halla utilizes Digital Alpha Open VMS type hardware platform but MHI adopted HP-Unit version for the future. Thus, in order to prepare for Technology transfer between Halla and MHI shipbuilding, LLC, Halla provided Tribon software on HP hardware for migration of Tribon database and drawings.

The Procedure of database conversion

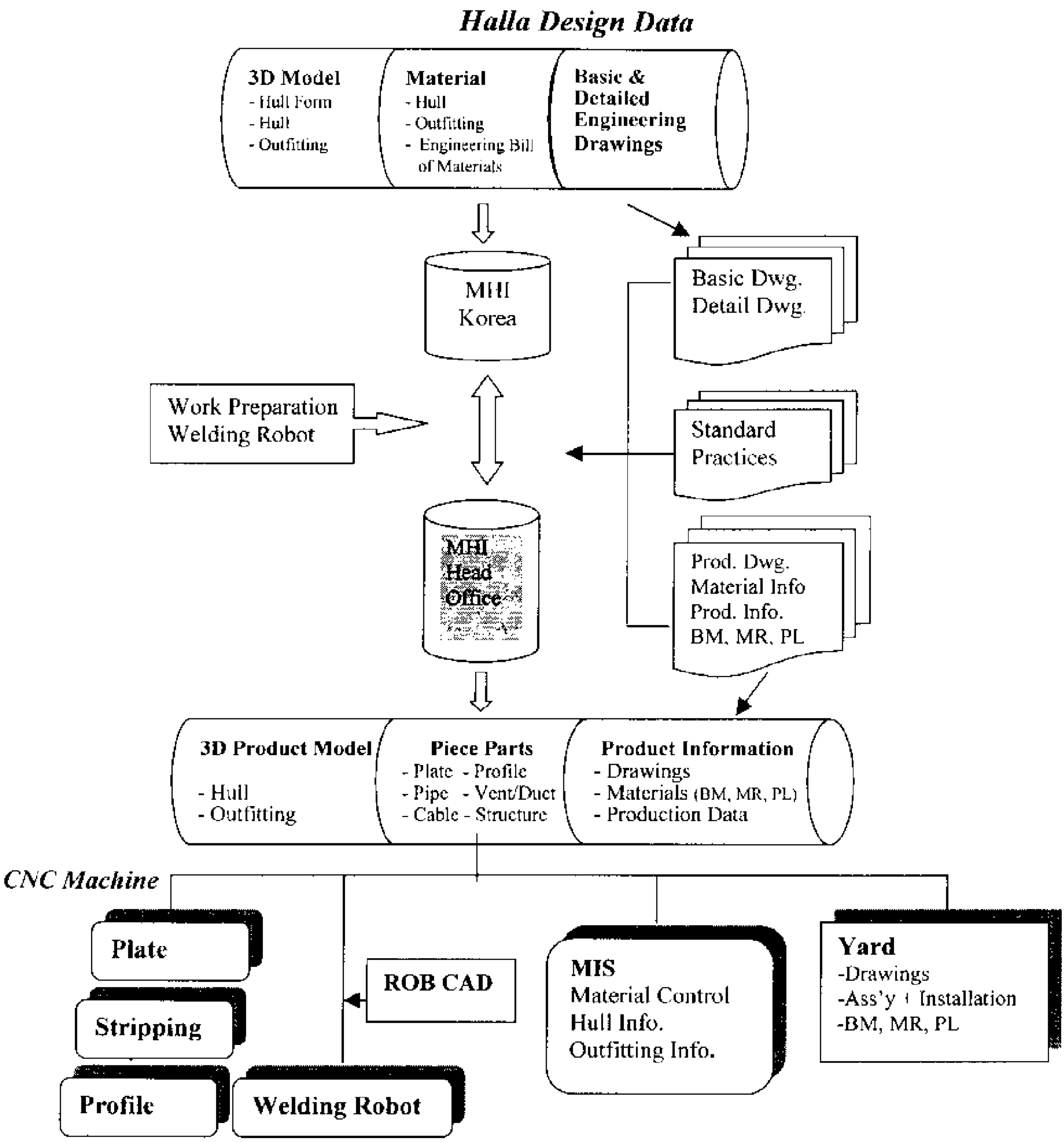
1) Tribon migration tool
   - Tribon V.30 & 4.0 SA009 for DEC Alpha and HP Unix Version

2) Migration working procedure of database
   - Conversion of index database to ASCII file with SA009 in DEC
   - ASCII file transfer from DEC to HP with ftp command
   - Conversion of ASCII file to index database with SA009 in HP

3) Procedure for conversion of sequential database
   - Conversion of sequential database to index database with DBUTIL in DEC
   - Conversion of index database to ASCII file with SA009 in DEC
   - Data file transfer to HP with ftp command
   - Conversion of ASCII file to index database with SA009 in HP
   - Conversion of index database to sequential database with DBUTIL in HP

5.3 Design Data Transfer Procedure

After migration of Halla's Tribon database and drawings from Digital Alpha Open VMS to HP Unix Version, all kinds of Halla's design information will be proceed for production as below:



Overview of MHI Shipyard

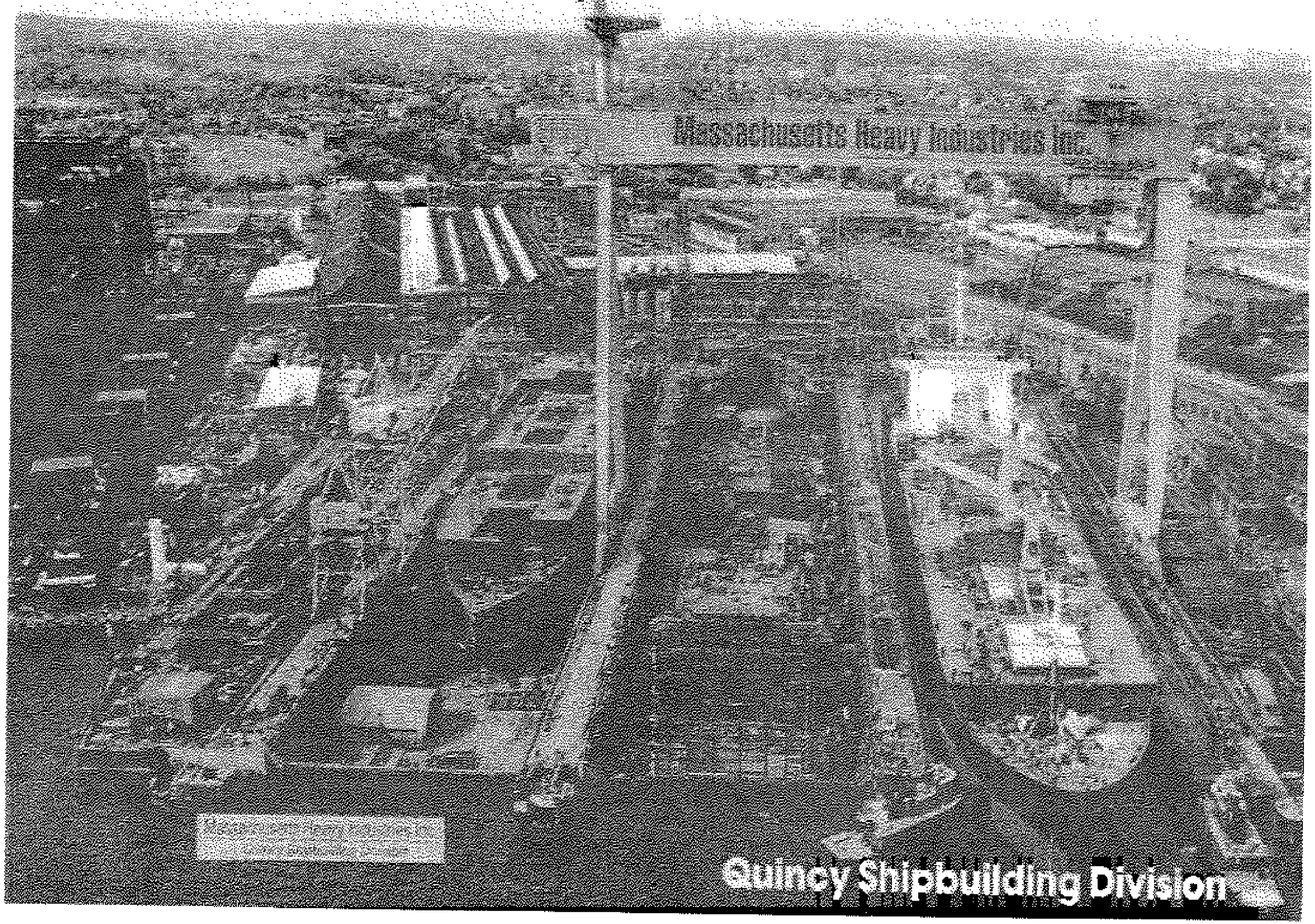
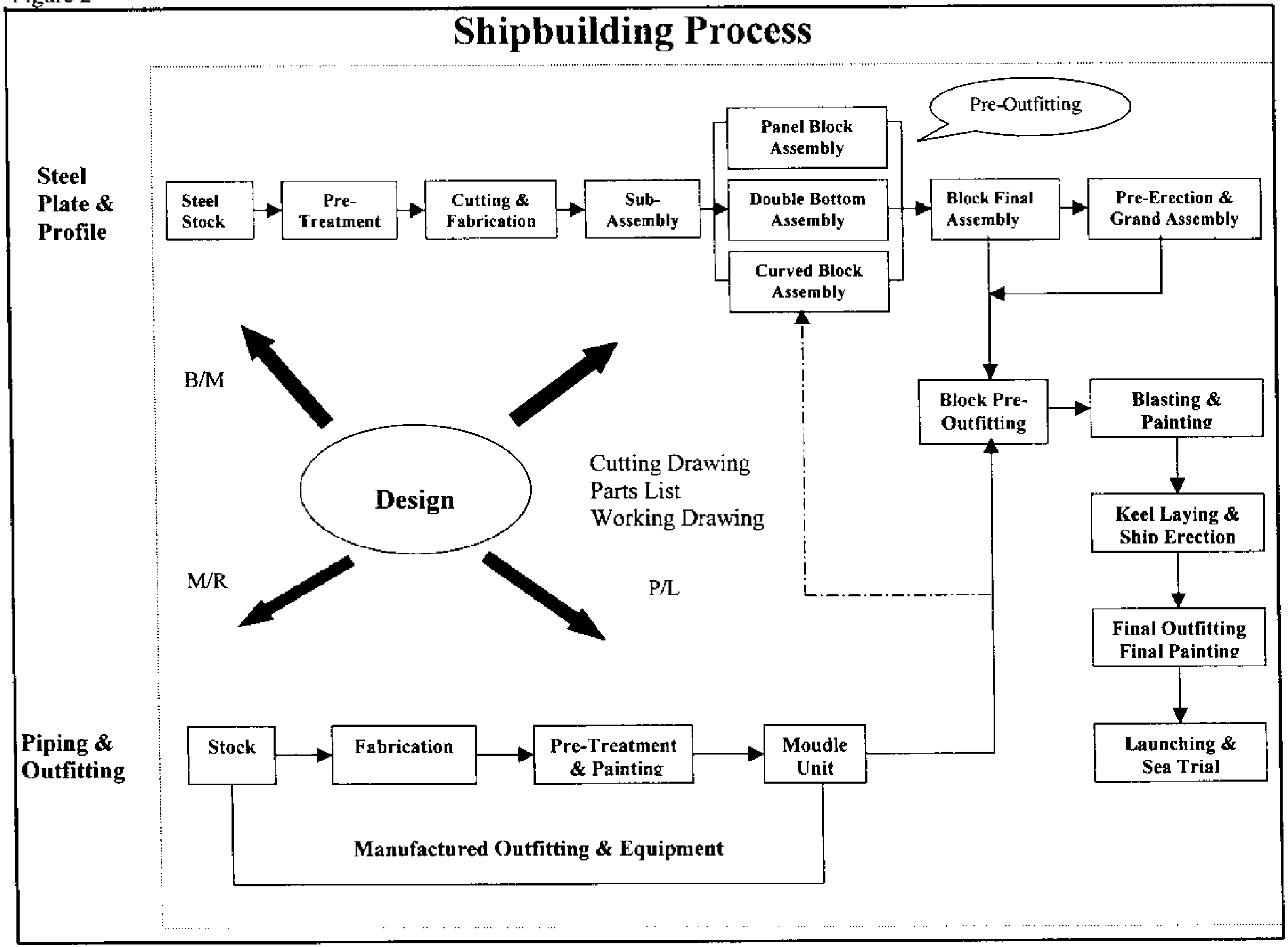


Figure 2


Shipbuilding Process
Steel Plate & Profile
Steel Stock
Pre-Treatment
Cutting & Fabrication
Sub-Assembly
Panel Block Assembly
Double Bottom Assembly
Curved Block Assembly
Pre-Outfitting
Block Final Assembly
Pre-Erection & Grand Assembly
Block Pre-Outfitting
Blasting & Painting
Keel Laying & Ship Erection
Final Outfitting Final Painting
Launching & Sea Trial
B/M
Design
M/R
Cutting Drawing
Parts List
Working Drawing
P/L
Piping & Outfitting
Stock
Fabrication
Pre-Treatment & Painting
Moudle Unit
Manufactured Outfitting & Equipment

6. Application of Tribon and Implementation of Automated Robotic Production

MHI adopted Tribon 4.0 software for computer aided design and advanced integrated production system.

A distinguishing feature of the Tribon system is the focus on the management of the underlying design and production information. All data including Halla's well proven design data can be stored in the Tribon Product Information Model, which consists of data objects, organized in structures natural to the shipbuilding process.

All kind of design information including logistics will be forwarded to production via MHI Network System as shown on Figure 3 and Figure 4 shows shipyard network configuration.

In order to achieve high productivity, MHI adopted faster and highly flexible-manufacturing technologies using automated robotic production system as below:

<u>Cutting Line</u>

- 4 sets of MG CNC plasma cutting machine with Beveler and marking torch
- 1 set of MG plasma and Oxy-fuel CNC cutting machine with triple torch Bevel Rotator
- 1 set of Koike Stripping CNC machine with 24 Oxy-fuel torch and plasma end cutter
- 1 IGM profile cutting Robot

<u>Forming Area</u>

- DNC control 2000 ton x 17m, Roll-Press, 1000 ton x 5m portal press, 400 ton framebender, 500 ton press and 300 ton bender

<u>Subassembly Line</u>

- 1 IGM Welding Robot with Roller Conveyor
- 1 IGM Welding Robot with Skid Conveyor

<u>Panel Line</u>

Ogden panel production line with conveyor system consists of:

- Magnet Bed with Input and Output conveyor
- Seam Welder
- Stiffener Fitter and Transfer car
- Stiffener Welder
- Web Fitting Gantry
- Web Service Gantry and Load-out Beam

<u>Double Bottom Line</u>

- Slitting system for threading webs through stiffener
- 2 sets of ABB Egg-box Robot Welding Gantry
- Load Out Station

<u>Final Block Assembly</u>

- 200 ton hydraulic lifting train transport system
- Load Out Station

Fig.3 TOTAL NETWORKSYSTEM

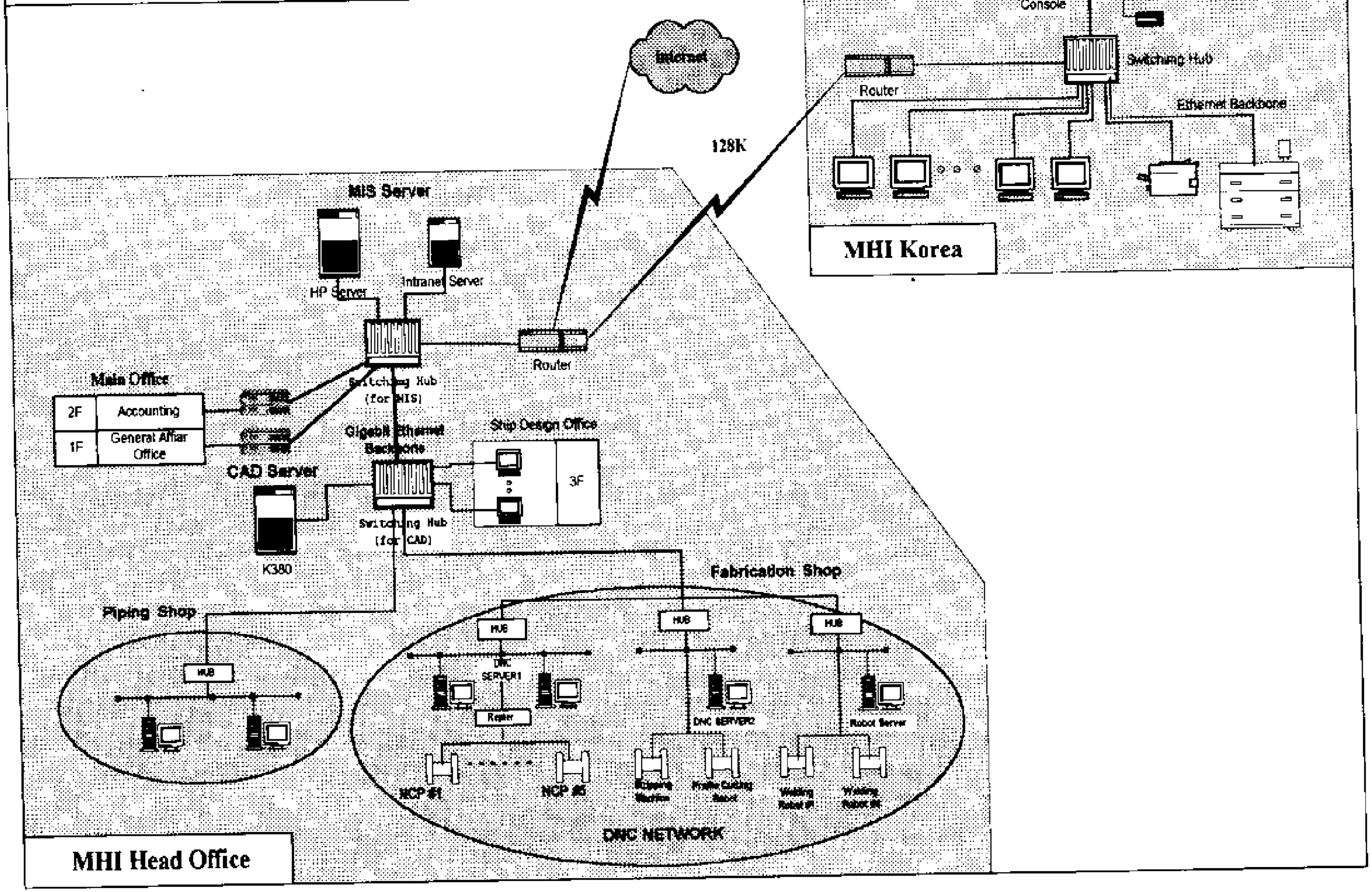
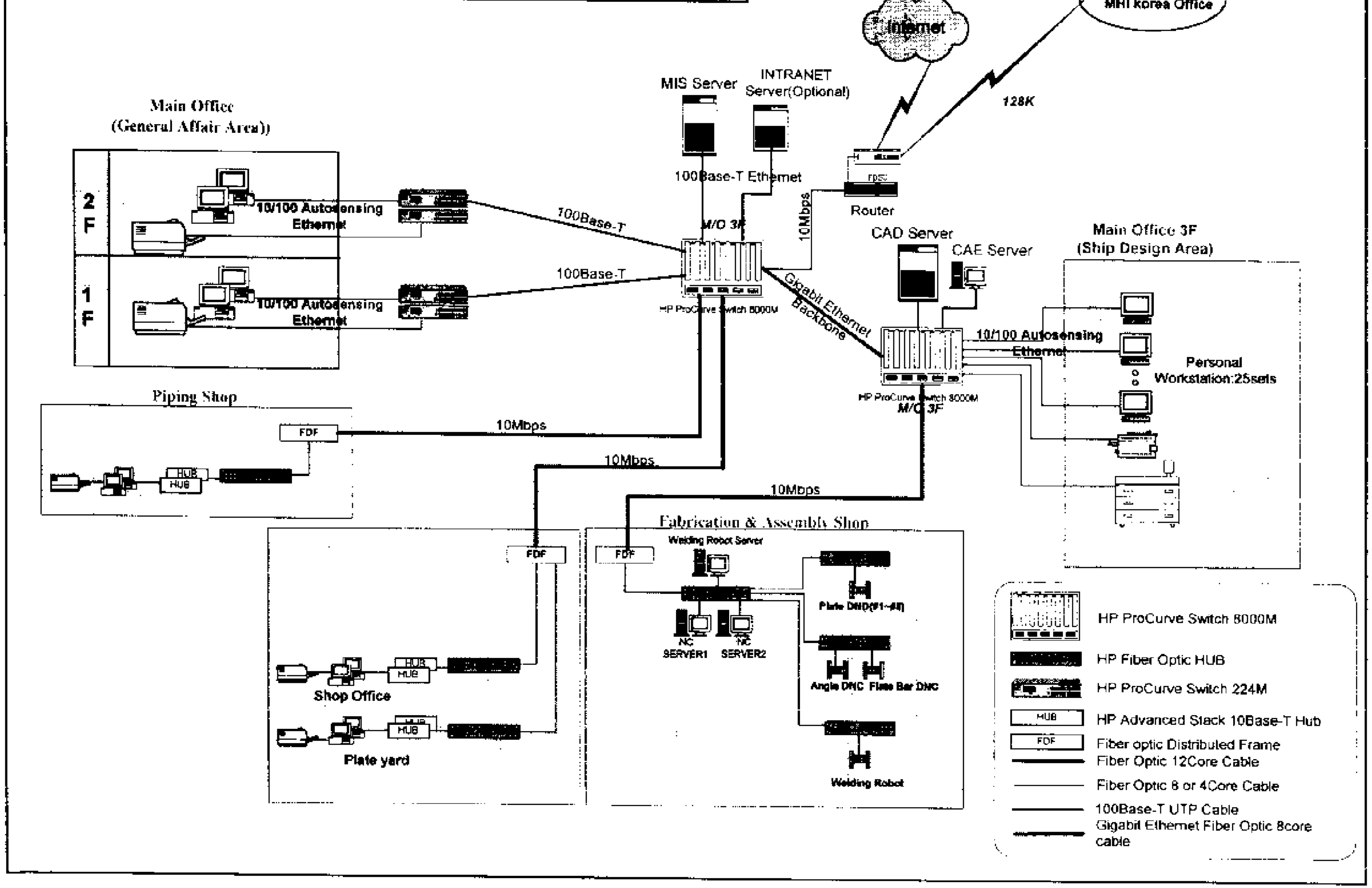
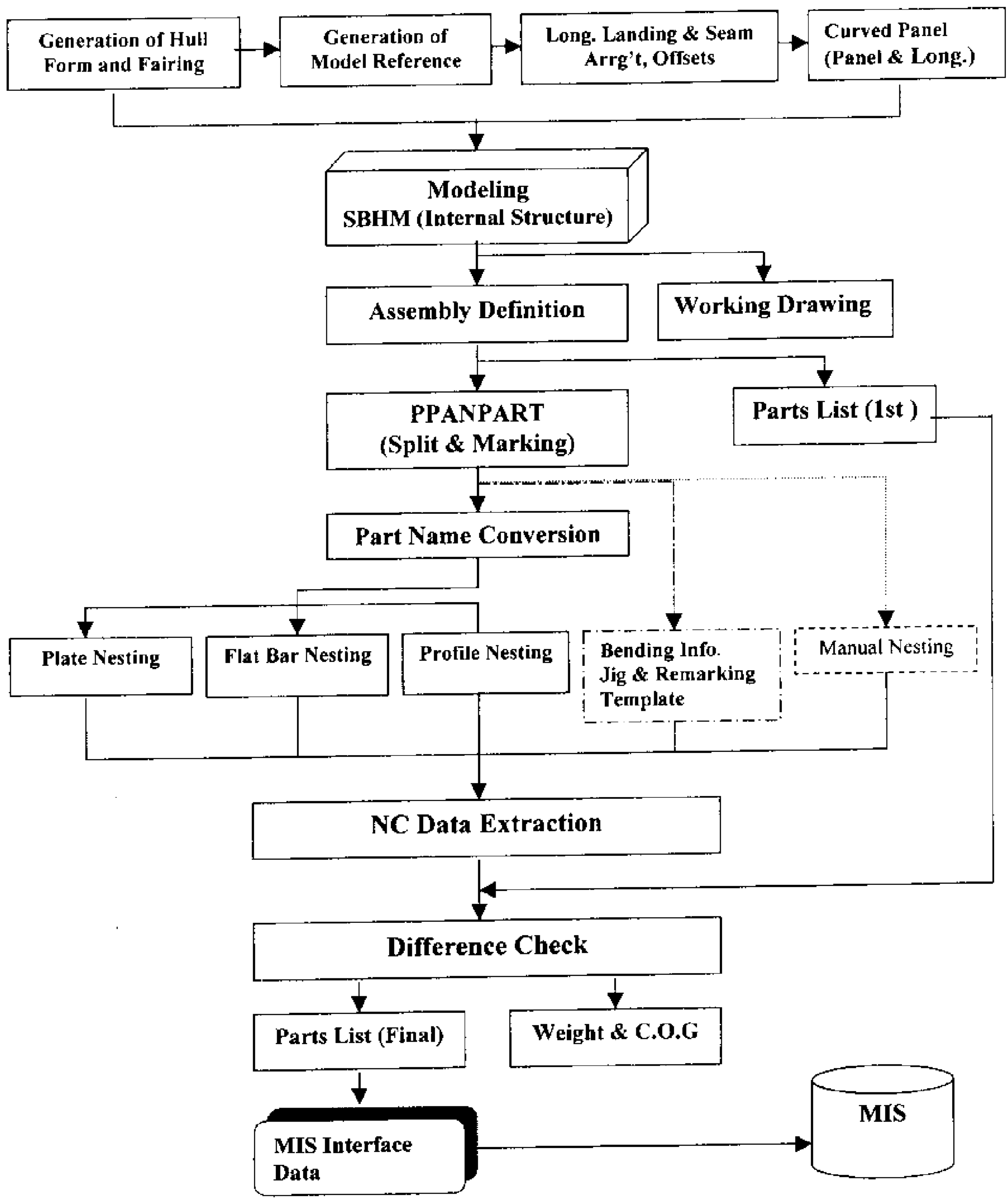


Fig.4 SHIPYARD NETWORK CONFIGURATION



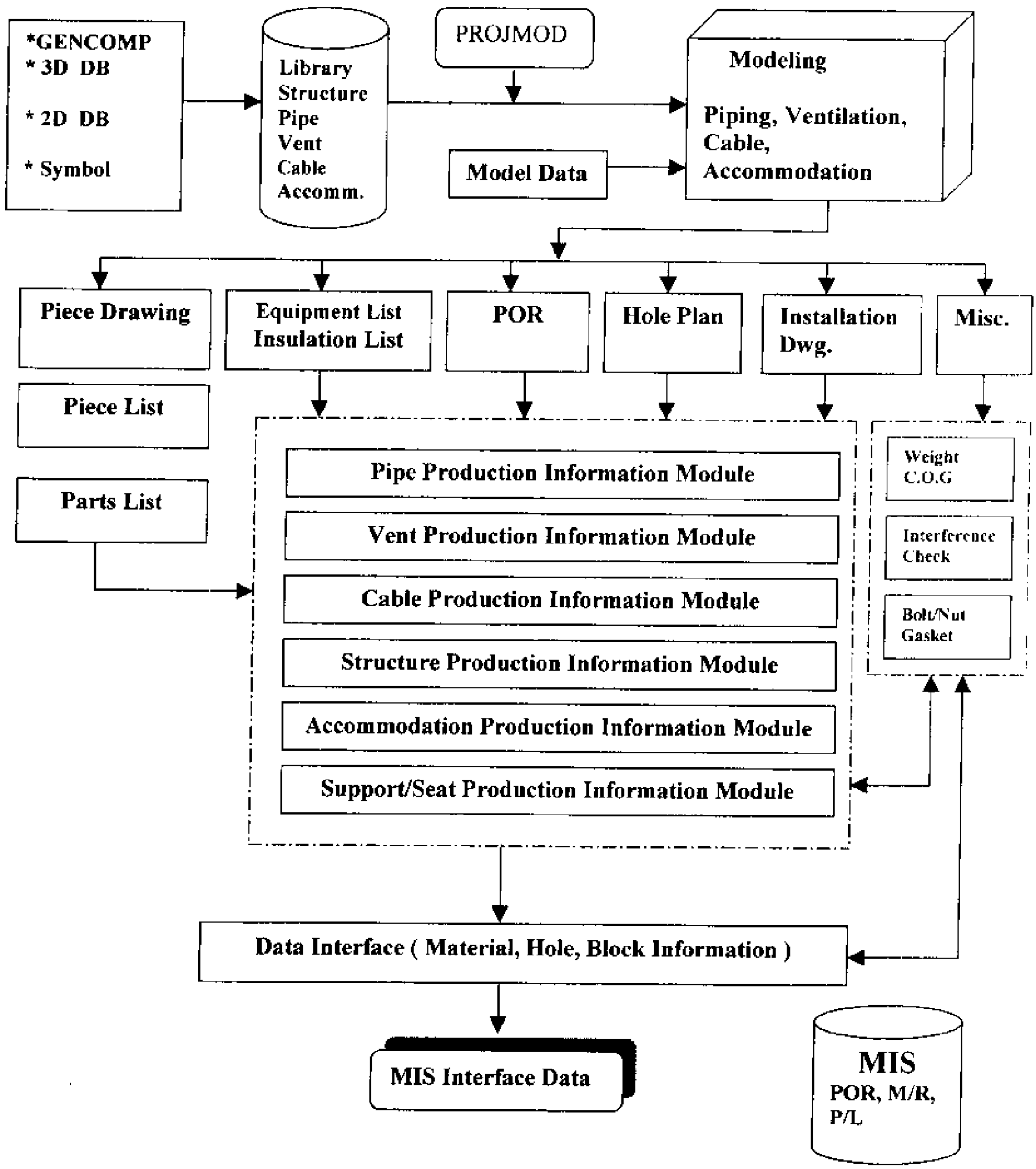
6.2 Overview of the Hull CAD/CAM System

Tribon Hull is an application for the design of main Hull structures.
All kind of hull production information will be produced through Generation of Hull form, modeling, automatic parts development to production information as below.



6.3 Overview of the Outfitting CAD/CAM System

All outfitting production information will be provided using Tribon Pipe, Tribon Cable, Tribon Structure, and Tribon Equipment with Tribon General Design as following steps.



6.4 Automated Robotic Production System

1) Robot Operating System

All welding robots, including profile cutting robot, are Tribon interfaced with off-line programming.

IOPS, IGM Off-line Programming System (ROBCAD) which is 3D-CAD/CAM tool for off-line programming and simulation is provided to process automated robotic production.

The ROBCAD consists several modules of ROBCAD/Base, ROBCAD/Arc, Auto planner, OLP Interpreter, Calibration Unit and Tribon interface.

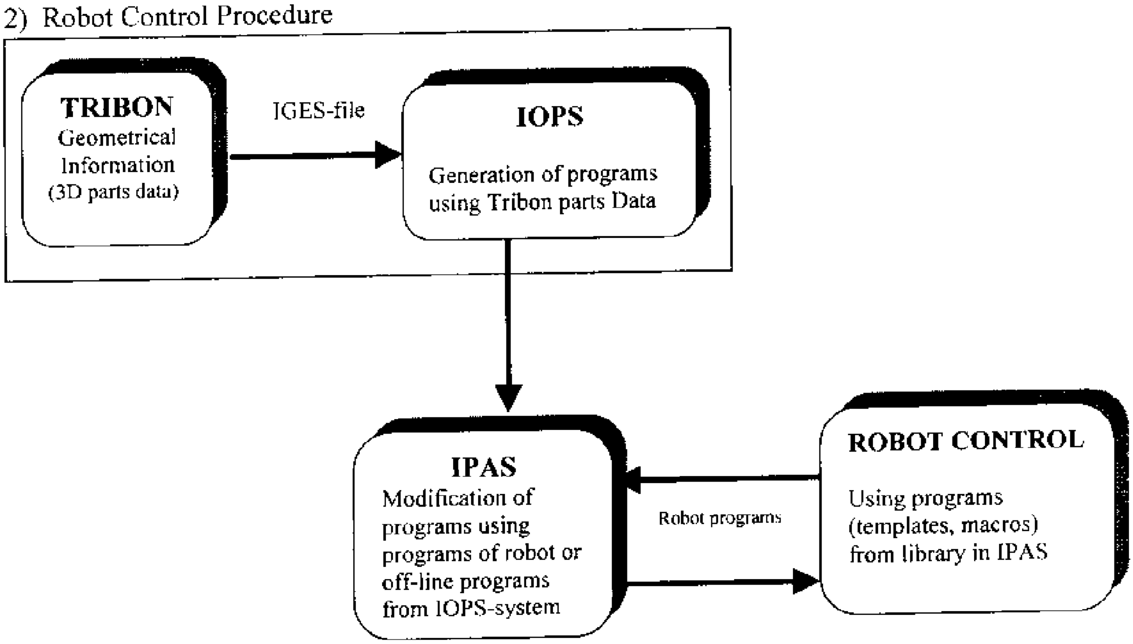
ROBCAD/Base
Mechanical and kinematics modeling work cell layout, simulation, data management. and task description language and drafting module.

ROBCAD/Arc
ARC welding process module.

Auto Planner Module
For profile cutting robot.

OLP Interpreter
Controller language IGM2, measured robot, communication controller.

2) Robot Control Procedure



Robots, slides and manipulators are modeled in IOPS. Fixtures are also modeled.

Workpieces can be modeled in IOPS through Tribon interface using piece parts data in 3D Product Model.

All welding process can be simulated in real time by 3D representation. Accurate simulation of programs and the robot motions allow the operator to view and analyze the production process, program flows, I/O signals, cycle times and eventual erroneous work can be checked in advance.

The robot line is controlled by a number of computers and all information given to the system in the Work Preparation Phase is saved and used by the different sub-control-systems contained in the production line.

7. Conclusion

MHI Shipyard is outfitted and engineered to optimize the use of fully automated and semi-automated production process and work station.

The chosen equipment and its arrangement assure safety, quality and minimum work content for each value-adding step in the shipbuilding sequence. Their features reduce time and cost and raise quality and safety.

Through an integrated computer-aided design and computer-aided manufacturing system using Tribon and modern accuracy control robotic equipment, MHI will be capable of providing the best quality vessels to its customers and establish and undisputed competitive position not only in the United States but also in the international market place

APPLICATIONS OF CAD/CAM VR IN SHIPBUILDING

Dr. Oskar LEE Kwok Lum
Shipbuilding & Offshore Engineering Department
Ngee Ann Polytechnic of Singapore

Abstract

In recent years, Singapore marine industries have to compete in world market with escalating cost in manpower and fixed assets. To solve the problem, some shipyards have started investing in regional production and repair facilities in the Asia Pacific to improve profitability. By sub-contracting some of the labour intensive hull assembly work overseas in lower cost countries, shipyards are able to remain profitable and competitive on the world market. However, the need to plan production and train workers overseas has called for new methods in planning assembly procedures and training manual workers to erect ship hulls. One of the many solutions explored is the application of Virtual-Reality (VR) to Computer-Aided-Design/Manufacturing (CAD/CAM) techniques.

This project started with an idea to facilitate VR based digital assembly techniques to our shipbuilding industries. The basic objectives are firstly to assist shipyards to train overseas workers who may have relatively low education level and also very little shipyard experience. Secondly, precise planning of overseas production may help to reduce the capital investment and implement just-in-time manufacturing techniques. Existing CAD/CAM techniques and capabilities of the aerospace industries, which have billion dollar development budgets, also help to shape our ideas. Feasibility studies on a low budget lead to the adoption of much lower cost PC based system with little compromise in performance. The ideas have been presented to and well received by the local society of naval architects and marine engineers, although the recent Asian economy crisis has somewhat diluted future development interests.

The paper starts with a short summary of the application of PC-based CAD/CAM techniques in the local shipyards. The implementation of an integrated CAD/CAM production system will trace through the change in manufacturing processes undertaken by a local shipyard. The paper ends with the definition of development work in software programming in order to have a more interactive VR tool for planning ship production. Further work is directed towards the development of fully interactive VR techniques to overcome some of the difficulties experienced in this feasibility study.

Introduction

The Background

In the last few years, Singapore has progressed from a developing nation into an industrialised and cosmopolitan city in Southeast Asia. Whilst the population enjoys the higher standard of living, the local industries have to deal with the rising cost of asset and labour as a result of a buoyant economy. Many shipyards in Singapore have ventured into the Asia Pacific region to establish factories and production facilities. The intention is to counter the escalating asset and unit labour cost back at home as the marine industries have to remain competitive in world market. To solve the problem, shipyards have started investing in regional countries where the cost base is much lower. Both production and repair facilities have been established in various countries in the Asia Pacific.

With proper planning and management, such strategic deployment of production and repair facilities will contribute to improved profitability and long term growth of the otherwise labour intensive and slower growth marine industry.

Problems in Venturing Overseas

In sub-contracting some of the labour intensive hull assembly works overseas in lower cost countries, shipyards can hope to increase profitability and remain competitive on the world market. However, new ventures also create new problems in production, logistics and planning in general. Whilst the use of CAD/CAM production techniques in head-quarters for design and fabrication is well established as described previously[1], the implementation of complete production in some countries is not so practical. By the nature of the stage of development, the lower cost countries also imply a less efficient infrastructure and a less educated workforce. Also, these countries generally have less political and social stability, so one has to be cautious to balance risk and returns in long term capital investment. By and large, the business strategy will be one of limited investment to start up an offshore assembly facility to service the main shipyard back in Singapore. Complete assemblies and construction blocks can be produced and assembled in the lower cost countries and towed back by barges for final assembly of a complete vessel. Such a strategy involves limited investment in yard facilities and tends to lower the risk of business during the infancy of the overseas venture. However. the infrastructure, logistics and labour force problems still need to be handled carefully in order to achieve a viable and profitable operation.

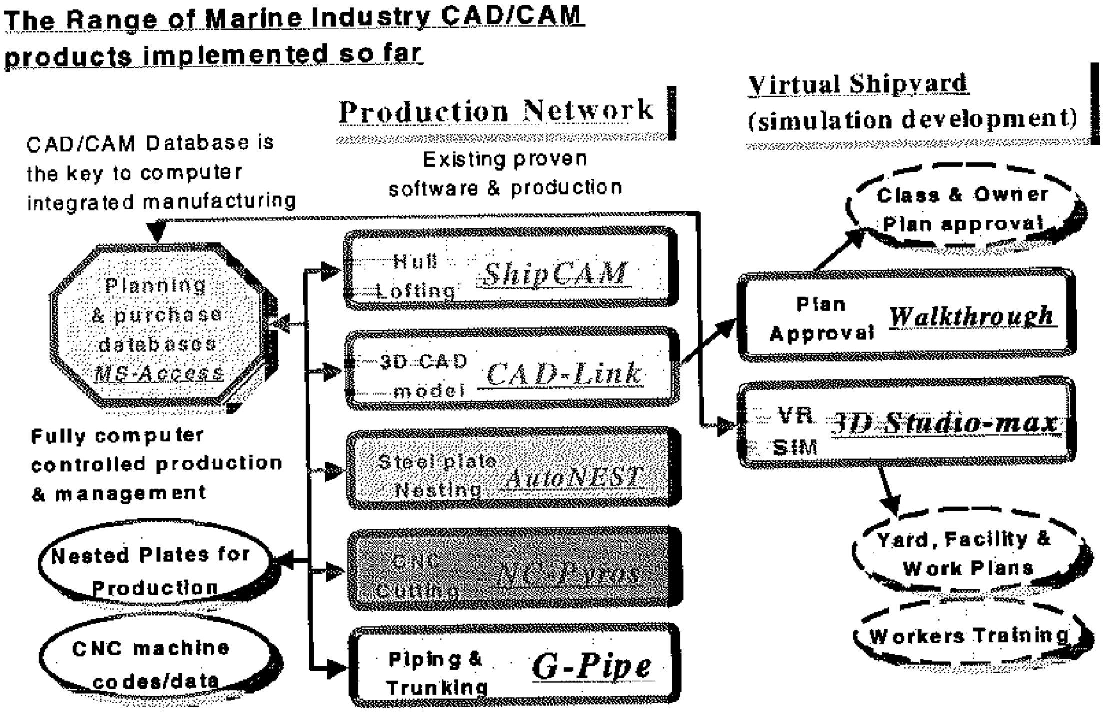
Possible Solutions via VR

Most of the infrastructure problem can be solved by careful logistic and fabrication planning to ensure smooth production of the vessel overseas. However, the problem of a less educated workforce highlights a need to train overseas workers specially to cater for their lower education level and the lack of shipbuilding and repair experience. Both problems have called for new methods in planning assembly procedures and training manual workers to erect ship hulls. This paper centres on some of the many solutions explored by the researchers of this institution together with the assistance of shipyards and computer software developers. Two CAD/CAM based techniques have been identified as potentially beneficial in solving the above problems. Both methods make use of the concept of Virtual-Reality (VR) based Computer-Aided-Design/Manufacturing (CAD/CAM) techniques. To ensure full compatibility of the emerging VR technology with the existing CAD/CAM production as described in [2], feasibility studies are made with some existing off-the-shelf VR orientated software. These feasibility studies allow the shipyards to appreciate the effect of the end product without a major capital investment in software development by either the industry or this education institution. The development of these feasibility studies culminated in the bulk of this technical paper.

Techniques in Brief

One technique is the widely explored simulation of "walk-through" of three dimensional(3D) spaces within structures. This technique is considered to be useful for owner and classification society plan approvals as one can then identify both global and minor interface errors by taking a walk through the actual internal structures of a vessel. The second technique is by simulation of 3D digital assembly of all the structural parts of a ship within the computer prior to any in situ problem being encountered on the shop floor. This technique has been applied in aircraft engineering [3] and is very effective in production planning and potential trouble shooting. Of course, the VR technique deployed herein cannot match the same sophistication of that used in the aircraft industries which have the luxury of heavy investments in main-frame computers and dedicated software. However, sufficiently powerful PC and software is used in the VR feasibility studies to ensure potential users can appreciate the end-effect of digital assembly.

Figure 1. Feasibility Study taking CAD/CAM into the Virtual Reality Environment



Existing CAD/CAM

CAD/CAM implementation

The shipyards concerned generally have some existing CAD/CAM production processes in place and there is little incentive in making changes to the current practices in view of the capital investment and worker re-training required. Full details of the typical CAD/CAM processes have been described previously [1 & 3], only a summary is outlined below for completeness in this paper. All the CAD/CAM design and production techniques are already implemented via PC networks without the luxury of workstations and mainframes. Most, if not all, of the CAD/CAM software are used by local shipyard workers with little or no formal qualifications and minimum training. One of the important criteria of these projects is to ensure all software utilised are PC-based and user-friendly enough to be readily learnt by the average worker who already has some PC-based knowledge due to the wide-spread usage of PC's.

Figure 1 is a summary of the software integration processes and gives a clear outline of the production CAD/CAM tools in use for the projects concern. In the central column of the figure is a listing of the software tools given in the sequence of the CAD/CAM production processes. A key feature is the use of ***Microsoft Access*** as the common database which helps to track all the key dimensions, weights and CG of all parts used in manufacturing and also the virtual reality applications.

Mould Loft Activities

Starting at the top of the central column in figure 1, ***ShipCAM*** is the complete fairing and lofting tool which essentially replaces all design activities from the mould-loft shopfloor by means of computer lofting. For the vessels studied, ***ShipCAM*** has been used to fair the complete hull of the vessel by means of manipulation of 3D splines in the computer. The computer-generated hull of the vessel is a 3D surface model at full scale stored and further manipulated by the software. ***ShipCAM*** then generates internal stiffener cutouts, individual expanded shell-plates, plate-forming templates, inverse-bending templates for stiffeners, pin-jig assembly offsets and other essential data for the downstream CAD/CAM production processes.

3D Structural Detailing

Following on from *ShipCAM* is ***CAD-Link***, which is an ***AutoCAD*** based drafting tool used for detailing all of the internal structural parts of the vessels. ***CAD-Link*** is a critical aid in detailing all parts of the ship as it has full 3D error-checking characteristics built-in. While the designers may draw in 2D, the software automatically assembles the 3D internal structures by extruding the 3D solids within the computer. The detailed part can be tested for form, fit and tolerance and its CNC machine cutting path is checked even before the profile cutting codes are created for the machine used on the shopfloor. Again, one of the biggest benefit of using ***CAD-Link*** is associated with the database which helps to track all the structural parts used later in production and also the VR project activities.

Nesting and CNC-cutting

Downstream from ***CAD-Link*** are the stock plate nesting and CNC machine code generation activities. Some yards use the semi-automatic nesting facility within ***CAD-Link*** software whilst others make use of a fully automatic parts-nesting software called ***AutoNEST***. The latter is a well-proven tool originally destined for the huge electronics PCB nesting industries; hence some 3000 licensed users of ***AutoNEST*** exist worldwide. Nesting with either software is fairly straight forward since both operate within ***AutoCAD***; the definitive standard for PC based computer drafting in Asia. The CNC machine codes are generated from ***NC-Pyros***, another user-friendly PC-based software which takes the drawing from ***AutoCAD*** and convert the lines and arcs into machine usable codes. ESSI and G-codes have been employed for the projects concerned. However, both the nesting and CNC coding operation are well-proven and do not form the basis of future development, further details of these activities and software are available in the references [1, 3].

Piping System Design

Another downstream activity in parallel with ***CAD-Link*** is the design and layout of the systems of pipes, plants and accessories. ***G-Pipe*** is a dedicated software developed solely for the purpose of piping design and operates within the ***AutoCAD*** environment to take advantage of the already drawn internal ship structures. It has a very comprehensive library of valves and fittings approved by classification societies and it generates shop drawings from simple piping isometrics. More recent development of ***CAD-Link*** is aiming to overtake the G-Pipe lead by incorporating piping as part of the ship structural design software package. The immediate advantage derives from the use of a concurrent database, which can then reflect simultaneous modification of pipes and ship structure and vice versa. There is insufficient time and computing capacity on this project to take into account the piping walkthrough and assembly simulation. Further work may involve complete ship structures and full piping system assemblies although much will depend on the availability of full piping facility within ***CAD-Link*** as well as the future funding situation.

Virtual Reality Applications

Simulation Projects

As shown in figure 1, the ship design details for virtual reality (VR) development projects are derived directly from the output of ***ShipCAM*** and ***CAD-Link***, both of which store all structural parts details in the ***Microsoft Access*** databases (referred as ***ShipReport*** by the software developers). This arrangement is intended to minimise the re-training of designers and draftsmen in the shipyard if and when VR activities are introduced into the mainstream CAD/CAM engineering and planning office environment. An additional aim of this project is to demonstrate to the shipyards concerned that relatively unskilled students can learn the production software and apply it to the design of fairly complex parts of ships. In all, four students are involved in each of the VR feasibility study projects; all have little or no shipbuilding working experience.

For each project, two students are deployed to do the ship structure detailing using ***CAD-Link***; one is in charge of lofting the shell plates using ***ShipCAM***. The team leader is in charge of experimenting with VR effects via software such as ***Walkthrough*** and ***3D Studio Max***, both of which are ***AutoCAD*** compatible and are products of ***Autodesk Inc***. No effort has been directed towards Parts Nesting and Computer Numerical Control (CNC) plate cutting as these operations are thought to gain little from VR simulations. In all, two ships and one oil-rig designs have been used for the feasibility study. The vessels in the study are either under construction or are proven designs previously delivered, so full CAD/CAM details are already available and proven. The designs of two ship sections have been used, one is a double-skin section of a supply vessel and the second is the complex stern structure of a tanker. The oil-rig design is a standard jack-up type with a triangular platform supported by three legs constructed from tubular trusses. Only parts of the oil-rig are used in the simulation due to time constraint for the study as well as limited availability of high-end PC's for the processor intensive simulation project.

The project teams have taken care not to venture into the area of ergonomics of design, which is also the major area of current VR applications around the world. Common VR applications are similar to the work reported by Dai & Reindl [5], most of which centre around human movements in a VR environment. The projects described below are more akin to those outlined by Jayaram [6] with the intention of combining the power of VR to existing CAD/CAM implementations. Ultimately, one would target future development towards the same goal as envisioned by Costea [7] in designing VR around knowledge based system. For now, the Asian economic crisis is looming and project resources are few, so one has to be more down to earth in the VR approaches. The project teams look for ways to achieve the result with minimum addition of both hardware and software to the existing CAD/CAM implementations. The objective is simply to investigate if the VR environment will help shipyards to be more productive and profitable in planning the logistics towards building the ships for timely delivery.

3D Space Walk-through

There is very little work involved in preparing this part of the project since ***Walkthrough*** is an established tool developed by ***Autodesk Inc***. for use with ***AutoCAD*** based drawings. The use of ***Walkthrough*** software is relative simple and straight forward as long as a 3D CAD drawing has been prepared. Once the drafting of 2D structural drawings in ***CAD-Link*** is complete, the full scale 3D model is readily built by the standard ***CAD-Link*** functions. This 3D model is then used for the walk-through exercise after rendering by the ***Walkthrough*** software (which is the only time-consuming part if one is waiting for a low-end PC to render a large complex structure).

The physical effect demonstrated by the software is fairly good, considering the minimal amount of work required to prepare the model. The user can literally move into every known space within the 3D structural model by manipulation of the PC-mouse in association with the software tools menu. However, the navigation within the 3D structural model requires some experience and prior knowledge of the structural design as one can get dis-orientated quickly after moving through the "maze" of ship structure. One possible improvement of the software would be via the incorporation of two additional view ports to include a floor plan and elevation indication. This arrangement will enable the user to navigate more effectively and eliminate the potential confusion and getting lost in a large 3D structural "Maze". Further discussions and development with the software are now underway via the local authorised training centre for ***AutoCAD*** and ***Autodesk Inc*** in general.

Overall, the users find the software effective and also fits the low budget dedicated to this study. There is therefore no need to develop separate software to enhance the performance of walk-through of 3D structural model. Although the experience of this technique is limited to a short study, users have appreciated many of the potential benefits. In a plan approval situation, owners and class surveyors can readily discuss desired modification right at the spot of interest within 3D structural model. For design optimisation exercises, operators and designers can walk through the 3D structural model together and discuss the aesthetics and ergonomics of the overall space and cabin partition. One must bear in mind that any ship structural changes arrested at design stage is seven to ten times cheaper to implement than one ordered at the production stages. The benefits to owners, surveys, designers and shipbuilders should culminate into tremendous savings in man-hours and work stoppages.

Digital Assembly

The existing software market is short of low cost tools for this application and the project team has to make do with the limited 3D multimedia and animation tools available. The complete assortment of parts for the 3D model of a vessel is taken from ***CAD-Link*** together with the database stored in ***Shipreport*** so that the whole assembly process is driven by the database. All the parts of the 3D model readily built by ***CAD-Link*** are sequentially assembled in the digital shopfloor within ***3D Studio Max.*** The feasibility studies in the digital assembly process turned out to be extremely tedious, and some initially planned processes have been ruled out due to the time constraint.

Within ***3D Studio Max***, every part used for the digital assembly process is sorted and rendered accordingly. The parts are then positioned according to a pre-defined path so that they meet each other in 3D space in an ordered sequence. The teams literally have to "stage and shoot a movie" based on their knowledge and plan of the intended production assembly processes. As described above, the objective is to demonstrate the end-effect of digital assembly, the lack of a fully interactive and user-friendly piece of software is not the critical consideration compared to cost and availability. A number of digital assembly movies of various parts of the above two ship sections have been created via ***3D Studio Max*** and production time decreases with the experience of software usage. Figure 2 and 3 are computer screen shots of the assembly sequences of an oil-rig. Figure 2 shows the assembly of the bottom structure and the strengthening bulkheads, starting from scratch and finishing with all tank-tops and bulkheads in place. Figure 3 is a continuation of the assembly processes with the shell plates and bottom reinforcement ring segments being "hoisted" into place. The compartment thus completed forms only the supporting structure for one of the three "jack-up" legs of the oil-rig.

Figure 2. Digital assembly of internal structures of a Jack-up compartment

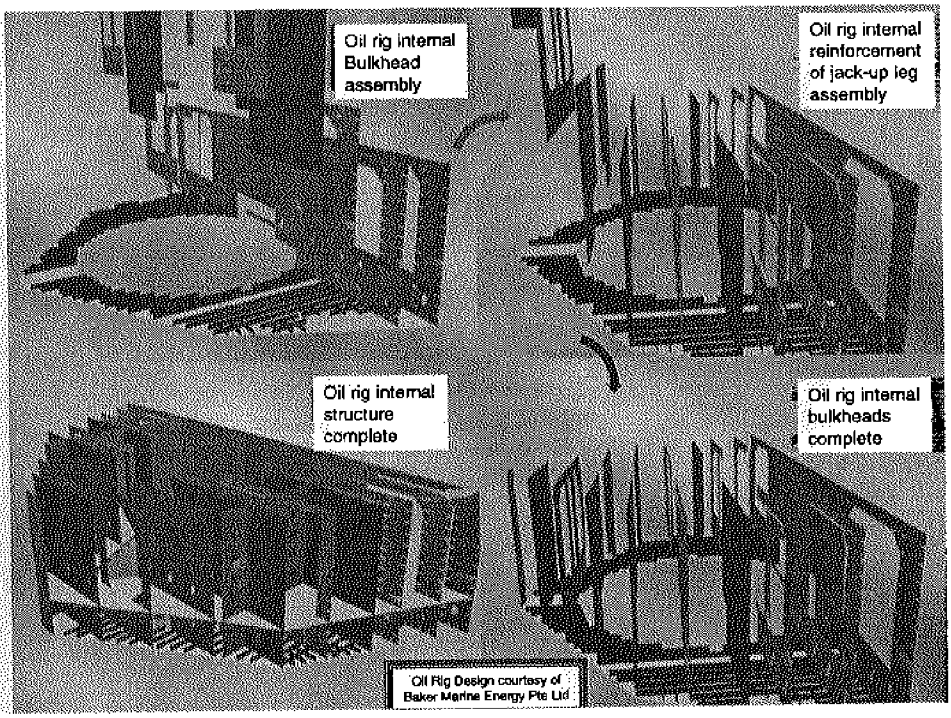


Figure 3. Digital assembly of external structures of a Jack-up compartment

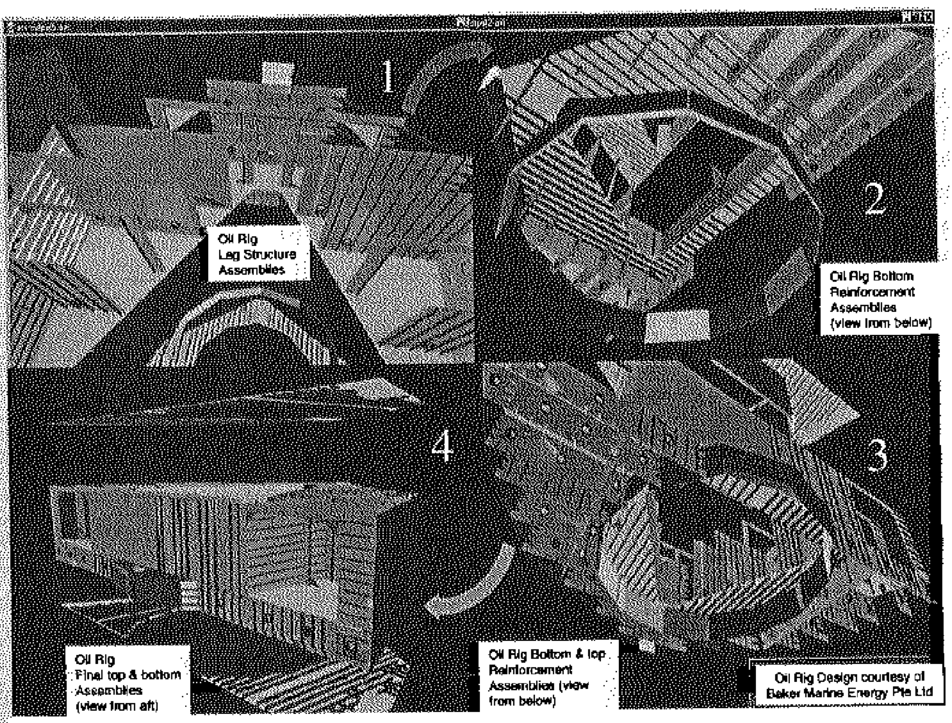
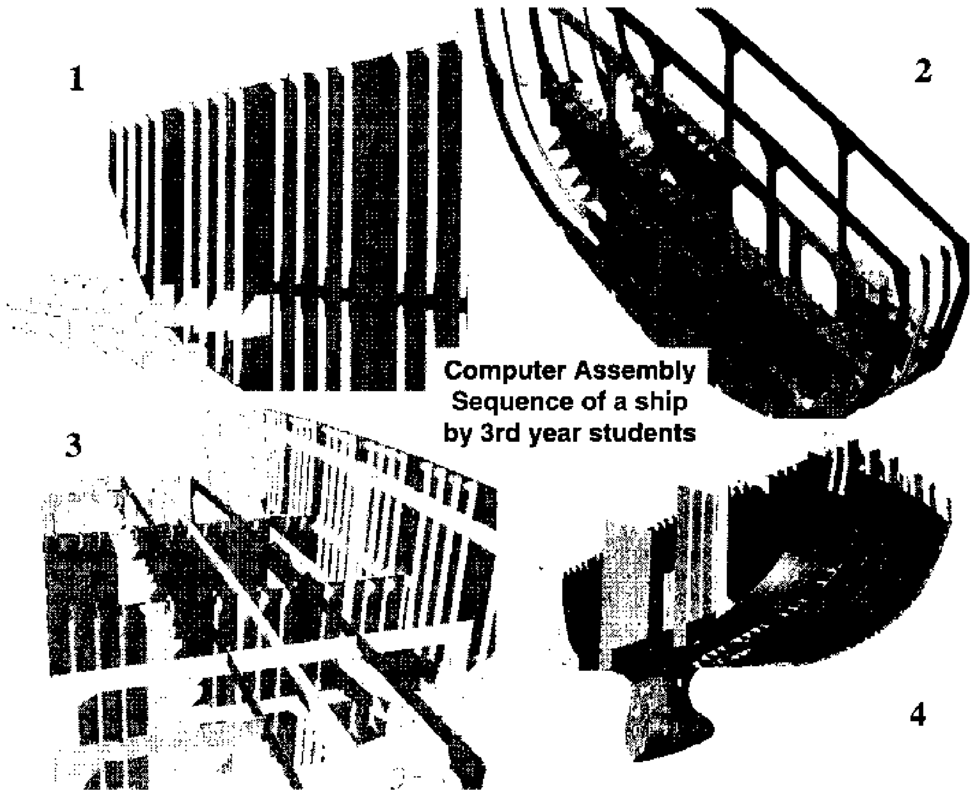


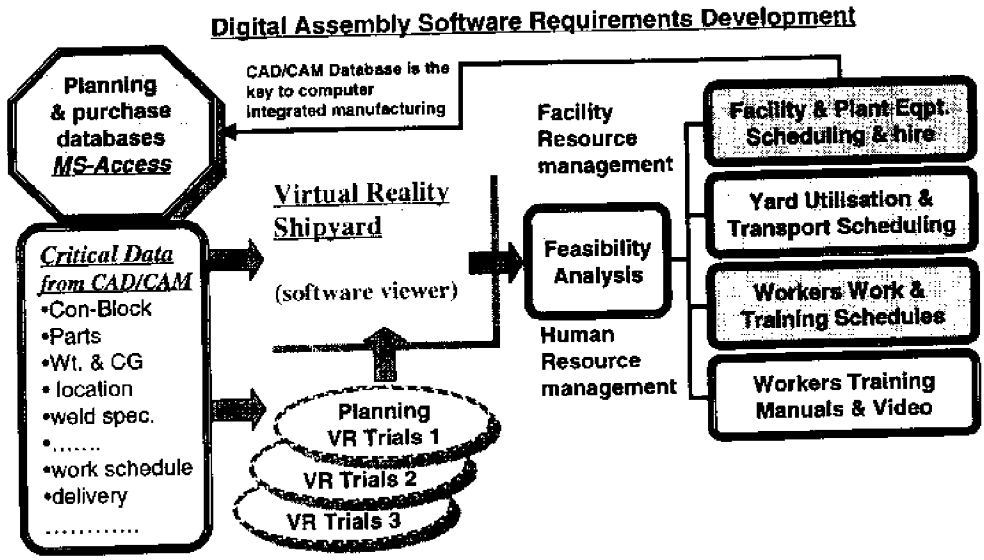
Figure 4. Digital assembly of the stern section of a 7000 DWT oil tanker



For completeness, some snap-shots of the digital assembly of a tanker are also included herein. Figure 4 shows some screen shots of the digital assembly of the stern section of a 7000 DWT oil tanker. The main challenge and differences in assembling a ship are the need to cater for plate forming and line-heating of the curvaceous shell plates of the stern area of the ship. However, the tedium of checking plate forming and plate edge allowances is well taken care of by the ***ShipCAM*** software in this case. The many solid floor construction in the stern area of the ship only adds to the value of the feasibility study as accessibility for welding and ease of construction become feedback parameters in the development work.

Overall, the ***3D Studio Max*** software performed the tasks as intended and the experience gained has prompted the team to define the software functionality of future software development for digital assembly. Apart from providing a platform for shipbuilders to train shipyard workers overseas, the digital assembly study has demonstrated more benefits. Some shipyard planners believe they can now plan the crane utilisation, jig usage and other production tooling so accurately that they can save the capital investment of some of these facilities overseas. Finally, the digital assembly processes also highlight to them the potential manpower and equipment deployment when assembling more difficult sections or building blocks.

Figure 5. Future development of fully interactive VR software for shipyard digital assembly



Future Development

Much of the follow-on work originally intended would have to depend on funding and support of the various industries described below. The Asian economic crisis has certainly blown a big hole in the balance sheet of many industries, the marine industry included. The last year has seen a couple of medium shipyards consolidating and re-aligning the respective business activities. The economy turmoil has devalued the currencies of many developing nations around Singapore; in turn making the labour cost of these countries more attractive for labour intensive industries. On the other hand, the political and social instabilities in these countries have aggravated the investment risk. At this point, it is unclear whether the economy crisis will drive the Singapore shipyards towards accelerated development in CAD/CAM and VR in order to counter the effect of much higher unit labour cost as a result of greater currency stability in Singapore. This period of economy uncertainty has caused the delay and curtailment of many projects and one can only hope that technological advancement for the sake of better business profitability will prevail.

The feasibility studies of 3D digital assembly and 3D space walk-through of two major ship structural sections and an oil-rig has been successfully completed by relatively inexperienced workers (students). The study reveals the potentially great benefits to shipbuilders by applying these VR based CAD/CAM techniques. In the case of 3D walk-through of ship structures, the techniques and software exist and it is a simple matter of applying it within a CAD/CAM environment. For digital assembly of ship structures, the desired software tools do not exist on the market and further research will be necessary to develop dedicated software. This study has provided the team with an in-depth knowledge of the desired functionality and processes in truly user-friendly digital assembly software. In the coming months, some further application studies of digital assembly of offshore oil production vessels will be made in response to the needs in this buoyant market. Meantime, the definition of new software functionality and features are underway to develop the desired digital assembly software.

With the addition of these new CAD/CAM techniques, the original intention of a simple but integrated PC-CAD/CAM process for the marine industries [4] has truly come a long way.

Further work is aimed at the definition of development work in software programming in order to have a truly interactive and user-friendly VR tool for planning ship production. Figure 5 is a summary of the software requirements development currently taking place and gives a hint of the multitude of parameters which determine the course of assembly sequences. The left side of the figure is the all important CAD/CAM database which stores all the essential details of the parts to be used in the VR shopfloor. Although the type of software viewer has not been decided, it will have to contain multiple view-ports which give supplementary information on the actual layout of the shipyard. Some form of Artificial Intelligence (AI) will be ideal in the feedback loop which determines the best course of action. The right hand column of the figure outlines the output essential to the management of all the shipyard resources, whether machinery or human. The operation of a shipyard may be a far cry from the existing practices in Singapore, but then we are at the cross-road of progress in these changing times.

Acknowledgements

The author wishes to express his appreciation of software and hardware support from Albacore Research Ltd of Canada and Peripheral Solutions Pte Ltd of Singapore respectively. The design details of ship and oil rig in software development and integration have benefited from the kind support of Keppel Singmarine Dockyard and also Baker Marine Energy Pte Ltd respectively. Many thanks to Ngee Ann Polytechnic for funding the publication of this paper and the effort of students in making this project a success.

References

1. Oskar K.L. Lee and Rolf G. Oetter (1997) "Integration of CAD/CAM in Smaller Shipyards", Proceedings of the 6th International Marine Design Conference, Newcastle, UK.
2. Oskar K.L. Lee, (1997) "Shipyard2000", Proceedings of the 11th Asian Technical Exchange and Advisory Meeting in Marine Structures, Singapore.
3. Steven G. Lynn, (1996) "CAD/CAM in the Aerospace Industry", The CAD/CAM Handbook, McGraw-Hill.
4. Oskar K.L. Lee, (1996) "Ship Production using PC-based CAD/CAM", Proceedings of the 10th Asian Technical Exchange and Advisory Meeting in Marine Structures, Pusan, Korea.
5. Dai & Reindl, (1996) "Enabling Digital Mock-Up with Virtual Reality Techniques", Proceedings of ASME Design Technical Conferences, Irvine, CA., USA.
6. Sankar Jayaram, (1996) "Virtual Reality and CAD/CAM", The CAD/CAM Handbook, McGraw-Hill.
7. Ileana Costea, (1996) "The Role of Knowledge-Based Systems in C4", The CAD/CAM Handbook, McGraw-Hill.

PDOS SYSTEM IN SUMITOMO YOKOSUKA SHIPYARD

Masao Saito
Ship & Steel Structure Group, Sumitomo Heavy Industries, Ltd.,
Natsushima 19 Yokosuka 237-8555,Japan

Kazuhito Matubara
Yokosuka System Office, Sumitomo Heavy Industries, Ltd.,
Natsushima 19 Yokosuka 237-8555,Japan

ABSTRACT

Sumitomo Heavy Industries, Ltd. have developed a production management system 'PDOS' for Yokosuka Shipyard and have started to apply it for they're manufacturing of merchant vessels. 'PDOS', which stands for Production Digital data Operation System, is a sub-system of Sumitomo CIM system 'SUMIRE', and aims at production process closely connected to planning product design. That means, the production process is planned concurrently with the product design based on its detail information. Therefore, it could contribute to less lead time throughout building vessels.

Outline of PDOS system is shown in this paper.

INTRODUCTION

Sumitomo Yokosuka Shipyard has been building mainly merchant vessels since 1971. We have always carried out many various improvements with the intention of providing for a customer with a best vessel.

CIM system's development is one of improvements for manufacturing of vessels. In our Shipyard, 'C1-Project' was carried out from 1989 to 1994. 'C1-Project' which studied next-generation advanced computer system.

After 'C1-Project', we have developed 'SUMIRE' which stands for *Sumitomo Manufacturing Innovation and Re-engineering*. Ship manufacturing is complex and large-scale project. It generates enormous volumes of information for each ship at short lead times. Hence SUMIRE system is aiming for an information management system for design and production fields. It is required that design and production stage have to be carried out concurrently and consistent. SUMIRE system has been resolved those problems.

SUMIRE system is consists of the sub-systems. We have already reported several sub-systems which is mainly CAD system. (1)

This time, we have report production field sub-system of SUMIRE system. This sub-system of SUMIRE is called 'PDOS' (*Production Digital-data Operation System*).

We developed PDOS with the intention of a productivity increase and a reduction in the overall duration of a ship product. We have always carried out for that purpose of many various improvements in each job stages. (IE, TQC etc.) Though we would not obtain a real productivity, only those improvements. In order to rise in production throughput, we must mange consistency scheduling and planning.

OUTLINE OF SUMIRE SYSTEM

SUMIRE system consists of the sub-systems shown in Figure 1 with some other miscellaneous sub-systems.

Basic Design System

Basic Design System mainly manages initial specification data of vessel, such as particulars of machinery and equipment, material of piping, painting, etc.

Conventional CAE Systems

Conventional CAE Systems, such as propulsion analysis, stability calculation, stress and fatigue analysis, etc.

IDOS (Initial Design for Optimization, by Simulation) System

IDOS System is a simulation based design system to define satisfactorily optimum basic hull structure at an early design stage. Materials data of first planning stage for PDOS are produce from IDOS. IDOS model is also utilized as an initial model for H and F systems.

H (Hull) System

H (Hull) System manages detailed hull structure model. Hull-parts data for PDOS are produce from H model. Also data for CAM systems are produced from H model.

F (Fitting) System

F (Fitting) System manages detailed fitting model include machinery, equipment, pipes, electric cables, etc. Fitting-parts data for PDOS are produce from F model. Also data for CAM systems, data for procurement system are produced from F model.

Steel Material Procurement System

Steel Material Procurement System manages purchase order and receipt of steel material. Steel material data for PDOS are produced from Steel Material Procurement System. Also this sub-system controls delivery of steel material from stockyard to fabrication shop.

Fitting & Equipment Procurement System

Fitting & Equipment Procurement System manages purchase order and receipt of fittings and equipment. Fitting & Equipment's parts procurement data for PDOS are produced from Fitting & Equipment Procurement System.

H and F are advanced modeling systems that have been developed together by Ishikawajima-harima Heavy Industries, Ltd. (IHI) and Sumitomo.

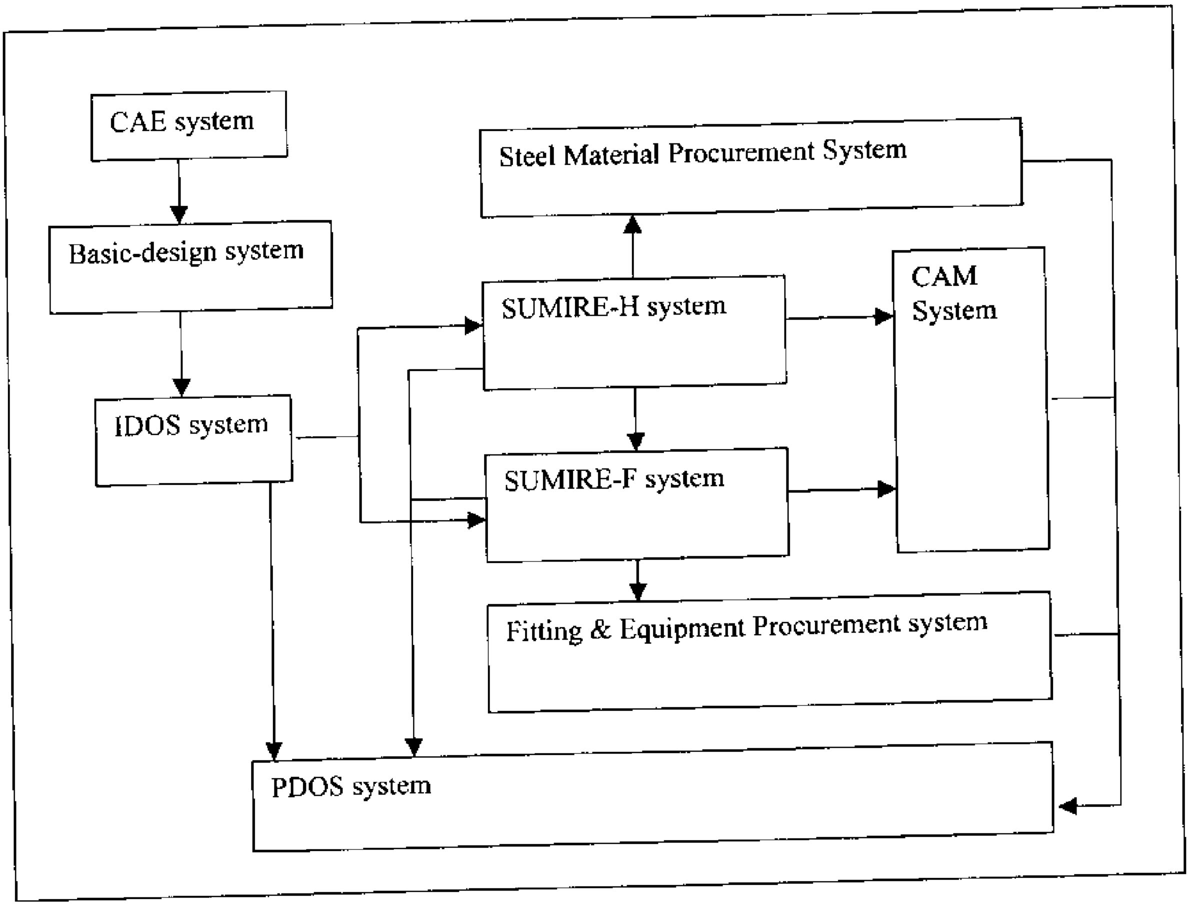


Figure 1. Outline of SUMIRE System

PDOS CONCEPT

In the following, the main characteristics influencing the required production total information system.

1. Ship manufacturing is complex and large-scale projects. Hence only a few planner can not control actual schedule. Therefore a production-planning system consists of several planning layers. Such systematization aims not only realization of flexible scheduling, but also the promotion of decentralized control such as independent planning.

2.The system's function as exchanging each layer's information. Also member of production planning group can operate own scheduling each layer. Hence they have been realized in high productivity plan and scheduling under the constraint of other layer's schedule and other stage's schedule.

3.The system is handled parallel progress projects.

4. The system can be managed on parts and materials data by the design system and by procurement system.

5.The system can be managed process flow diagram, relation between process and design parts, materials.

6.The system is set up on integrated data base for centralize information. But functional distribution while trying to maintain compatibility and easy operations.

Considering those situations, PDOS system consists of some concept module, which describe as will be shown later.

Manufacture Control Unit Definition Module

Manufacture control unit is defined as a group of works and parts. Production process/schedule etc. is defined by each unit. Those units in PDOS are proposed as "Bird model", which consists of process wing, parts wing, and unit body.

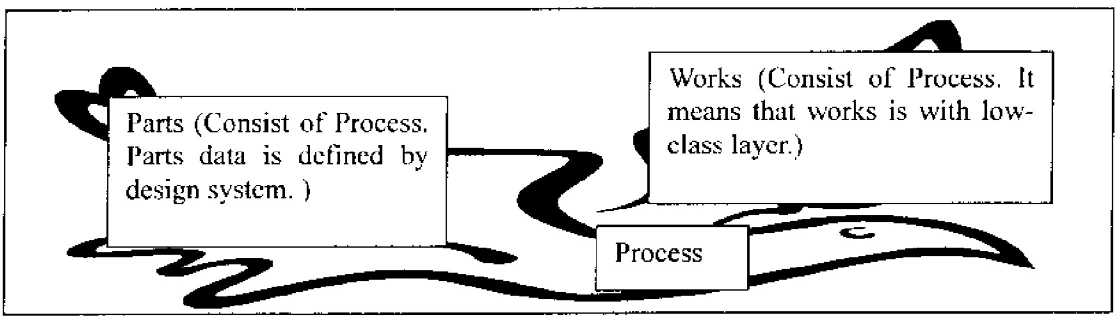


Figure 2. Bird Model

Process Planing Module

Process plan consists of work sequence and description of each work. Also process plan has some layers. Process-information of these layers has been inherited on upper-class layer's process information. Hence process planning is easy operated.

Scheduling Module

Production schedule is closely related to process plan. Scheduling module has several layers, similarly to Process Planing Module.

Linking with CAD Data module

The whole parts data is designed by SUMIRE system, and transferred to PDOS. Each process in PDOS is managed linking with CAD parts data.

Linking with Actual Results of Work module

Managers have to check actual results of work, such as schedule, cost, efficiency, etc., to the plans at any time. Each process in PDOS is managed linking with such actual result data.

Work Order Sheet module

Work order and its description are issued based on PDOS process plan data. Work order description consists of schedule, assembly details, assembly sequence, and 3-D view.

System architecture

All data of PDOS is integrated in RDB on EWS. Editing of production process network, scheduling and estimate of resource functions covers a PC. Execution of optimizing of scheduling etc. for to be demanded by PC is covered by the EWS.

PDOS SYSTEM

Application modules are shown in Figure 3.

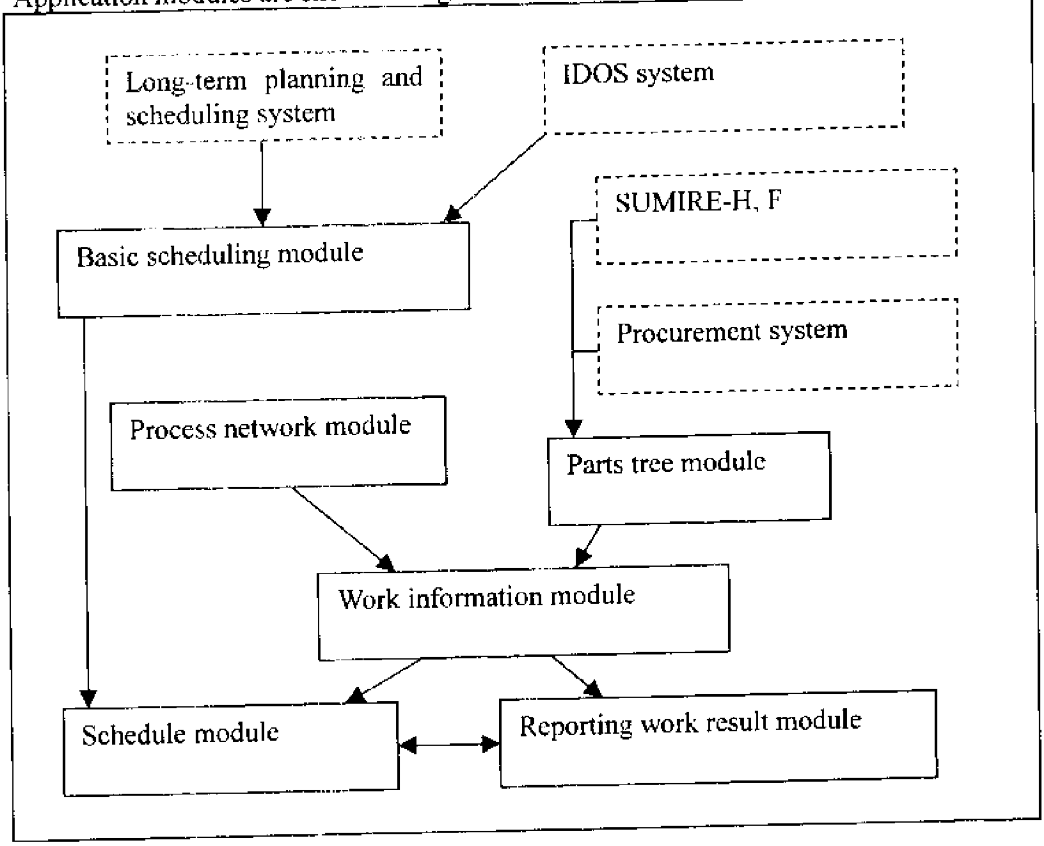


Figure 3. Application modules of PDOS

Long term planning and scheduling

Long-term planning is carried out before the initial design. It covers over 2 years. This system has empirical parameters/values to estimate man-hours of each production stages. This system is operated in shipyard management Dept. PDOS is received milestone date (launching, delivery, etc.) from this system.

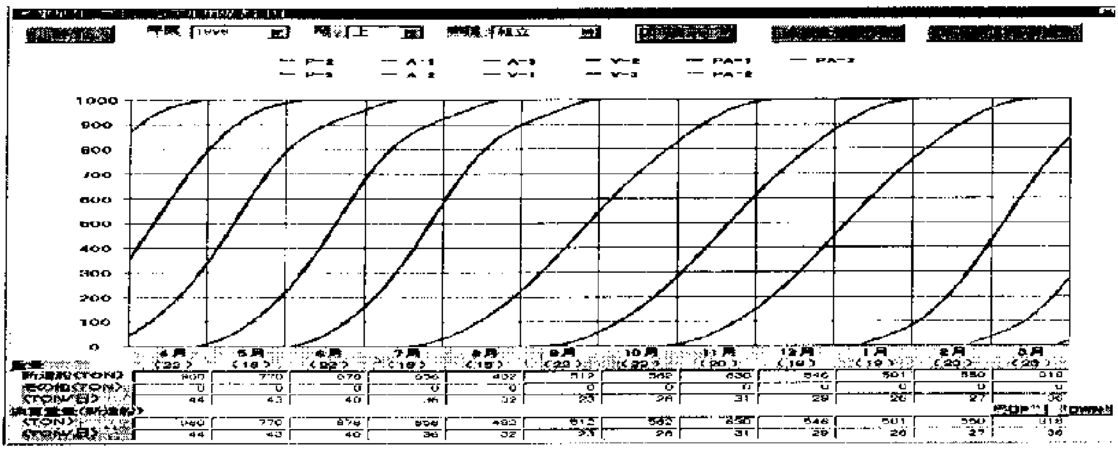


Figure4. Long-term planning

Parts Tree

Basic planning unit is a block, which means a finished good by assembly stage. Design section is defined to only parts as parts-name, sub-block-name, production-stage-name and block-name. Though at detail planning, planner must be defined various unit of parts. This unit is said an interim-product. For instance, pre-assembly unit is consists of several sub-blocks. That unit is not defined by design stage clearly. From this, the system is easily defined to an interim-product on Parts tree editing screen. Naturally enough, the system is received parts-name etc. from SUMIRE-H. F. Figure 5 shows an example of Parts tree editor

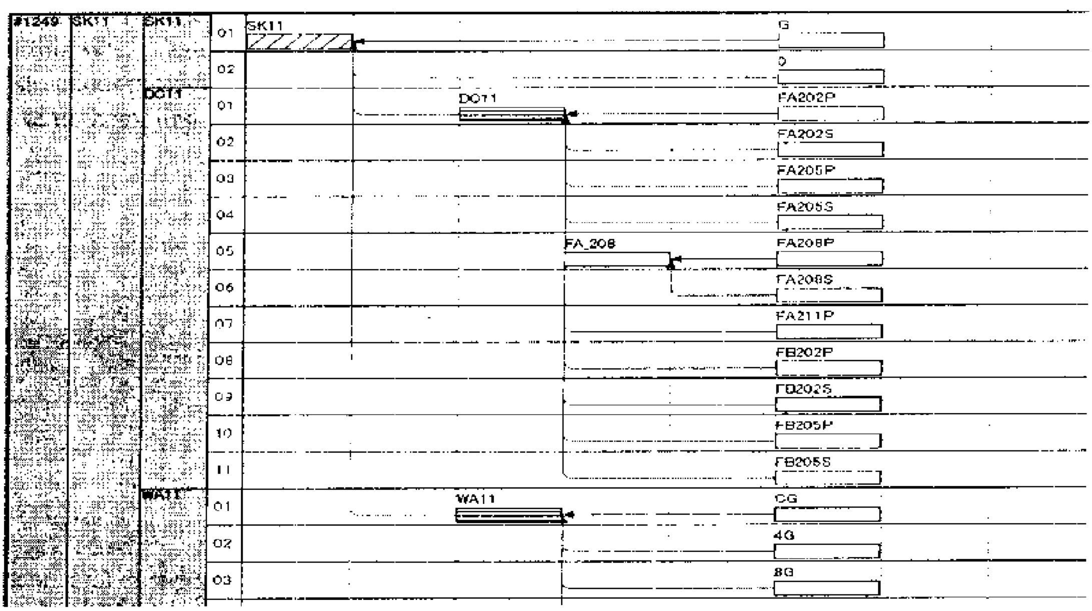


Figure 5. Parts Tree for Blocks

Work Instruction Drawing

On the other hand, 3-D assembly drawings are called up easily on display by SUMERE-H, F-system. Let us recall that we have defined the *interim-product* already. Hence the system can be obtained some product information at each *interim-product*. The information is as follows: weight, dimensions, specifications, weld length, standard man-hour each work, etc. Also the 3-D assembly drawing are can be made use of a work instruction sheet for worker.

Figure 6,7 shows an example of 3-D assembly drawing and Parts list.

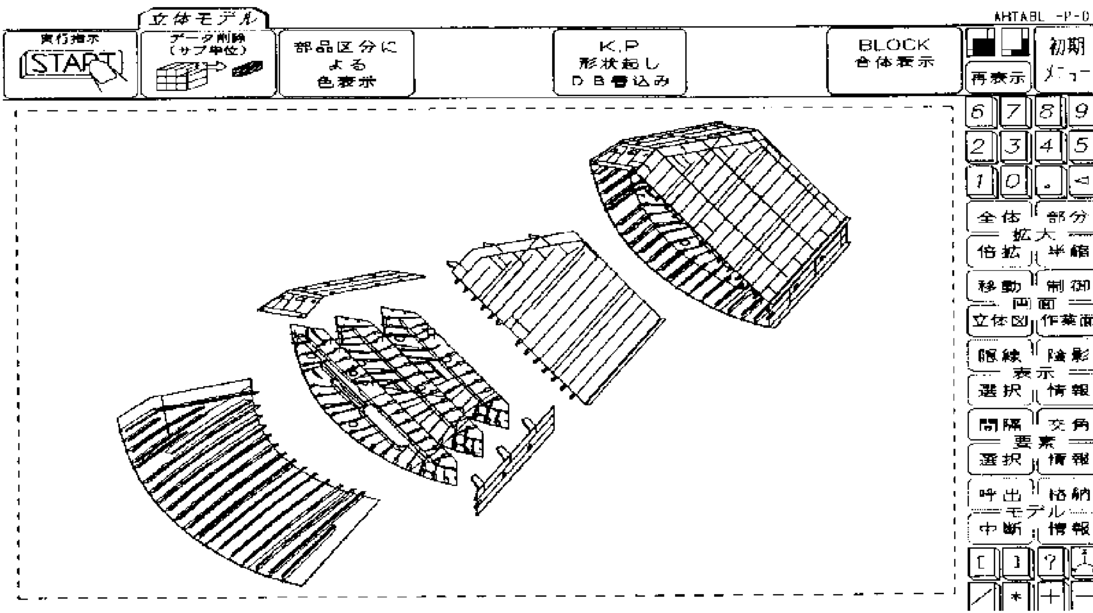
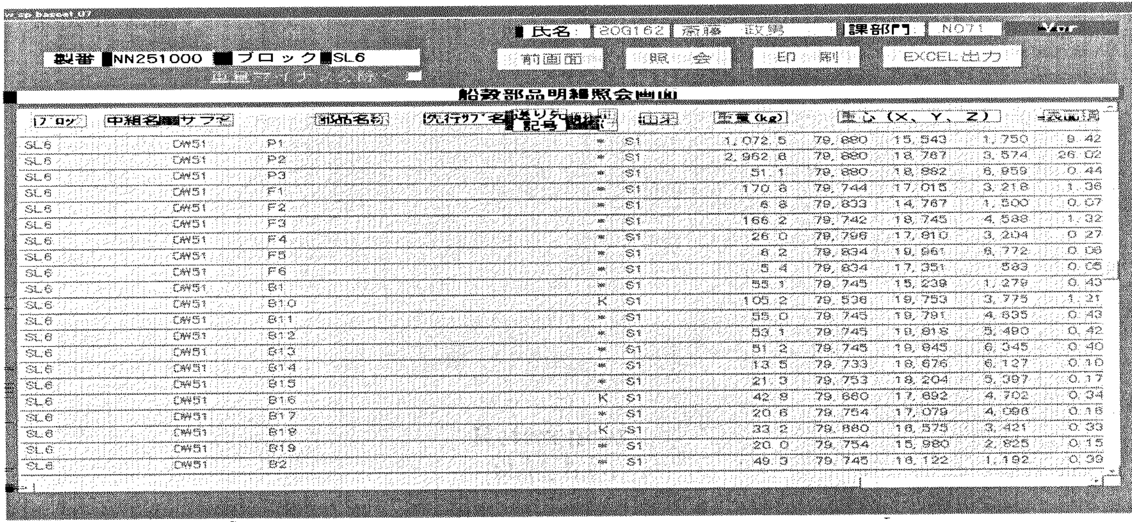


Figure 6. 3-D Assembly drawing



ブロック	中組名	サブ名	部品名称	先行サブ名	送り先記号		重量(kg)	重心 (X、	Y、	Z)	表面積
SL6		DW51	P1		*	S1	1,072.5	79,880	15,543	1,750	9.42
SL6		DW51	P2		*	S1	2,962.8	79,880	18,767	3,574	26.02
SL6		DW51	P3		*	S1	51.1	79,880	18,882	6,959	0.44
SL6		DW51	F1		*	S1	170.8	79,744	17,015	3,218	1.36
SL6		DW51	F2		*	S1	6.8	79,833	14,767	1,500	0.07
SL6		DW51	F3		*	S1	166.2	79,742	18,745	4,588	1.32
SL6		DW51	F4		*	S1	26.0	79,796	17,810	3,204	0.27
SL6		DW51	F5		*	S1	6.2	79,834	19,861	6,772	0.06
SL6		DW51	F6		*	S1	5.4	79,834	17,351	583	0.05
SL6		DW51	B1		*	S1	55.1	79,745	15,238	1,279	0.43
SL6		DW51	B10		K	S1	105.2	79,538	19,753	3,775	1.21
SL6		DW51	B11		*	S1	55.0	79,745	19,791	4,835	0.43
SL6		DW51	B12		*	S1	53.1	79,745	19,818	5,490	0.42
SL6		DW51	B13		*	S1	51.2	79,745	19,845	6,345	0.40
SL6		DW51	B14		*	S1	13.5	79,733	18,676	6,127	0.10
SL6		DW51	B15		*	S1	21.3	79,753	18,204	5,397	0.17
SL6		DW51	B16		K	S1	42.8	79,660	17,692	4,702	0.34
SL6		DW51	B17		*	S1	20.6	79,754	17,079	4,096	0.16
SL6		DW51	B18		K	S1	33.2	79,880	16,575	3,421	0.33
SL6		DW51	B19		*	S1	20.0	79,754	15,980	2,825	0.15
SL6		DW51	B2		*	S1	49.3	79,745	16,122	1,192	0.39

Figure 7. Parts list

Basic Production planning and scheduling

This module supports basic production planning for a ship. When using this system, design information not supplied by SUMIRE-H, F only supplied by IDOS system and empirical values. This system purpose as follows: principal milestone of production schedule (*block* erection schedule, start time of fabrication, main engine erection date, main inspection of every kind date, etc.), first stage receipt date of materials, fittings and equipment, basic production process each *block*.

This modules is consists of some sub-modules, which is *erection schedule, principal milestone schedule* and *macro-schedule.* Following as shows an example of those schedules.

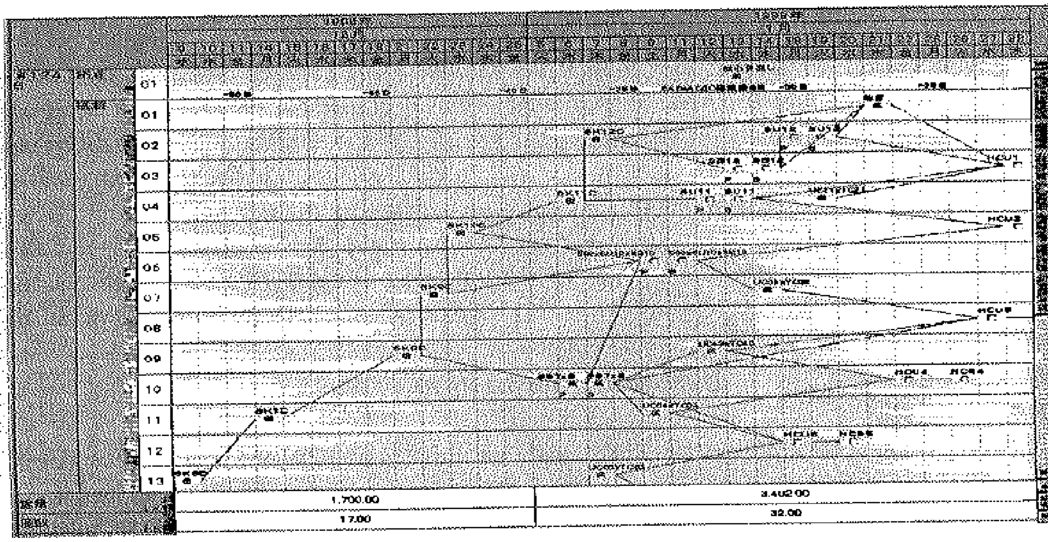


Figure 8. Erection schedule

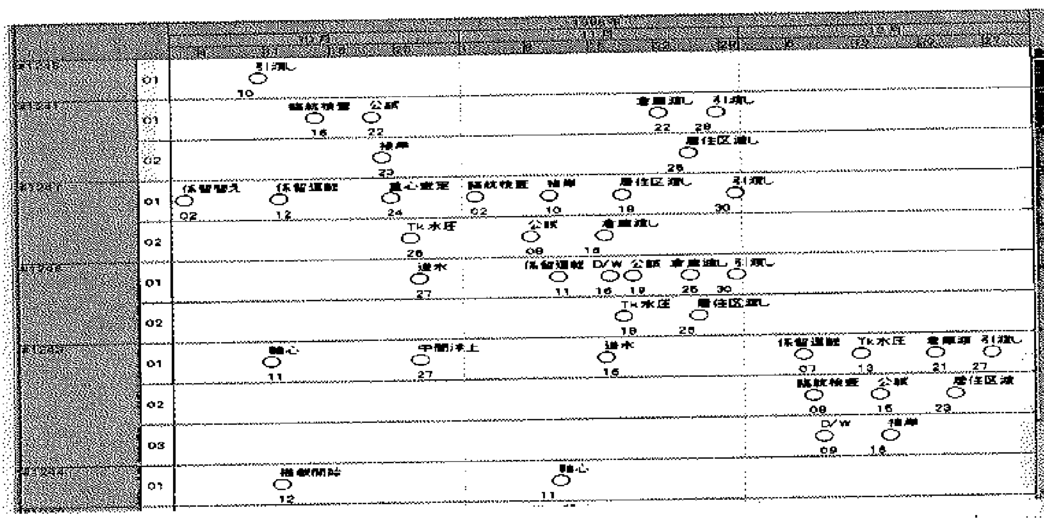


Figure 9. Principal milestone schedule

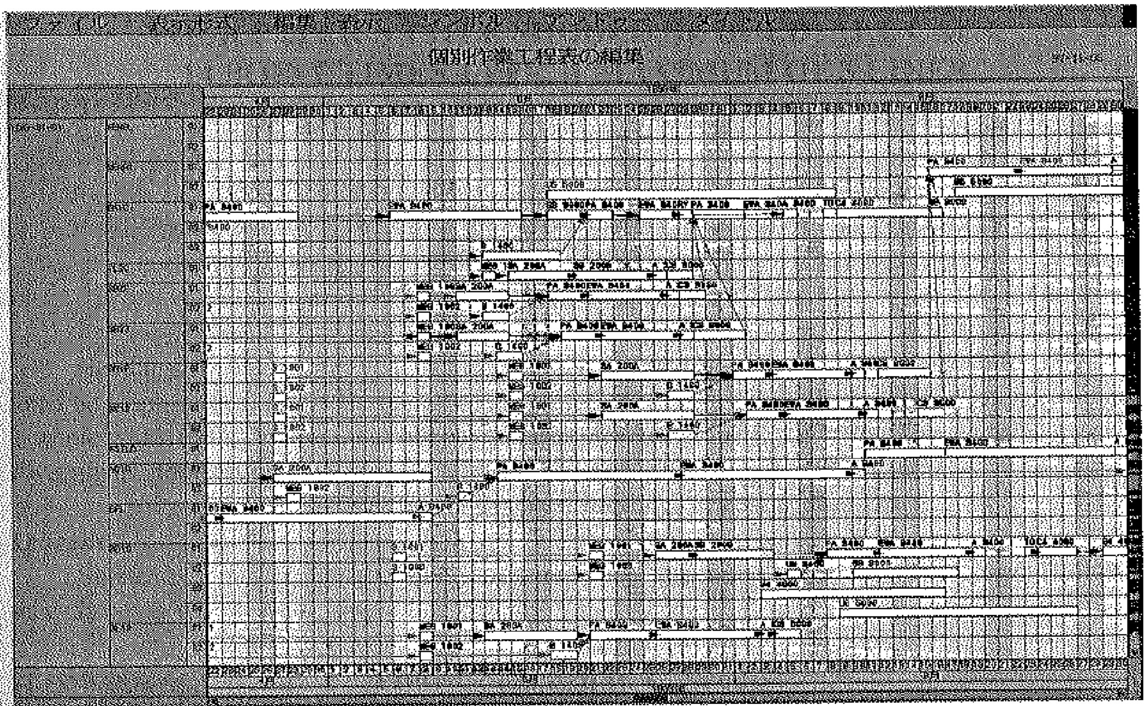


Figure 10. Macro-schedule of a ship

Production process

Production process of purpose is defined a *work unit* clearly, and other hand each *work unit* have to be related to parts data. Hence production process is concerned in the *interim-product* at Parts tree module. Basic planning time, the *interim-product* is equaled to *block*. While detail and actual planning time at each work stage, we must be defined the *interim-product* that is consists of a few parts comparison with *block.*

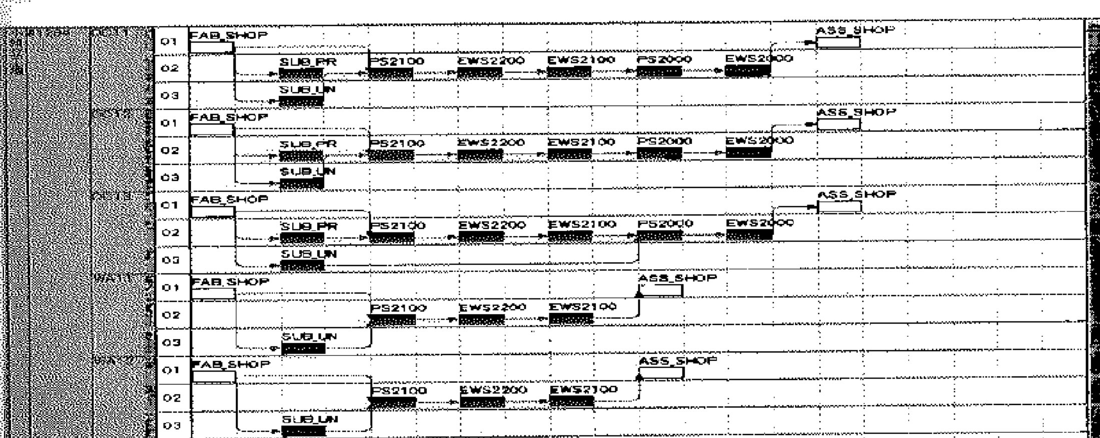


Figure 11. Network of Production Process

Planning and scheduling

This module supports actual production planning for each shop. When using this system, design information already supplied by SUMIRE-H, F. This module can be made schedule support that real a mount of parts, standard man-hour, etc. is computed at SUMIRE-H, F.

Let us assume that the production process and the part tree are defined on this level beforehand.

On the other hand, actual production planning is practiced under a production constraint that means as follows: amount of worker, restriction on production plant, principal milestones, date of the front and the rear *work process*, etc. PDOS is integrated all data on database. Consequently it can be check easy the schedule in charge of shop that is maintained proper under the constraint.

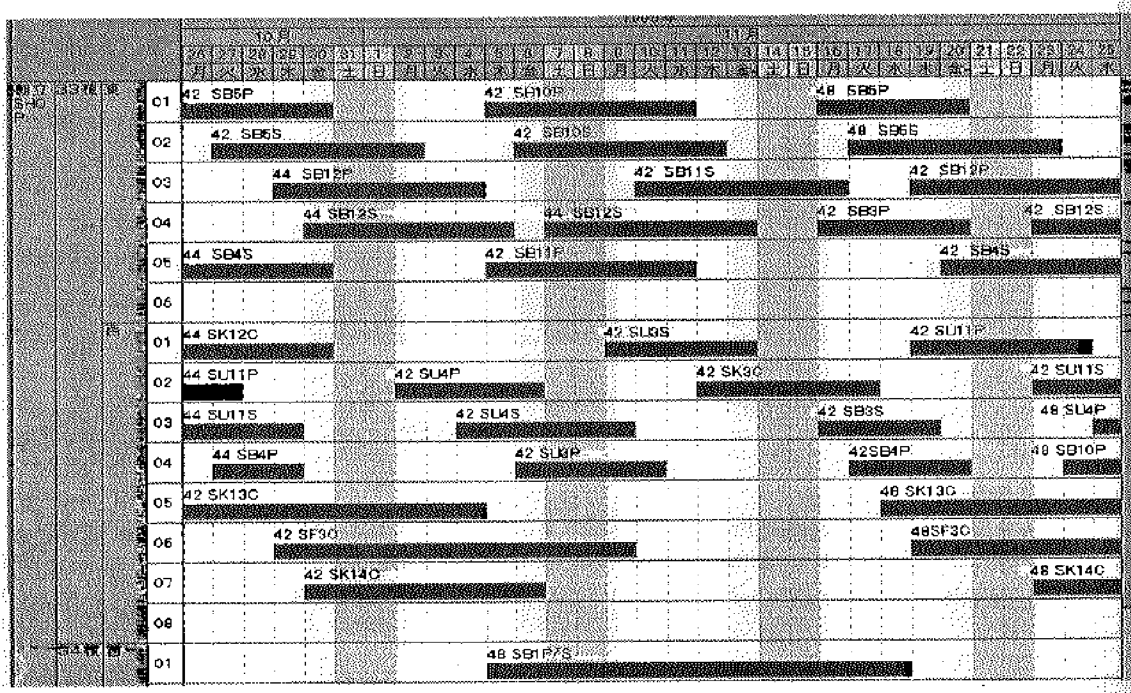


Figure 12. Scheduling for a shop

In a similar way, the system supports the scheduling of tact time of a conveyer line. It can be easy scheduling simulation automatically.

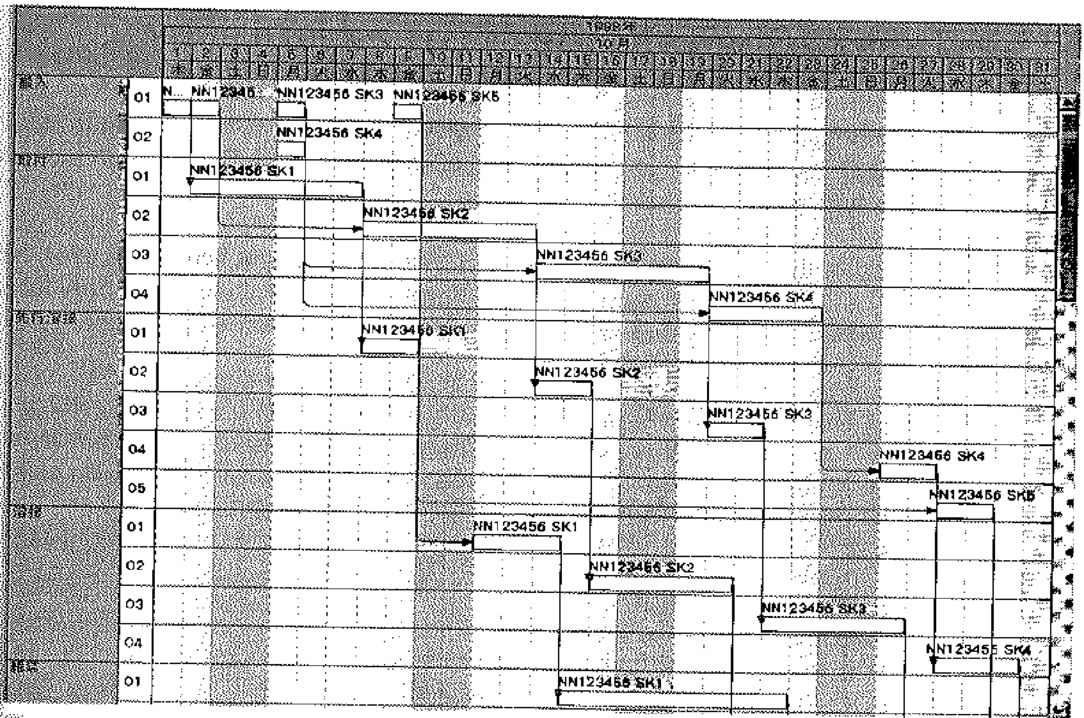


Figure 13. Scheduling for a conveyer line

Progress Control

After it makes scheduling for a shop, an instruction sheet at each work are supplied to the workers. Every day, all foremen input actual man-hours and progress of work of this instruction sheet. The system can be managed progress of work and a production index.

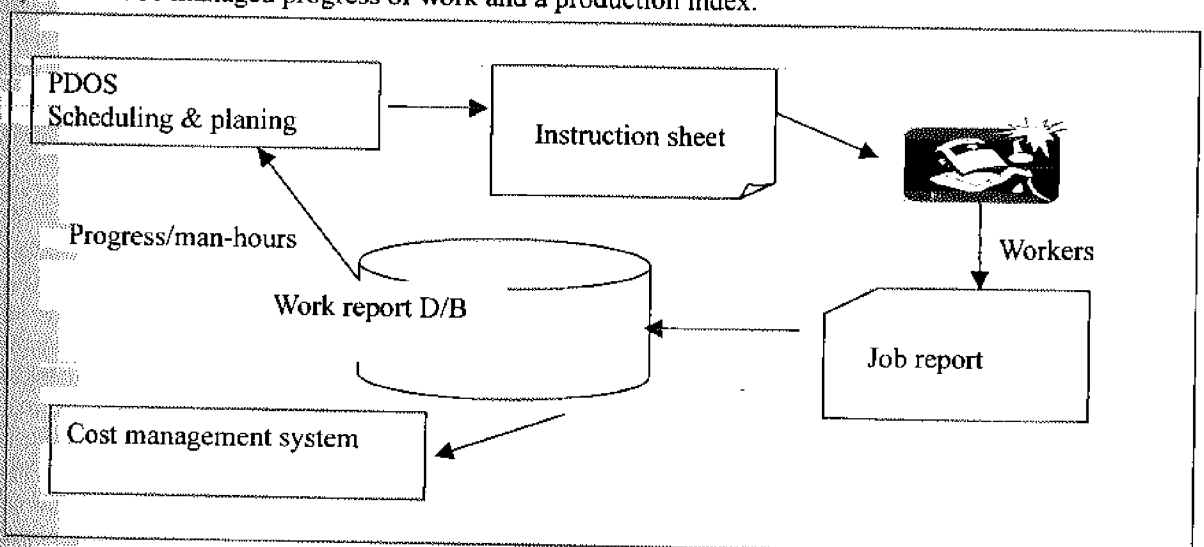


Figure 14. Outline of Progress Control

CONCLUSION

SHI's production information system "PDOS" has been introduced. PDOS has realized to support production information database. SUMIRE total-system are accelerating the further improvement of productivity in both design and production stage, and this system has been proceeding successfully with satisfactory results.

However, the author thinks that the development of PDOS has not yet been completed. Now the author thinks, that PDOS is not only caused the movement on the re-engineering of production process, but also have to be simulation and estimation system of total productivity at initial design stage. This systems' concept is shown in Fig.15.

Sumitomo Heavy Industries, Ltd. will continue to improve SUMIRE system.

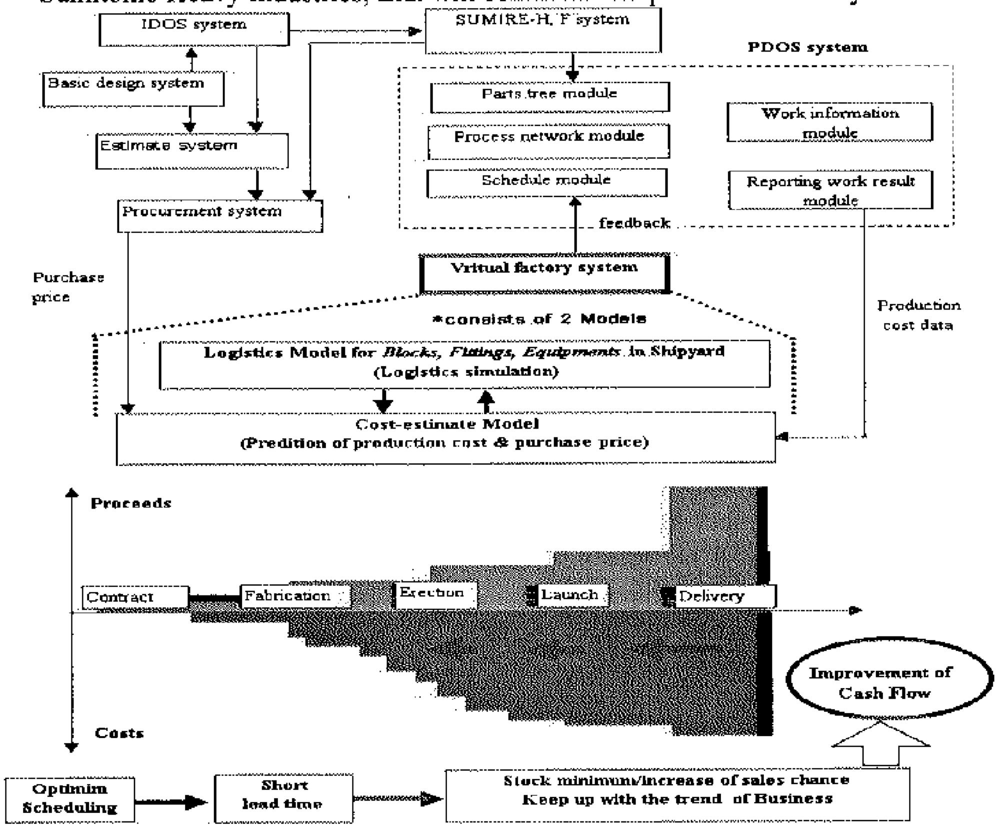


Figure 15. Concept of simulation & estimation system

Author would like to thank the assistance from CIM project that supported by the Nippon Foundation funded by motorboat racing revenues.

REFERENCES

1. Tanigawa, F., SUMIRE system in Sumitomo oppama shipyard, ICCAS, 1997.

OPERATIONAL CONSTRAINTS IN COMPUTER-AIDED SHIP DESIGN: MODELING THE REQUIRED "VIRTUAL SOLIDS"

*Nickolas S. Sapidis** & *Gabriel Theodosiou*
National Technical University of Athens, Athens, Greece

Abstract

Current Computer-Aided Design/Engineering (CAD/CAE) technology offers sophisticated *complete product models* for a ship's structure and systems (propulsion plant, electrical power generation, etc), yet it ignores the *free space* inside a vessel's compartments, the single most important commodity for every-day operations like cargo management, service and repair of machinery, etc. Thus, CAD/CAE systems offer not much help to the designer who must confirm that a particular configuration meets *operational constraints* related to standard operations that will be performed inside the vessel he/she designs. Virtual reality techniques cannot solve this problem as, at best, they only significantly enhance the visual perception of 3-D models. This paper demonstrates that operational constraints relate primarily to modeling and managing *virtual solids* representing "required free spaces" that must exist at specific locations inside the ship. Various instances of virtual solids related to engine-room layout are identified and a model appropriate for all of them is proposed. This model is based on state-of-the-art solid modeling techniques, thus it produces objects directly usable by a CAD system. Finally, geometric-modeling problems are identified related to efficient virtual-solid management/interrogation, and solutions are proposed. Implementations in the AutoCAD system are presented and examples are discussed.

1. Introduction

CAD/CAE software has reached a remarkable level of maturity offering robust techniques for defining and manipulating the complex 3-D objects involved in ship- and mechanical-design [11] [16] [20]. Sculptured surfaces, like a ship's hull and decks, are designed using the Surface Modeler of the CAD system, based on B-spline techniques that allow automation of many design steps [20]. The structural elements and mechanical subsystems of the ship are designed using the second major component of any CAD system, the Solid Modeler (SM) [4]. In order for the SM to support the many modeling- and analysis-operations involved in ship design, it offers a variety of representations for solids, ranging from exact (but complex) constructive solid geometry and boundary representation, to simple (but approximate) triangulation models and bounding boxes. *Constructive solid geometry* applies regularized set operations (regularized union $\bigcup^*$, intersection $\bigcap^*$, difference $-^*$, etc) to primitive, i.e., simple, solids to create complex objects [10]. *Regularization* (indicated by the superscript $*$) ensures that the result of the set operations applied on n-D solids is either an n-D object or the null set, i.e., regularization disallows the result to be an entity of lower dimensionality. On the other hand, *boundary representation (B-rep)* defines a solid by maintaining a mathematically complete description of its boundary, composed of vertices, edges, and faces [4]. *Sweeping* is often required by applications, refering to the solid produced by moving a 2-D or 3-D object G ("generator") along a *trajectory* $l = l(t), t \in [t_1, t_2]$ [20]. Thus, the mathematical definition of a "swept solid" is $S_N(G, l) = \bigcup_{t=t_1}^{t=t_2} G(t)$, where N is the dimensionality of G [18] [20]. Although many researchers have studied sweeping since the early days of CAD, still no satisfactory solution for the 3-D problem is available, and thus, only 2-D sweeping, $S_2(G, l)$ is available in the majority of current CAD systems [18]. The special case where the trajectory is a

*Also with: *Marine Technology Development Co., P. Faliro, Greece* straight-line segment (i.e., a vector) perpendicular to the plane of G is called an *extrusion*.

Approximate solid-models are used for accelerating graphics and interrogation operations. The most useful one is also the simplest one, the *bounding box*: given an object G and any coordinate system $<xyz>$, the bounding box BG is defined to be the smallest parallelepiped with faces parallel to $<xyz>$ such that $G \subset BG$. Bounding boxes are used by "preliminary tests" leading to cost-effective algorithms for solid-solid interference and other operations. Subdividing recursively the bounding box of a complex CAD model into subboxes produces another approximate solid-model, the *octree model* [1], which divides the given model into submodels corresponding to individual subboxes. Thus, a spatially specific operation need only consider appropriate submodels identified automatically using the octree model.

The present research deals with the application area of *"object layout"*, refering to design techniques for a highly complex scene with thousands of objects, like a ship's interior structure and engine-room [2] [3] [17]. The particular example that we will have in mind, in this paper, is a ship's engine-room (ER), which corresponds to the worst-case situation. Although the solid-modeling tools, briefly described above, improve continuously, this is not true for object-layout techniques, thus "objects are usually laid out ... with careful mouse manipulation, or by numerically specifying transformations ... This is a time-consuming and error-prone process" [17]. Published works are scarce (see references above) and have focused only on the *geometric consistency* of a layout, which relates to collision-avoidance and positional constraints. Our research considers also *operational constraints* related to operations like cargo management (for commercial ships), mission completion (for naval ships), emergency operations (fire extinction), and especially service and repair of mechanical systems, which is the subject of this paper. These operational constraints relate primarily to modeling *virtual solids* representing "free spaces" that must exist at specific locations inside the ship so that the work-steps implied by an operation may be performed [6]-[8]. Section 2 of this paper discusses CAD-modeling of free-space requirements for ship engines. Section 3 focuses on a vital subproblem of free-space modeling: simplification of planar domains. The complete Free-Space Model is described in Section 4, followed by introduction of the required CAD-process accelerators (for the combined Ship- and Free-Space-Models) in Section 5.

2. Free-Space Requirements in the Engine Room

While the layout of a ship's engine-room (ER) is primarily the shipyard's concern, certain demands made, e.g., by the engine builder must be considered to ensure that all mechanical systems can properly function. Important equipment, like pumps, coolers, and monitoring stations, should be readily accessible for servicing. Cleaning of cooler elements or filters, work on separators, emptying tanks must be possible at any time without hindrance. There should be sufficient free floor-area for temporary storage of pistons, timing shafts, exhaust turbochargers, and also for the maintenance personnel. Indeed, manufacturer's documentation ("project guides") and drawings always specify free-space requirements for dismantling a system and performing maintenance and/or parts-replacement operations [6]-[8].

The present research has focused on free-space requirements related to the ship's main engine(s) and auxiliary engines. Our study has considered manufacturers' specifications, literature and drawings for current systems as well as those available in the last ten years. The encountered descriptions of required Free Spaces (FSs) may be classified into three categories:

Case 1. Complete geometric definition of FS as a generalized cylinder of given *base* (a polygon, or more generally, an arbitrary planar domain) and *height/length*. *Example:* Fig. 1(A) presents the FS for maintenance of an engine: both the base and the length of the FS are fully defined by the engine manufacturer.

Case 2. Partial definition of FS by a "height/distance" from a solid's face or "principal plane". Here, the manufacturer's description is, in general, incomplete and must be complemented with additional information by the designer. *Example:* Fig. 1(B) (which was copied from [7]) shows that the space required for removal of piston/liner of the main engine is specified by the height **E** from the crankshaft centerline to the lower edge of the deck beam (above the engine).

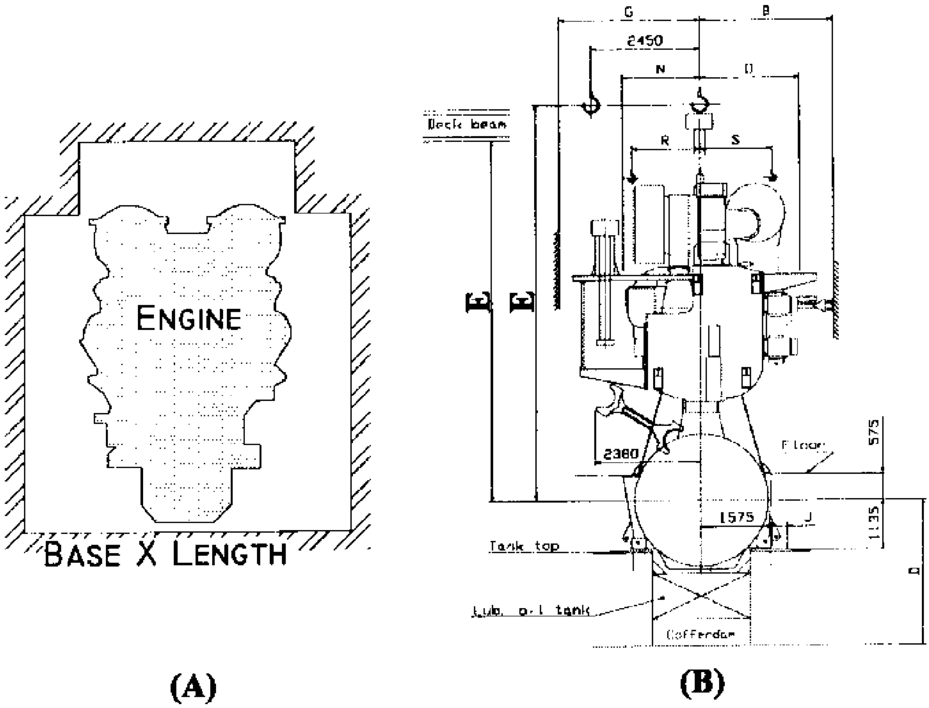


Figure 1: Required Free Space of Case 1 (A) and of Case 2 (B).

Case 3. Complete geometric definition of FS by specifying movement of a 2-D or 3-D object along a trajectory (which is a planar curve or polygon). *Example:* In Fig. 2, all dismantling spaces are defined as movements of the corresponding engine components along straight-line paths.

The manufacturers' FS requirements are implemented in a CAD system as 3-D "virtual solids" called Free Volumes (FVs), which the designer can model/interrogate using standard CAD techniques. User intervention is unavoidable for all cases of FSs, as explained below.

2.1. User Interaction in Free-Space Modeling

The designer must interact with the CAD system to complete the definitions of case-2 FSs. For this purpose, the system should make available to him/her: (a) appropriate 3D view(s) of the related part of the ER, and (b) sections of the objects, in that part of ER, where the sectioning plane is identical with the principal face or plane used in the definition of the FS. On these sections the designer will draw appropriate base(s) which in combination with the manufacturer's "height requirement" fully defines an FS.

The designer must also add to all FS definitions important nongeometric information addressing the issue that FSs are not always completely free of intrusions. Indeed, some of them are allowed to interfer with specific objects. For example, in Fig. 1(B) (see also case 2 above), the manufacturer's incomplete FS definition will be complemented by a base to fully define the required FS, which is clearly not completely free of interferences. At least the objects *main engine* and *crane hook* must be allowed to interfer with this particular FS, information that must be specified by the designer. This example makes also obvious that forcing all FSs to be completely

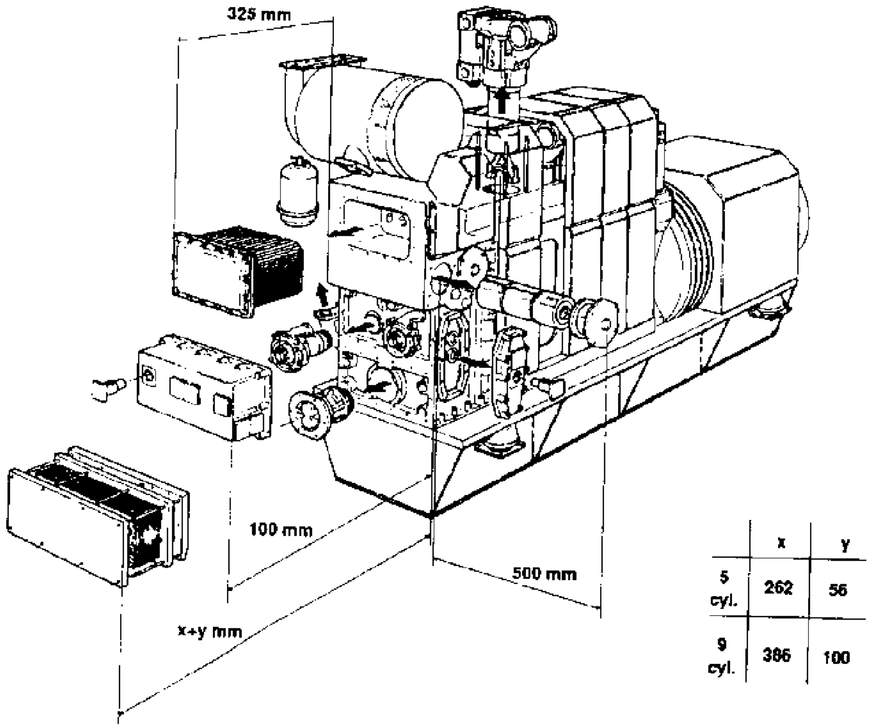


Figure 2: Required Free Spaces of Case 3 for the MAN B&W L16/24 Engine [6].

free of intrusions would require defining the corresponding virtual solids using a "general purpose" solid-model, like the B-rep, instead of the highly economical extrusion used below.

2.2. Free-Volume Definition in a Solid Modeling System

Since the free-volumes are 3-D geometric solids, the most appropriate component of a CAD system to define them is the the Solid Modeler (SM). Although SMs are designed to handle "real solids" like structural elements and mechanical components, their kernel techniques are purely geometrical and thus directly applicable also to free-volumes as virtual-solids, i.e., objects of zero mass-density.

It is straightforward that FVs of case 1 or 2 can be directly described as extruded solids, which is also the case for the FVs of case 3 defined by a moving 2-D object. FVs defined as solid-sweeps cannot be directly implemented in a CAD system, due to unavailability of a 3-D sweep operation [18]. Thus, we use the following procedure to approximate the solid-sweep $S_3(R, \mathbf{w})$ defining the solid produced by the solid R moved a vector $\mathbf{w} = AB$ (see Fig. 3):

Step A. A new coordinate system $c_w \equiv < Oxyw >$ is defined, which is compatible with $\mathbf{w}$ and has its origin O at the point A. The bounding box BR_w of R with respect to c_w is calculated (Fig. 3(B)).

Step B. The solid R is projected onto the plane xy using, e.g., the *project-onto-plane* tool employed also by the CAD-component creating standard engineering drawings. The exterior boundary of the projection is identified and transformed into a planar domain Δ_R (Fig. 3(B)).

Step C. If h_w is the height of BR_w (along the $\mathbf{w}$ direction), let $\mathbf{w}' = AB'$ be a vector parallel to $\mathbf{w}$ with length $|\mathbf{w}'| = |\mathbf{w}| + h_w$ (Fig. 3(C)). Then, the extrusion $S_2(\Delta_R, w')$ is obviously a superset of $S_3(R, \mathbf{w})$ and also a good approximation of that solid.

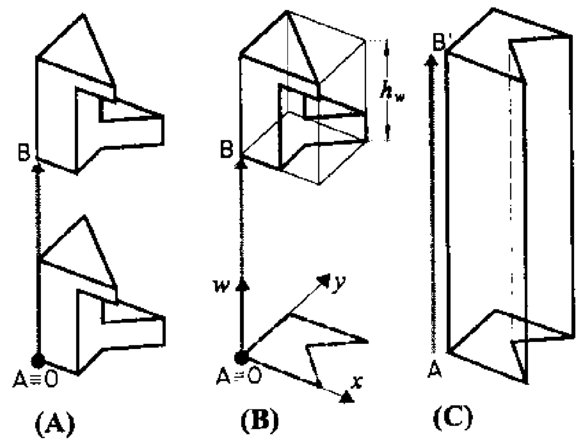


Figure 3: Approximating a Solid-Sweep with an Extrusion.

2.3. Approximate Free-Volumes for an Economical Free-Space Model

Even if one isolates the ship's engine room, the number of objects/parts involved in the model is extremely high, resulting in significant requirements on memory & disc space as well as on efficiency of modeling/interrogating algorithms and data-management techniques [9]. Modeling FVs as solids can significantly increase the number of objects in the CAD data-base, thus this proposal is viable only if FVs are guaranteed to be economical regarding required storage. Although the extrusion is a highly economical model, as it can be stored as a couple of { *generator*, *height* }, still it may produce an FV with a large number of faces when the *generator* has many edges (note that the number of faces in FV is equal to the number of edges in the generator plus two). Indeed, the generating planar-domain D often includes many small edges (especially for FSs of case 2 and 3) as D is obtained from a system-manufacturer's final, i.e., detailed CAD-model or drawings. In conclusion, efficient FV-modeling requires solution of the following *planar-domain simplification problem* (see Figs. 4 & 5).

Problem 1 *Let D be a domain in the plane, and $Cr(D, D^*)$ a criterion ensuring that the domain D^* is an acceptable approximation of D. Specify a new domain D^s, such that (i) D is a subset of D^s, which in turn is a subset of the convex hull of D, (ii) D^s satisfies the above criterion, and (iii) the number of edges in D^s is as small as possible.*

The next section clarifies statement of the above problem by introducing the necessary notation for describing planar-domains bounded by line-segments or convex-arcs. More complex domains have not been identified in our study of engine manufacturers' "free space requirements" [5] yet both the theory and the algorithm proposed below may easily be extended to the case of arbitrarily-shaped domains bounded by spline curves. Regarding the approximation criterion $Cr(D, D^*)$, it is obvious that the space occupied by $S_2(D^*, \mathbf{w})$ beyond the boundary of $S_2(D, \mathbf{w})$ should be minimized. Indeed, the criterion used below ensures that the additional occupied space is either not usable (for layout purposes) or too small. Finally, a robust simplification-algorithm is proposed and implemented in the AutoCAD system.

3. Planar-Domain Simplification

The planar domain D, to be simplified, is a two-dimensional (2-D) solid without holes, defined by a 2-D Boundary-representation, *B-rep(D)*, as specified by standard solid-modeling theory [4] [20]. B-rep(D) gives an informationally-complete description of the boundary of D, the "polyline" Λ consisting of a finite number of linear or convex edges. Λ is defined by its vertices $\{P_j;\ j = 0, 1, \ldots, N-1\}$ and edges $\{E_j;\ j = 0, 1, \ldots, N-1\}$, where each edge is defined parametrically $E_j = E_j(t), t \in [a_j, b_j]$, with $E_j(a_j) = P_j$ and $E_j(b_j) = P_{j+1}$ (Fig. 4). Vertices and

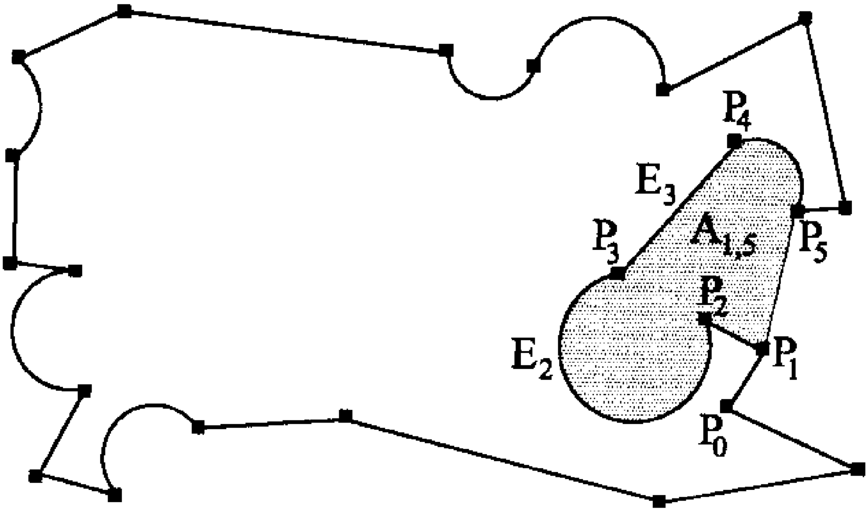


Figure 4: Planar Domain D Defined by a Single Polyline. P_1P_5 is a Fence of D.

edges are numbered sequentially corresponding to a counterclockwise (ccw) traversal of Λ. Every point $P \in \Lambda$ is associated to its *neighborhood* $\mathcal{N}_D(P)$ specifying where the interior of D, $Int(D)$, is with respect to Λ [10]. $\mathcal{N}_D(P)$ is defined to be equal to the intersection of a disk $\Delta(P, \rho)$, centered at P with radius $\rho \to 0$, with D. Current solid models offer very efficient representations for neighborhoods employing only tangent-vectors to the edges of D [10]. We complete the notation used here with $\mathcal{E}(P, Q)$, where P and Q are two points on Λ with P preceding Q (for a ccw traversal of Λ): $\mathcal{E}(P, Q)$ will denote the part, or *subpolyline*, of Λ that starts at P and ends at Q; each one of the points P, Q may be either a vertex or an "interior point" of an edge.

3.1. A Strategy and Criterion for Domain Simplification

In order for the simplified domain D^s (see Problem 1) to contain as few edges as possible, the simplification algorithm must employ only the following *simplification operation:*

(a) identify couples of vertices $\{P_k, P_l\}$ such that the line P_kP_l does not interfer with $Int(D)$ and approximates satisfactorily the subpolyline $\mathcal{E}(P_k, P_l)$, and

(b) replace $\mathcal{E}(P_k, P_l)$ by P_kP_l.

This requires introduction of the following (see Fig. 4):

Definition 1 *If P_k, P_l (with $k < l$) are vertices of D, the line-segment P_kP_l is called a fence of D if and only if P_kP_l does not intersect $Int(D)$. The subpolyline $\mathcal{E}(P_k, P_l) = E_k \cup E_{k+1} \cup \ldots \cup E_{l-1}$, which will be replaced by P_kP_l (if P_kP_l is included in a simplification), is called the f-subpolyline of P_kP_l.*

Regarding the criterion for acceptable approximation (see (a) above and Problem 1), this must refer to the "error volume" $S_2(D^s, \mathbf{w}) - S_2(D, \mathbf{w})$. Thus, for a single fence P_kP_l and the corresponding subpolyline $\mathcal{E}(P_k, P_l)$ the error is measured by a generalized-cylinder $W_{k,l}$ with basis $A_{k,l}$, defined by the closed polyline $\mathcal{E}(P_k, P_l) \cup P_lP_k$, and height $|\mathbf{w}|$. Aiming at minimizing loss of usable ER space, we propose:

Criterion 1 *The fence P_kP_l is a satisfactory approximation of $\mathcal{E}(P_k, P_l)$ if and only if*

$$\text{PRIMARY:} \quad \textit{the volume } Vol(W_{k,l}) \textit{ is not accessible,} \tag{1}$$

or

$$\text{SECONDARY:} \quad Vol(W_{k,l}) \textit{ is accessible but too small.} \tag{2}$$

The above two conditions should preferably be quantified in terms of quantities related to D only. As $Vol(W_{k,l})$ may be accessed only through the parallelogram generated by P_kP_l, we measure its accessibility by the length $|P_kP_l|$. Also, the straightforward choice to measure the size of $Vol(W_{k,l})$ is the area $|A_{k,l}|$ of its basis. Thus, an alternative statement for the above criterion is:

Criterion 2 *Given a domain D, a fence P_kP_l is considered to approximate satisfactorily D (i.e., the subpolyline $\mathcal{E}(P_k, P_l)$), if and only if*

$$\text{PRIMARY:} \quad |P_kP_l| \leq L_{small} \tag{3}$$

or

$$\text{SECONDARY:} \quad \textit{(3) is not true} \quad \text{and} \quad |A_{k,l}| \leq A_{small}, \tag{4}$$

where L_{small} and A_{small} define minimum passage-width and minimum usable area, respectively. Both limit-values A_{small} and L_{small} may be automatically set as follows: A_{small} is the ratio of the "smallest usable volume" (e.g. = "volume of smallest object in the ER") over the length $|\mathbf{w}|$, while L_{small} could be set equal, e.g., to the largest dimension of the smallest object in the ER.

Every fence satisfying Criterion 2 is characterized as a *candidate simplification edge* or *cs-edge* — the corresponding subpolyline $\mathcal{E}(P_k, P_l)$ is called a *cs-subpolyline* — and it may be included in a simplification of D. Criterion 2 does not suffice for the development of an optimal domain-simplification algorithm. One must further deal with the problem that often Criterion 2 will identify two or more cs-edges $\{P_iP_j, P_mP_n, P_qP_r, \ldots\}$ whose cs-subpolylines partially overlap. In this case, only one of these cs-edges will be included in the simplification, which requires answering the question: "which one of the cs-edges $\{P_kP_l, P_mP_n, P_qP_r, \ldots\}$ is the best simplification edge?" For this purpose, we associate to each cs-edge P_kP_l a measure of its appropriateness-for-simplification, the quantity $\sigma_{k,l}$:

$$\sigma_{k,l} = \begin{cases} \dfrac{|P_kP_l|}{L_{small}}, & \text{if } \dfrac{|P_kP_l|}{L_{small}} < 1, \\ 1 + \dfrac{|A_{k,l}|}{A_{small}}, & \text{otherwise.} \end{cases} \tag{5}$$

The quantity $\sigma_{k,l}$ may be used for many purposes, e.g., to differentiate between cs-edges and other edges, yet, its principal purpose is to differentiate among cs-edges: clearly, the smaller $\sigma_{k,l}$ is, the better P_kP_l fits Criterion 1. The above discussion is summarized by the following criterion, which characterizes simplifications produced by the algorithm proposed below.

Criterion 3 *Given a domain D, a cs-edge P_kP_l is a simplification-edge (or s-edge) of D if and only if for every other cs-edge P_mP_n with $\mathcal{E}(P_m, P_n)$ overlapping nontrivially with $\mathcal{E}(P_k, P_l)$ either the number of edges in $\mathcal{E}(P_m, P_n)$ is larger than the number of edges in $\mathcal{E}(P_k, P_l)$, or $\sigma_{m,n} > \sigma_{k,l}$.*

Definition 2 *If Λ is the boundary of D, a simplification Λ^s is a closed polyline, each edge of which is either identical with an edge of Λ, called unsimplified edge or u-edge, or an s-edge of Λ.*

3.2. The Domain Simplification Algorithm

The algorithm described below accepts as input B-rep(D), i.e., the N edges of Λ, along with L_{small}, A_{small}. The output of the algorithm's main part (Steps A and B) is again Λ, but with some of its subpolylines marked as *"simplification subpolylines"* or *s-subpolylines*. The B-rep definition of the simplification D^s is built in Step C. where s-subpolylines are replaced by the corresponding straight-line edges.

Algorithm SIMPLIFY_BOUNDARY(Input: Λ, L_{small}, A_{small}: Output: Λ^s)

STEP **A** Mark all edges as u-edges.

STEP **B** For $i = N,\ N-1,\ N-2,\ \ldots,\ 2$:

{

B.1. Consider the set $\mathcal{UE}$ of all subpolylines of Λ with i consecutive u-edges

B.2. If $\mathcal{UE}$ is empty, then *goto*_CONTINUE, else:

B.3. For each subpolyline $\mathcal{E}(P_j, P_{j+i})$ in $\mathcal{UE}$: If P_jP_{j+i} fits Criterion 2, then mark P_jP_{j+i} (resp., $\mathcal{E}(P_j, P_{j+i})$) as cs-edge (resp., cs-subpolyline).

B.4. If no cs-edge is found, then *goto*_CONTINUE, else:

B.5. (B.5a) Among all cs-edges: find the cs-edges that fit Criterion 3.

(B.5b) Mark these edges (resp. subpolylines) as s-edges (resp. s-subpolylines).

(B.5c) For all remaining cs-subpolylines: mark again their edges as u-edges.

(B.5d) Empty the sets of cs-edges and cs-subpolylines.

CONTINUE

}

STEP C Create the B-rep of the simplification Λ^s as follows:

C.1. For each u-edge E_j: transfer its the B-rep description from B-rep(D) to B-rep(D^s).

C.2. For each s-subpolyline $\mathcal{E}(P_q, P_r)$: add to B-rep(D^s) the new straight-line edge P_qP_r.

C.3. Identify straight-line edges in B-rep(D^s) that are *consequtive* and *collinear* and merge them into a single edge.

C.4. Complete B-rep(Λ^s) by specifying neighborhoods of new linear edges.

End_Algorithm

Notes. *About Step B.3:* Implementing Definition 1 as given above would result in an inefficient algorithm involving many applications of the *Line-Domain*-intersection $P_kP_l \cap Int(D)$ operation. [15] describes an alternative implementation of Definition 1 that determines if P_kP_l intersects D, by classifying P_kP_l with respect to the neighborhoods of the two vertices P_k and P_l, which is a very cost-effective solution involving only two vector inner-products.

About Step B.5: This is performed as follows. First, all cs-subpolylines are ordered in a list according to an increasing $\sigma_{i,j}$ value. Then, the following iterative process is performed: Consider the next cs-subpolyline $\mathcal{E}(P_q, P_r)$ on the list, with $\sigma_{q,r} \leq 2$ (in the first iteration, this is the first item on the list). Mark $\mathcal{E}(P_q, P_r)$ as an s-subpolyline and, in the remainder of the list, find all cs-subpolylines partially overlapping with $\mathcal{E}(P_q, P_r)$ and unmark them.

About Step C.4: The neighborhoods of the new edges are obtained from the neighborhoods of the corresponding s-subpolylines using the *Neighborhood Information Transfer* [12].

3.3. An Example of Domain Simplification

The SIMPLIFY_BOUNDARY algorithm has been implemented in the current AutoCAD system and in particular in ObjectARX, the C++ object-oriented programming environment where also AutoCAD is developed. Implementation of the algorithm of Section 3.2 requires standard solid modeling functions/tools readily available either in core AutoCAD libraries or in ACIS, which is used by AutoCAD as its geometric-modeling kernel. The current implementation employs a number of acceleration techniques which are detailed in [15]. The hardware platform is a standard PC: Pentium(166 MHz) with 64 MB RAM, 256 KB Cache.

Fig. 5 depicts two different simplifications (bold lines) produced for a domain (fine line) with 47 edges. The finer simplification (Fig. 5(left)) corresponds to smaller values of the quantities L_{small}, A_{small}. This simplification has 18 edges and required a total CPU time of 1.36 secs. The

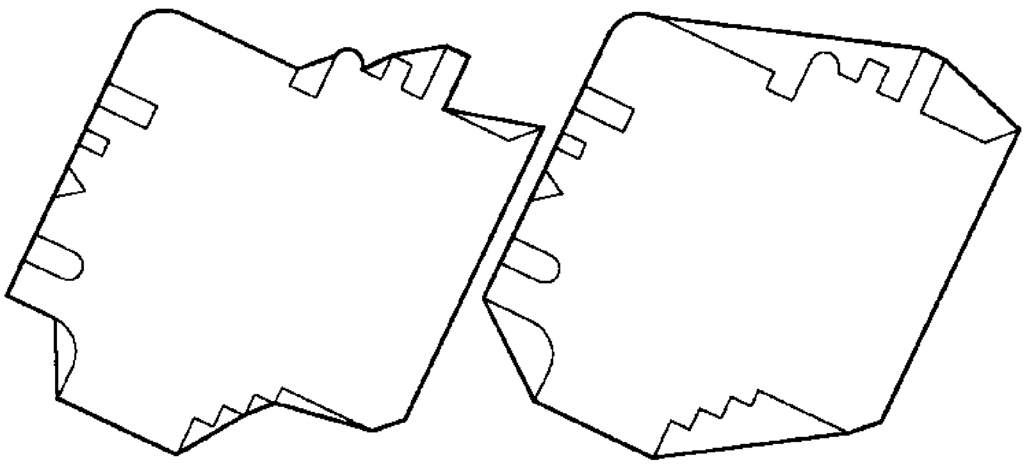


Figure 5: Example of Domain Simplification.

coarser one (Fig. 5(right)) has 10 edges and required a total CPU time of 0.55 secs.

4. Free Space Model: The Data Structure

The main purpose of the Free Space Model (FSM) is to collect and efficiently represent all Free Spaces required by an ER configuration. This model will be built in-parallel with the ship's 3-D CAD model and will require a varying amount of user interaction, which depends on the degree of completeness of the system-manufacturers' free-space specifications as well as on the designer's willingness to get involved in detailed free-space modeling. As far as we are concerned, FSM is a vital addition to the ship's 3-D CAD models to be used/updated throughout the ship's life-cycle to ensure that service and repair work can be performed even after, e.g., the structure of the vessel is modified or some mechanical systems are added/replaced.

Having these in mind, we propose a user-friendly data structure for FSM, where the main component is an *array of tasks*: see Fig. 6. Each element of this array is a *task*, well-defined by current ship technology, e.g., "main engine: piston removal", "auxiliary engine: pulling the charge air cooler element", etc. Each task carries a pointer to the ER *part(s)* it refers to. It is noted that as each part may point to more than one tasks, also each task must be allowed to refer to more than one parts. This is required, e.g., by a configuration where the task "maintenance of auxiliary engines" must point to two parts (auxiliary engines). This is also reflected in the example of Fig. 6, where Task-3 points to Part-1 and Part-2.

Although the strict majority of tasks can be performed in exactly one way, for few of them some manufacturers offer two or three alternatives. Typical example is the task "main engine: piston removal"; see, e.g., alternative procedures proposed in [7], which are reflected in the alternative values imposed on the distance E in the related drawing (copied in Fig. 2(B)). As a designer might want to record (e.g., for future reference) in FSM all these alternatives, we associate each task to a list of alternative *procedures* that can serve this particular task; e.g., in Fig. 6, Task-1 is associated to three procedures and Task-2 to two. Each procedure corresponds to a list of FVs all of which are required for implementing this procedure. (Again, we note that although most procedures correspond to one FV, there are examples, like the task/procedure "maintenance work on main engine", where a procedure corresponds to more than one FVs.) Since some procedures are possible only after one or more other procedures are performed, we must add to the "procedure" data structure also two kinds of pointers: incoming pointers from

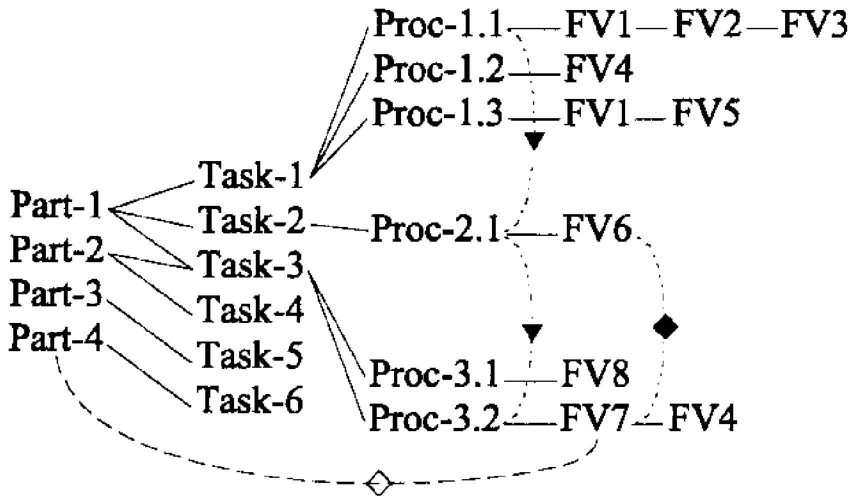


Figure 6: Free Space Model: the Data Structure.

other procedures that are required for the present procedure to be applicable, and outgoing pointers to procedures presuming the present one: e.g., in Fig. 6, Procedure 2.1 carries two such pointers.

Regarding the data structure describing each FV, we note that, in addition to a pointer to a "virtual solid" in the CAD data-base, *not shown* in Fig. 6, this must include pointers to parts that are allowed to interfere with the FV. Usually, interference is allowed only for very few parts like the part served by the procedure employing the FV; see § 2.1. Finally, the FSM must record exceptions to the rule *"each FV may be intersected by any other FV"* which generally holds in ER layout. For example, for procedures that must be executed always simultaneously, it may be necessary to ensure that the corresponding FVs do not intersect (another case where user interaction is unavoidable). This purpose is served by pointers added to each FV and pointing to FVs that must not intersect it. In Fig. 6, FV7 carries both a "part-interference" and an "FV-non-interference" pointer (marked with a white and black diamond, respectively).

5. Accelerating Model Interrogation Using Spatial Decompositions

Introduction of Free Volumes further increases the already huge number of objects/parts involved in a ship model, necessitating the development of sophisticated acceleration techniques for model interrogation operations and especially for *fast collision detection (FCD)*. A part's bounding box is the standard tool for accelerating collision detection. Beyond that, numerous techniques have appeared in the CAD and computer-graphics literature aiming at the above objective [1]. During the last few years, the problem of FCD drew increased interest from many researchers due to its relevance to new technologies like virtual reality and scientific visualization (see, e.g., [1], [19] and references therein). Unfortunately, these new results are not directly usable for accelerating CAD-operations as they deal only with triangulated models as opposed to complete product models based on exact solid modeling. Our literature search has concluded that, especially for the problem of ship-model management and interrogation, the classical model of recursive spatial decomposition (RSD) using octrees [1] [14] [19] continues to be the best available tool to accelerate operations like collision detection. Here are the major reasons:

- RSD is easily understood by non CAD-experts, not hard to implement, and very flexible to adapt to different CAD architectures.

- RSD is independent of the particular CAD-model and application, and it is able to transform into "local" even the most complex "global" problems.
- RSD offers, in addition to a secondary CAD model for the ship structure & layout, also a geometric model for the free spaces in the ship's compartments, that may be useful also for other purposes like cargo management, etc.
- RSD's only major disadvantage, dependance on the coordinate-system used (which creates problems when RSD is applied in mechanical CAD), is *"eliminated"* in the field of computer-aided ship design as ship structures and modules have unique "natural coordinate systems".

5.1. Spatial Decomposition Using an Octree Model

The first part of our proposal consists of constructing the following variant of the standard Octree Model: Let S be the complete model of the engine-room (or of the whole ship) incorporating CAD-models of structures, assemblies of mechanical systems, as well as the corresponding free-space model. The bounding box of S is translated into a parallelepiped Π defined as a solid. This is subdivided into eight equal parallelepipeds (octants) $\Pi_1, \Pi_2, \ldots, \Pi_8$, by planes perpendicular to Π's faces. Each Π_i is "solidified" and intersected with S_{i-1}, where S_{i-1} is obtained from S by removing those subassemblies, parts and FVs identified to lie wholly inside one (or some) of $\Pi_1, \Pi_2, \ldots, \Pi_{i-1}$ ($S_0 := S$). Each Π_i that intersects "too many" elements of S_{i-1} is further subdivided into octants, to which the above process is applied; see Fig. 7. The result is the

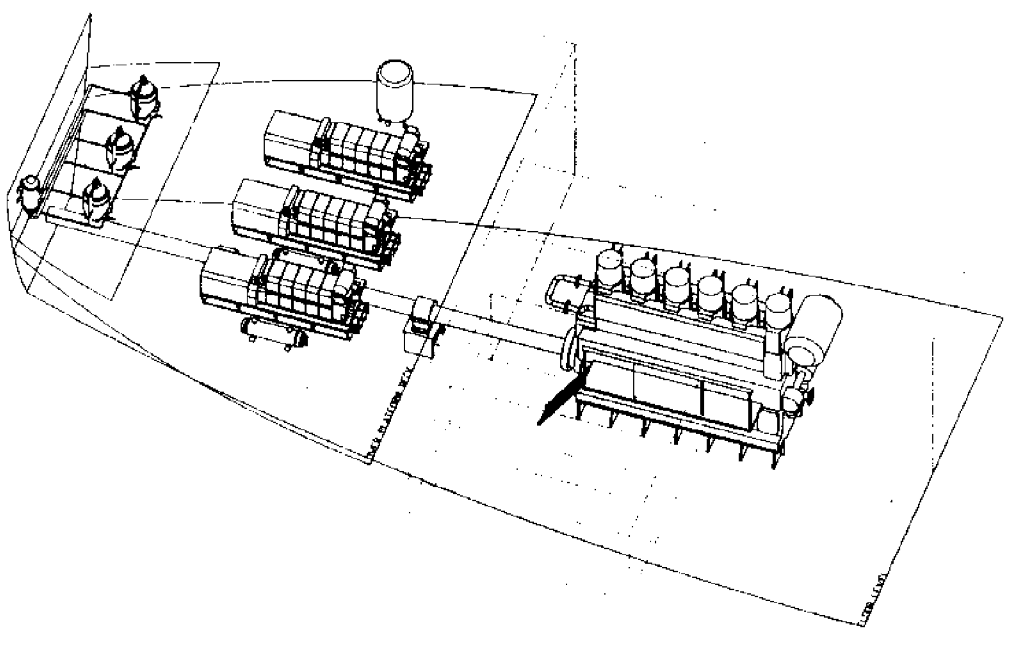


Figure 7: Constructing an Octree-Model for an Engine Room.

Octree OS, i.e., a recursive decomposition of Π, where each octant carries complete information regarding objects inside it, objects intersecting each face of it, etc [14]. It is emphasized that the geometric structures required for building OS, in particular the solidified octants, are removed from the CAD data-base after OS is completed. Thus, OS is a highly economical structure, where each octant is fully described by a single integer number, the octant's i.d. – simple formulas produce from this i.d. all geometric and topological properties of the octant, including the i.d.s of its neighbors, parent, etc — and pointers to the objects it includes/intersects.

Having completed the construction of OS, introduction of a new object E is followed by *classification* [13] of E with respect to OS, i.e., identification of a minimal set of octants completely enclosing E. Thus, detection of a possible collision, between E and S, is performed by considering only the small number of objects interfering with the octants identified above.

5.2. Improved Bounding Volumes Using an Octree

Availability of the Octree OS and of the related tools does not completely solve the problem of FCD, as even a single application of the Solid-Solid Intersection procedure SSI($solid_1, solid_2$) may be very costly. This is due to the high complexity of many ER solid-models (in terms of the volume of data stored), as they often fully-describe even minor shape details of parts, in accordance to current trends towards complete product models. Thus, it is vital to minimize applications of SSI using preliminary tests that rapidly identify nonintersecting solids by verifying that no overlap exists between "bounding volumes" completely enclosing respective solids.

Standard practice in current CAD systems is to use as bounding volume of a 3-D part P the bounding box BP, always included in the solid-modeling description of P. This is often inefficient as BP is generally a poor approximation of a complex part P. In view of the available octree-creation procedures described above, it is straightforward to see that a much better alternative is the union of a small number of non-overlapping "bounding octants", $O_j^P, j = 1, 2, \ldots, q$, with $\bigcup_{j=1}^{q} O_j^P \supset P$, constructed by applying the procedure of §5.1 to BP; see Fig. 8. This is done

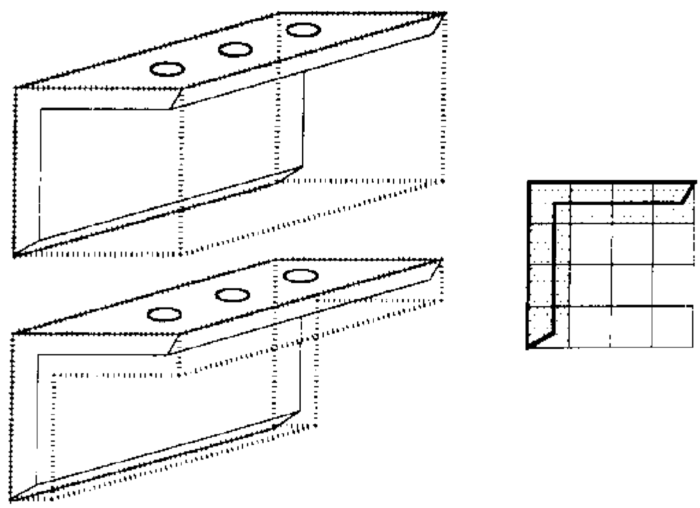


Figure 8: Extracting an Improved Bounding Volume from the Octree Model.

only for a solid P not well approximated by BP according to some criterion: the fact that SMs offer efficient calculation of parts' mass properties allows us to to measure at minimal cost the approximation error between P and BP using the quantity $Volume(BP) - Volume(P)$. We conclude this section by mentioning a variation of the above proposal: construct "bounding octants" of P by continuing subdivision of the ER octree-model OS only at the terminal octant containing P. Although plausible, this approach has two important disadvantages: it requires more memory and always produces significantly inferior "bounding octant" approximations.

6. Implementation of Free-Space and Octree Models in AutoCAD

This research is currently implemented in the ObjectARX environment of AutoCAD using the C++ language and the ACIS geometric-modeling kernel. It is noted that some form of Octree model has been already available in AutoCAD, yet this is used only for display acceleration and it is not available to users/developers as secondary geometric model. Fig. 9 depicts an example of an auxiliary-engine with free volumes: this is a solid model of the MAN B&W L16/24 engine

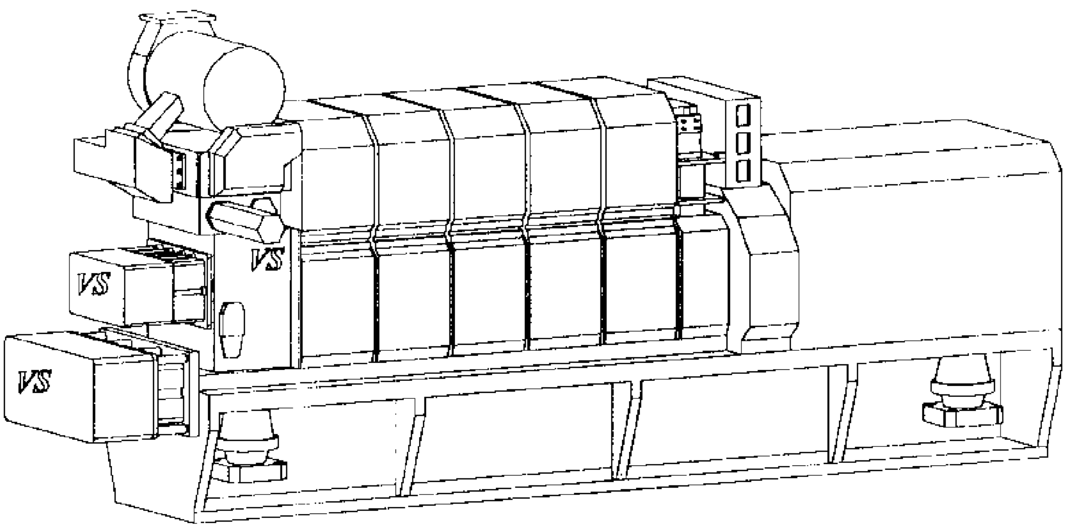


Figure 9: Solid Model of the MAN B&W L16/24 Engine Equipped with Three Free Volumes.

created by the authors on the basis of drawings provided by MAN B & W. Three of this engine's FVs (see also Fig. 2) are shown as virtual solids marked with "VS".

7. Conclusion & Comparison to Current Virtual Reality Systems

Existing Virtual Reality systems use CAD models of "real objects", like the structure and machinery in a ship, and attempt to "insert" the designer into the *CAD world*, so that he/she can inspect the models and virtually perform tasks. Surely, such techniques are beneficial as they improve visual perception of complex 3-D models, yet they offer no robust tools to the designer who must verify that a model meets its engineering and operational constraints.

This paper offers an alternative approach where tasks and constraints are transformed into *required free volumes*, which are inserted into the *CAD world* as "virtual solids". These virtual solids may be used either in conventional CAD systems or VR-enhanced systems. The vital improvement is that required free spaces are defined explicitly in the CAD model as solids (: called "virtual" but with only one difference, compared to other solids, that they have zero mass-density) coexisting with the "real" surfaces and solids of the ship. Thus, one can develop algorithms/tools to automatically verify, e.g., feasibility of maintenance operations, which form a concrete foundation for the designer to base his design-validation work.

Acknowledgements

This work was funded by the Hellenic Ministry of Development (PENED'95 research grant 880, awarded to Prof. P. Kaklis), and the Department of Naval Architecture & Marine Engineering and the Ship Design Laboratory of the National Technical University of Athens (NTUA). We thank Prof. H. Nowacki (TU Berlin) and Profs. J. Ioannidis, A. Papanikolaou, N. Kyrtatos, and P. Kaklis (NTUA) for their comments and suggestions. We owe special thanks to Mr. C. Ventouris (industry consultant) for his contribution and advice, to Messrs. S. Buchwald & M. Zaika (Applicon, GmbH) for suggestions and industrial data (including the model of the main engine in Fig. 7), to Messrs. J. Tzagarakis & Th. Papakonstantinou (Hellenic Shipyards, Co) for advice and technical information, and to MAN B&W Diesel A/S for technical information and for allowing us to include in this paper material from their publications.

References

[1] Barequet, G., Chazelle, B., Guibas, L., Mitchell, J., Tal, A. (1996), "BOXTREE: A Hierarchical Representation for Surfaces in 3D", *Computer Graphics Forum*, Vol. 15, pp. 387-396.

[2] van Emmerik, M., Rappoport, A., Rossignac, J. (1993), "Simplifying Interactive Design of Solid Models: a Hypertext Approach", *The Visual Computer*, Vol. 9, pp. 239-254.

[3] Fujita, K., Akagi, S., Nakatogawa, T. (1994), "Hybrid Approach to Plant Layout Design Using Constraint-Directed Search and an Optimization Technique", *Journal of Mechanical Design*, Vol. 116, pp. 1026-1033.

[4] Hoffmann, C. (1989), **Geometric and Solid Modeling**, Morgan Kaufmann.

[5] Kaklis, P., Sapidis, N. (1998), "Development of a Hybrid CAD System for Designing Surfaces and Solids: Application in Naval Architecture and Mechanical Design", Annual Progress Report, Depart. of Naval Architecture & Marine Engineering, NTUA.

[6] MAN B&W Diesel A/S, Holeby (1997), *Project Guide for Marine GenSets L16/24* (Ed. 03).

[7] MAN B&W Diesel A/S (1998), *Project Guide for the S60MC-C Engine*, 1996 (1st Ed.). Updates: http://www.manbw.dk.

[8] MAN B&W Diesel A/S (1998), *Programme Booklet for MAN B&W Diesel Engines*.

[9] Nilsson, P.-O. (1998), "Inside TRIBON 4 - Technical Features: High Performance Large Assembly Techniques", *KCS NEWSLETTER*, No. 35 (September 1998), pp. 2-3.

[10] Requicha, A.A.G., Voelcker, H.B. (1985), "Boolean Operations in Solid Modeling: Boundary Evaluation and Merging Algorithms", *Proceedings of the IEEE*, Vol. 73, pp. 30-44.

[11] The Royal Institution of Naval Architects (1995), **Proceedings of 'Computer Aided Design & Production for Small Craft' (CADAP'95)**, Southampton.

[12] Sapidis, N., Perucchio, R. (1991), "Domain Delaunay Tetrahedrization of Solid Models", *Intern. Journal of Computational Geometry & Applications*, Vol. 1, pp. 299-325.

[13] Sapidis, N., Perucchio, R. (1992), "Solid/Solid Classification Operations for Recursive Spatial Decomposition and Domain Triangulation of Solid Models", *Computer-Aided Design*, Vol. 24, pp. 517-529.

[14] Sapidis, N., Perucchio, R. (1993), "Combining Recursive Spatial Decompositions and Domain Delaunay Tetrahedrizations for Meshing Arbitrarily Shaped Curved Solid Models", *Computer Methods in Applied Mechanics and Engineering*, Vol. 108, pp. 281-302.

[15] Sapidis, N., Theodosiou, G. (1998), "Planar Domain Simplification for Virtual-Solid Modeling", submitted for publication.

[16] Shin, Y., Han, S.-H. (1998), "Data Enhancement for Sharing of Ship Design Models", *Computer Aided Design*, Vol. 30, pp. 931-941.

[17] Shinya, M., Forgue, M.-C. (1995), "Laying Out Objects with Geometric and Physical Constraints", *The Visual Computer*, Vol. 11, pp. 188-201.

[18] Sourin, A., Pasko, A. (1996), "Function Representation for Sweeping by a Moving Solid", *IEEE Trans. Visualization & Computer Graphics*, Vol. 2, pp. 11-18.

[19] Whang, K.-Y. et al. (1995), "Octree-R: An Adaptive Octree for Efficient Ray Tracing", *IEEE Trans. Visualization & Computer Graphics*, Vol. 1, pp. 343-349.

[20] Zeid, I. (1991), **CAD/CAM Theory and Practice**, McGraw-Hill.

ON APPLICATION CALS AND CAD/CAM SYSTEMS IN RUSSIAN SHIPBUILDING

Alexander N. Sitnikov, Alexander M. Plotnikov,
CRIST - Central Research Institute of Shipbuilding Technology,
St. Petersburg, Russia

Introduction

The Russian shipbuilding (design bureaus and shipyards) employs dedicated shipbuilding systems Foran (Spain), Tribon (Sweden), the Russian system «RITM-Sudno» (CRIST - Central Research Institute of Shipbuilding Technology, St. Petersburg), and universal systems Catia (IBM), Pro/Engineer (Parametric Technology), EDS Unigraphics, CADDS 5 (Computervision) etc.

The relevant issue today is that of system application of CAD/CAM to build a full-fledged electronic model for ship, to support this model at all stages of ship's life cycle, and to transfer the relevant information to other systems combined in one corporate system of an enterprise.

The corporate system is created by employing contemporary automated systems (CAD/CAM – Computer Aided Design/Computer Aided Manufacturing, PDM-Product Data Managing, PM -- Project Management, ERP/MRP II - Enterprise Resource Planning/Manufacturing Resource Planning) combined in one system on the basis of the contemporary information technology CALS and corporate computation network of shipyard.

CALS Technology or Ship Life Cycle Support

One way to raise the efficiency in the Russian shipbuilding is to employ contemporary information technologies in order to integrate the processes that are implemented during the ship's life cycle in terms of CALS-technologies (Continuous Acquisition and Life cycle Support).

Employment of CALS in the shipyard environment makes it possible:

- to expand the shipyard market niche and reduce the design and building time through cooperation with partners in real time processing at the stages of ship's life cycle. In fact, contemporary telecommunications have already made irrelevant the geographic location of partners. New opportunities of information interaction allow for cooperation in terms of virtual enterprises operating during the ship's life cycle. Cooperation becomes possible not only at the level of finished components – completing materials and articles, but also at the level of separate stages and tasks in the processes of designing, manufacturing and operation;

- to raise the efficiency of business-processes through integration of information and reduction in costs for paper document circulation, re-entry and processing of data, to ensure continuity of efforts in complex designs, such as a ship's design, and to allow for changes in the makeup of parties without losing the results that have already been obtained;

• to raise the attractive and competitive capacity of ships that have been designed and built in the integrated environment by employing contemporary computer technologies and are provided with information support facilities at the operation stage;

• to ensure the specified quality of products in the integrated system of life cycle support through electronic documentation of all processes and procedures etc. CALS application ensures not only «internal» information integration in terms of the corporate information system of shipyard, but also «external» integration for all parties involved in the ship's life cycle (shipowner – ship designer – shipyard – suppliers of materials and equipment, ship operation and reclamation). Thus, CALS is a global strategy intended to raise the efficiency of the Russian shipyards.

Integrated Information Models

CALS is based on a **set of integrated information models** for the ship's life cycle that are implemented in the course of its business-processes.

CALS is based on application of the following integrated information models:

1) **Ship model;**

2) **Business-system model.**

The ship model is generated by employing CAD/CAM, PDM, and special support facilities – CALS.

The business-system model includes models for the business-processes implemented during ship's life cycle, and data on the production and operation environment. The business-system model is generated by MRP/ERP, with Baan, SAP/R3, Symix, Oracle Application and other systems belonging to this class, and by project management systems MP – Microsoft Project, Time Line, Artemis Project, Prestige, Primavera Project Planner, Cresta etc.

CALS standards

CALS standards determine the format and content of information models for a product, its life cycle and environment:

• IDEF, ISO 10303 AP208 (STEP), ISO of 9000 series, ISO 15531 MANDATE – information description of product life cycle and business-processes implemented;

• ISO 10303 (STEP), ISO 13584 (PLIB) the product design and production model;

• ISO 8879 (SGML), ISO 10744 (HyTime), MIL-PRF-2000...2003,MIL-PRF-28003, MIL-M-87268, MIL-D-87269 the product operation model;

The **ISO 10303 STEP** standard (Standard for the Exchange of Product Model Data) determines a "neutral" format for product data representation as an information model.

Application of New Information Technologies in the Design of a New Shipyard

An investment project is currently developed in St. Petersburg for the purposes of building a contemporary compact-shipyard in the southern part of the city which is provisionally entitled ‹‹St. Petersburg Shipyards›› to replace three inefficient shipyards - ‹‹Baltic Shipyard››, ‹‹Admiralty Shipyards››, ‹‹Northern Shipyards››.

The design of the shipyard corporate system allows for implementation of CALS technologies (Fig.1).

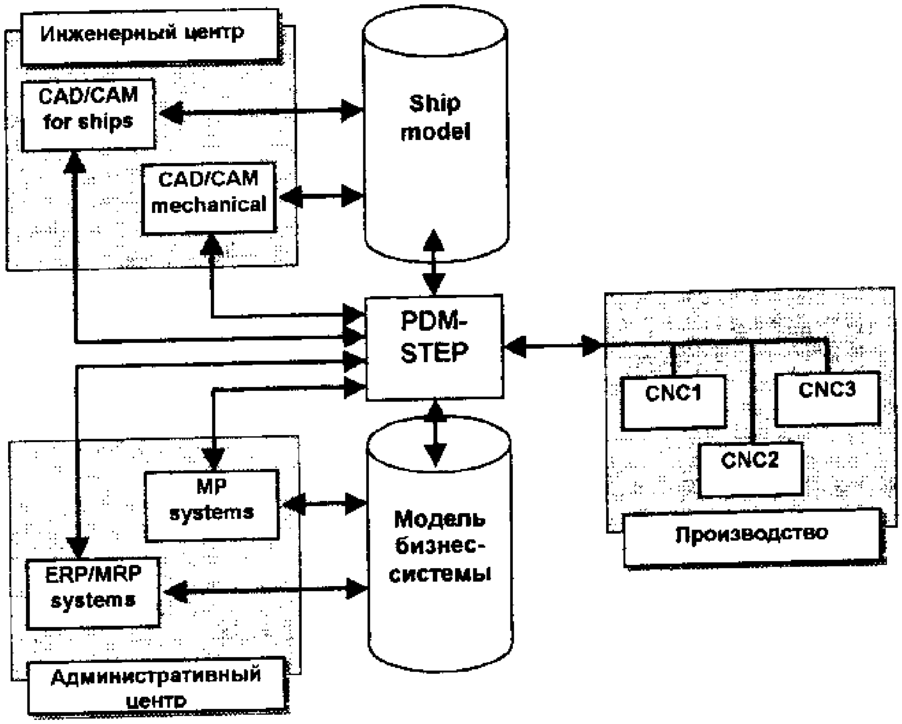


Fig. 1. Layout of Shipyard Corporate System

Automated CAD/CAM System Ritm-Sudno

The **Ritm-Sudno** system is an example of the Russian automated CAD/CAM system intended for shipbuilding applications. Conceptually, the **Ritm-Sudno** system incorporates the following modules: Hull, Mechanisms and Pipelines, and Outfitting.

This system may support the full cycle of technical preproduction engineering, including the design, technological preparation for production, and output of relevant data to production process management systems.

For the hull section the **Ritm-Sudno** system may ensure the entire design and technological preparation of ship hull manufacturing. This preparation includes creation of three-dimensional models of ship structures, development of nonflat components of ship plating, creation of a database for hull plates and shapes, nesting of material, calculation of control programs for NCU machines, output of relevant design and technological documentation. Integrated in the system are the programs for designing and calculation of data on the manufacturing of universal and special jigs and fixtures, some of them mechanized.

Support is provided for simultaneous work on different projects. When the designer turns from one design to another during one-run operation, the environment and relevant system settings are automatically readjusted. The user's settings for applied problems (calculation options, installation flags, control parameters, design catalogues, etc.) are also a function of the design processing and automatically reassigned. User's settings can be altered at any time.

To follow up changes in the ship series of the same design the database parameters incorporate attributes, such as *current order number* and *range of orders applied*. Employment of these attributes permits a reduction in the physical volume of database and allows the designer to work with different versions of a ship model.

Provision is also made for cutting out a part of hull structure at any level (module, section, unit) and for independent designer's work with this part.

Ship Specification

The **Specification** database contains detailed information on all components of ship structures. This information includes requisite data on a component, mass-weight characteristics and overall dimensions, coordinates for the component centre of gravity in the ship coordinate system, and parameters determining a technology for treatment of this component in a shop. The specification permits an output of the whole set of technological shop documentation. Special tools ensure adjustment for the form of "out" documents.

The system offers a variety of ways to replenish the specification. The most preferable way is automatic generation of specification during isolation of a component from a three-dimensional model of a ship structure. In this case the corresponding window setup options provide a means to control generation of separate parameters and features of the component.

Automatic communication is provided for the databases of nesting charts and the standard databases of the materials employed. Any change in the materials employed (brand of materials, thickness, overall dimensions of ordered plate) automatically entails readjustments in the specification and pattern card.

Information about **Specification** is stored as a relational database. An access to the database is provided through a standard SQL-interface. The system, however, is provided with a direct access to the database through special high-level language programs.

Some of the data stored in the database of this specification, are directly related to graphic information. With any change in graphic information, this permits automatic modification of the data contained in the relations of the relational database.

Three-dimensional model of ship structures

Three-dimensional modelling of ship structures solves the whole set of problems to set up ship surfaces and generate a design and technological database for ship structures. The main tasks performed at this stage are as follows:

- design of a lines model for hull as based on the main ship dimensions, calculation and output of the required documents in compliance with the lines drawing;
- calculation of actual frames, waterlines, and buttocks;
- generation of a three-dimensional design model for ship structures;
- breaking of a structure into construction units;
- calculation of butts, seams, and structural lines;
- determination of the system of bending and assembly allowances;
- setting of the welding by relevant lines;
- generation of all types of holes and cutouts.

Ship's hull lines can also be shaped in this system by the V-spline tools or they can be imported from the hull lines generation system developed by some other organization.

In building a three-dimensional model for ship's structures, provision is made for simultaneous isolation of hull plates and shapes and generation of a database for components. As this takes place, all technological parameters are determined. Communication with the specification and database of materials and standard designs is maintained automatically. Components are isolated from the model interactively, semiautomatically, or automatically. Automatic isolation of components is effected through a special software developed in terms of high-level languages or by the procedures developed in terms of a special problem-oriented language.

For nonflat surfaces, components are developed. During the development the calculations are made for all the data required for manufacturing the bending attachments and for effecting the bending itself. In calculation the values of longitudinal and lateral bending strains are controlled. The allowable values are empirically set in the system for the main ship's hull steels. If the bending strains exceed the allowable values, the system automatically offers acceptable designs (e.g. to set a bending allowance, or to set apart a component). During the development the overall dimensions of the component outline obtained are automatically checked for conformance to those of the order material.

Primary technological documents (technological cards, flowsheets, sketches, bills) can be produced at the stage of three-dimensional model generation.

Ship parametrization

The *ship parametrization* tools are designed to parametrically describe and generate the elements of ship's structures in the hull three-dimensional model, and hence to calculate hull components and processes for their treatment. Complete internal communication is provided for the design process model of ship's structures. The ship parametrization device can be employed in two modes: interactive operation and calculation per procedures.

The ship parametrization device is based on a problem-oriented language so as to effect ship's geometric constructions which ensure generation of ship's structure components, including plate elements with all their constructional and technological parameters. Geometric constructions can be effected both in a plane and directly in a three-dimensional model. The problem-oriented language includes statements for shaping the geometry of plate components and all technological operations therein. Furthermore, the problem-oriented language also includes statements for manipulation of lines and other objects of the three-dimensional constructional and technological model of hull, logic statements, and statements for computation of arbitrary arithmetic expressions.

A wide range of statements in the problem-oriented language permits development of parametric procedures. In this system the process of preparing data for such procedures is completely automatized. Allowance is made for various types of data, including the logic type, and references to the enumerated objects of the three-dimensional model. The text of the procedure, description of its parameters and initial data for calculation are held in the database. The required calculation procedure is selected from the procedure library through the list of current design components, the list of available procedures, or through a standard selection dialogue window.

There is a set of service functions for data manipulation. The group data assignment method and "default parameter" principle have also been developed. The filter system has a data sampling support and direct interface with the Specification database.

The interactive mode of operation is accompanied by automatic generation of language description, which substantially simplifies development of procedures and yields a powerful rollback system. Interactive debugging of user's procedures can also be effected. In the procedure binding of cutouts and scuppers to the relevant lines of the three-dimensional model, the framing cutout expansions are automatically allowed for.

Window setup parameters make it possible to modify and readjust user's dialogue dynamically, which maximizes incorporation of special design features and raises the efficiency of designer efforts.

Material nesting

A set of problems is solved to nest the plate and section material and generate an associated database for a nesting chart simultaneously. Charts can be generated in the system framework automatically and interactively. If the charts for nesting of plates are to be generated automatically, two programs are used.

One program implements original heuristic algorithms for obtaining automatically intricate patterns on a metal plate. These algorithms take proper account of the geometry of ship's components and special technological requirements that the shipyard may establish for location of parts in the charts.

The other program is based on the procedure of cutting out plates to produce arbitrary patterns automatically and irregularly. Theoretical developments that are known as the **Lipovetsky zone method** are also employed here. This algorithm is most efficient nesting of components with the form factor close to 1 (the form factor is the ratio of the component area to the minimum area of the enclosing rectangle).

In nesting, *commercial wastes* on a plate are automatically defined and taken into account for the purposes of further processing. By setting up relevant options and parameters in setup windows one may choose a strategy for location and spacing of small parts in cutouts, and to control the locations of long and narrow parts in a chart. Overall dimensions of a minimum *commercial waste* are adjustable values. By default the *volumetric commercial waste* is taken as a waste whose total area is at least 1 m^2 with the width of at least 400 mm.

Upon completion of nesting the path is automatically generated for cutting out plates on NCU machines (Fig. 2). If required, it can be adjusted in the interactive mode. A wide spectrum of postprocessors is provided to generate control programs for gas- and plasma-cutting machines.

Technological requirements for location of components in a chart and parameters of process facilities are set up in the relevant setup windows.

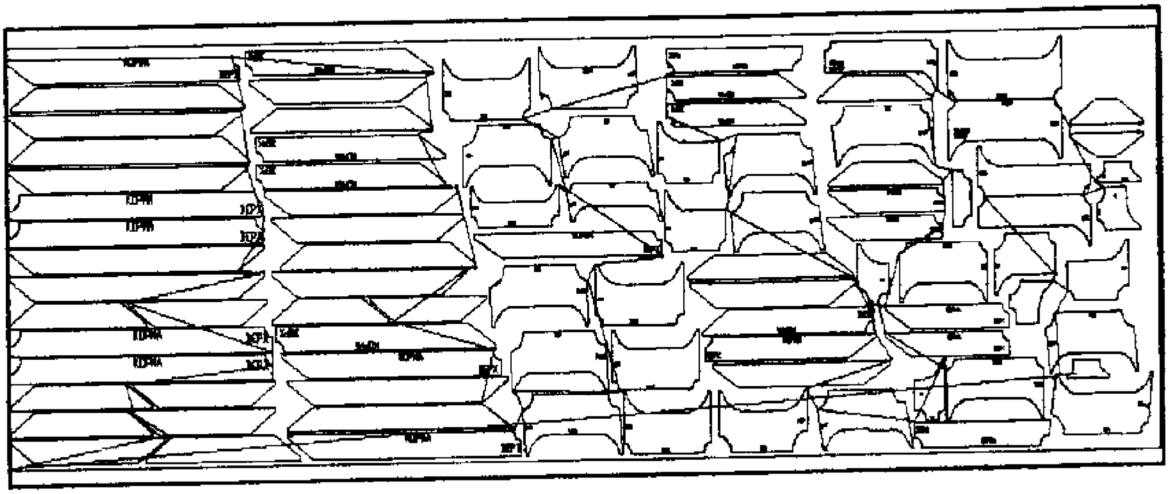


Fig. 2. Nesting chart Indicating a Path for Cutting out Parts

The main program for automatic nesting of section material or shapes is the procedure of linear optimal pattern cutting of bar material. In solving this class of problems, the classical approach developed by L.V. Kantorovich and V.A. Zalgaller, is to reduce these problems to a linear programming problem that can be solved by the column generation method. The efficiency of a problem depends on column generation algorithms that are based on the branch and bound methods, dynamic programming methods, and heuristic methods. Development engineers offer algorithms for application of methods of dichotomy to a broad class of combinatorial optimization problems. One of the algorithms that allow the linear pattern cutting problem to be solved by the method of dichotomy without linear programming is implemented in the system discussed here.

Design of special jigs and fixtures

Besides control programs for NCU machines, the system allows for the design and calculation of data on manufacture and application of the general-purpose and process jigs and fixtures, including mechanized accessories, in shipyard shops. These programs permit the calculation of data for manufacturing both the bending and the assembly and welding jigs and fixtures. Additional data are produced to adjust and check the fixtures for proper application in the technological process. These fixtmes may include bending templets, bending frames, assembly and welding jigs, both standard and universal (coke) beds. At the same time the drawings are developed for marking plates when the framing is to be installed during section assembly. Data are calculated for checking the sections during assembly and welding. Furthermore, data are also calculated to check the position of the ship hull in the shipbuilding berth.

Output of Working Drawings

The user's fundamentally new software - *inscription constructors* and *templet constructors* have been developed for generation and output of shop drawings on the basis of the three-dimensional model of ship structures. *Constructors* make it possible to automate the output of text information, special symbols and designations into a drawing and ensure automatic enumeration of components. Software tools permit generation of the library of inscriptions and special symbols to be entered in the drawing. The system of setup windows allows for the control of the form and amount of the text and graphic information on drawings as well as the control of its separate elements (size, extendable arrows, underlining, etc.). The standard inscriptions and templets held in the library can be changed at any time.

The drawing can be developed in two modes: the automatic generation of the Specification database and the application of the text database that has been created in compliance with the specification (Fig. 3).

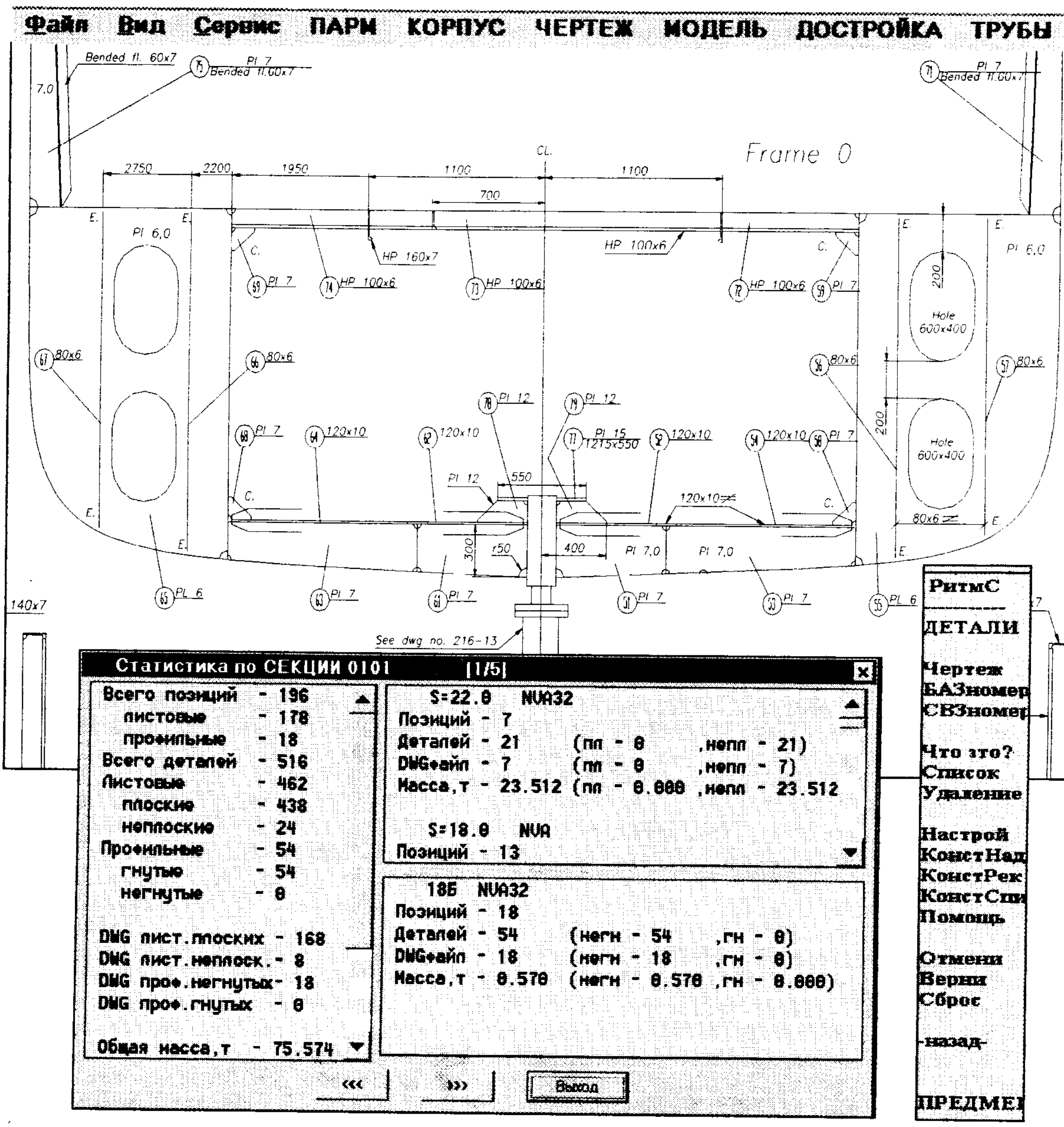


Fig. 3. Development of Ship Drawings and Specification

Laying-out of Facilities and Routing of Pipelines

The problems to be solved at this stage are as follows: location and spacing of machinery and facilities in the three-dimensional model of ship's hull, routing of ship's pipelines and preparing technologically the production facilities for manufacturing and installation of ship's pipelines (Fig.4).

The three-dimensional ship's structure model is generated for pipelines and machinery. Pipelines are routed and located in the ship's space. At the same time the standard reference text databases are generated for ship's machinery and equipment that are automatically related to their three-dimensional models.

The three-dimensional model permits the partitioning of pipeline routes into pipes and the generation of the article database for the system of pipelines. Primary technological documents (sketches and control programs for pipe bending) are issued. The partitioning of routes into pipes is effected by a special optimization program automatically.

The work with pipeline elements has been appreciably automated and simplified. Installation of these elements in the model complies with the parametric description of their dimensions. Support is provided for the following types of pipeline elements: separate flanges and their connections in assembly, unions, T-joints, welded pipe joints, valves of various types. Pipeline elements are all related to the text databases through parametric description of their dimensions. Besides the setting of these elements in pipeline routes, provision is made for their independent installation in a given point of space.

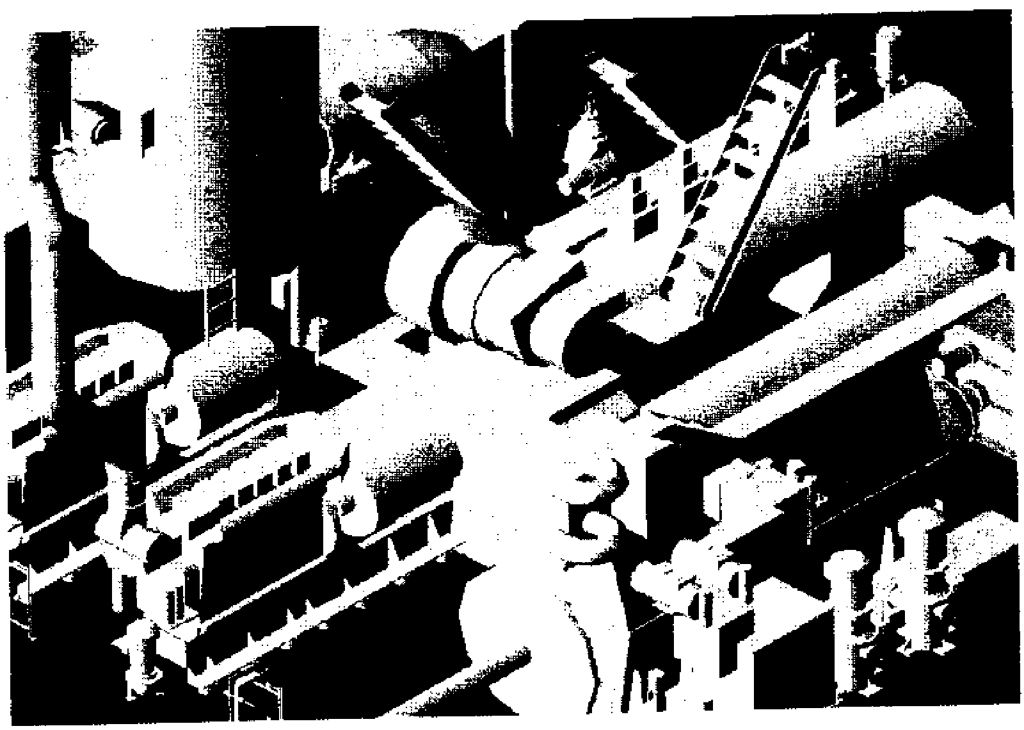


Fig. 4. Detail of a Location Model for Machinery and Pipelines

Interactive Three-dimensional Modeling of Outfitting Elements

The problems to be solved at this stage are as follows:

- generation of a database for outfitting elements;
- selection of equipment, hull fittings and furniture by the predetermined parameters from databases;
- setting the selected equipment in the three-dimensional hull model;
- modeling of the units and parts of furring ceiling and enclosure;
- modeling of paint coating, insulating covers, etc.;
- checking of the elements of hull, mechanisms, equipment, and furniture for interference with due regard for service zones after they have been installed in the model.

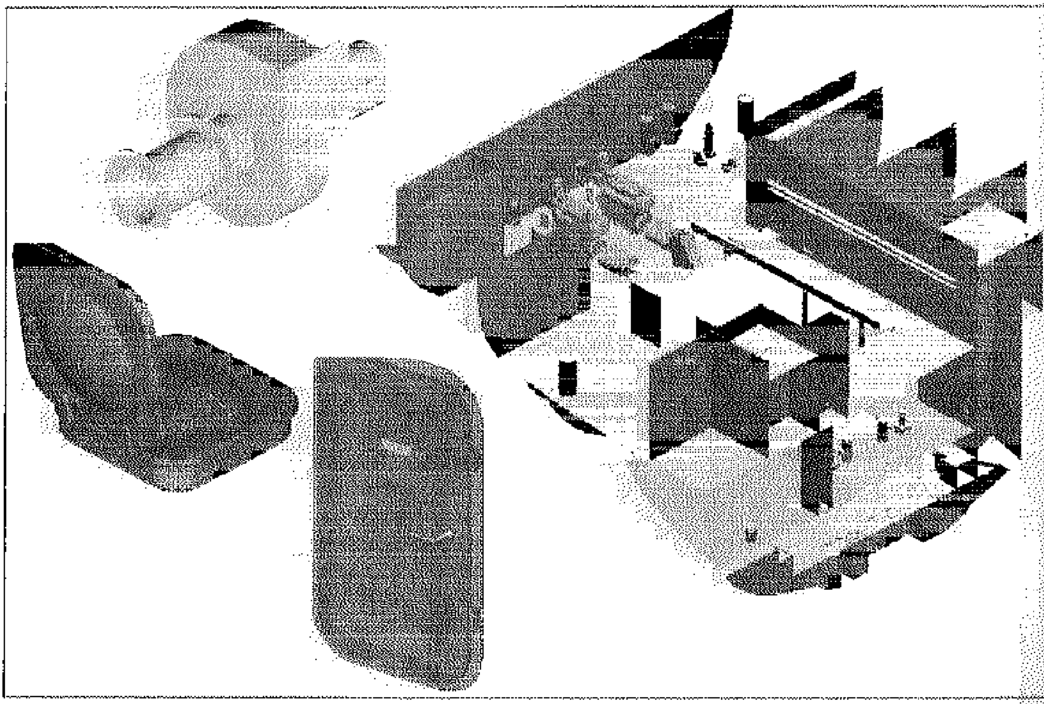


Fig. 5. Detail of Hull with Functional Elements. Individual Outfitting Elements.

Adaptation of System to Shipyard and Design

The **Ritm-Sudno** system has flexible tools for setting the forms of "out" documents to meet the shipyard requirements. User can independently design the form and text data to be placed in the "out" document.

To raise the efficiency of User's efforts the system embodies the tools for dynamic control of instructions. The dynamic control is in effect the possibility to use window adjustment and dynamic readjustment of the execution algorithm for separate system instructions (Fig. 6). The dynamic control is most efficient where the three-dimensional modelling requires operations of the same type or when the input requires determination of the repeated series of data.

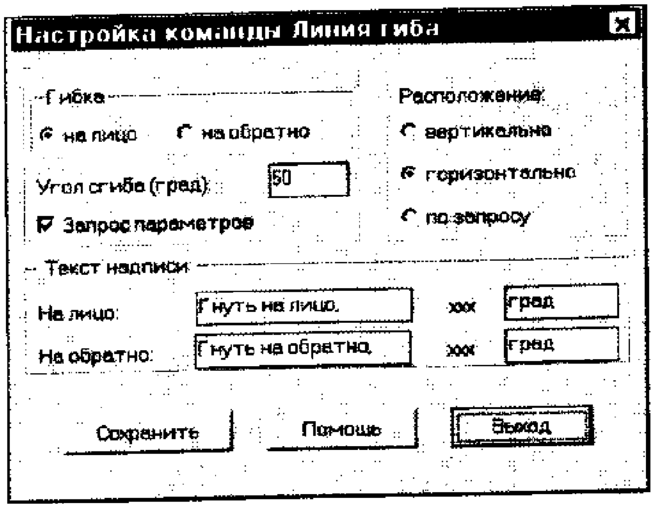


Fig. 6. Detail of Window Setup of System Instructions to Design

Elements of data management and control have been developed and installed in the system. These elements include the tools for fast scanning of text databases with the extended system of user filters. Implemented are the possibilities to obtain statistics on the database for drawings / sections / brands / profiles. Tools are installed to search for details through layout cards, and vice versa. Provision is made for all possible ways to enter instructions (screen and panel menu, plotboard, dialogue windows, keyboard entry, batch files). The system has an integral user aid and context-dependent hints. Provision is made for an integral reference system and process tracing. The window setup provides an easy way to readjust the file system and change the modes and control parameters for calculation. The developed interface ensures a fast and simple setup for the required configuration of technical aids, and for the design and special features of shipyard building and equipment technologies. The system software modules employ one database supporting both graphics and text data.

Contemporary development aids

The **Ritm-Sudno** system employs **AutoCAD-14** graphics editor and **FoxPRO** type database management system as a general system basis. In the development of application software the system implemented contemporary tools for development of user applications. The capabilities of **AutoCAD-14 Mechanical Desktop** applications are extensively used to model intricate ship surfaces.

The **Ritm-Korpus** system is oriented to application of personal computers, such as IBM PC of 486 and Pentium models.

Operation and Management

PLANNING AND CONTROL OF THE "COMPACT SHIPYARD 2000"

H. Anders, Aker MTW Schiffswerft GmbH, Wismar, Germany
C. Massow, unique informationslogistik GmbH, Bremen, Germany

Abstract

The shipbuilding of Aker MTW Werft GmbH can look back on a tradition of more than 50 years. More than 350 ships of 27 different types had been delivered. That is a result which is being successfully continued with the modernisation of the shipyard completed in 1998 to the "Compact Shipyard 2000".

In 1998 the MTW started a joint project to develop a prototype of an integrated planning and control system. It will base on the overall concept of the compact shipyard 2000.

With respect to the project results this paper will describe the characteristics of the compact shipyard approach. It will deal with the requirements towards planning and control. The integrated planning and control architecture and the approach of using reference models for planning will be outlined. Finally will introduce the software, which was developed in the project to transfer the approach into action.

The Compact Shipyard

The strength of MTW is the capability to manufacture special and standard vessels like tankers, gas, shuttle and chemical tankers, passenger vessels, containerships, ferries and offshore-components and steel components with outside hull production.

By the modernisation of MTW to the "Compact shipyard 2000" the shipyard has developed into an ultra-modern and productive shipbuilding location of strong competence in the field of technical product innovation. Efficient manufacturing plants and a consistent orientation of productivity according to the world standard have opened the entry to new markets. After a building period of 4 years and with an investment volume of DEM 600 million the "Compact Shipyard 2000" was completed in 1998. (1)

To give an idea of the complexity of the planning and control we will start with a brief technical specification of "Compact Shipyard 2000".

The **plate stock** is fully automated and controlled and monitored by a stock management computer. The plates are delivered with a degree of cleanliness of 2.5 and in primed condition. (1)

Cutting hall. The code of plates and of the individual parts to be cut is applied by means of a marking station with the help of a master computer. 3 underwater plasma cutting plants (WIPC plants) working with plasma cutting torches with pollutant gas exhausters ensure cutting. The plants are provided with plasma marking devices to apply reference marks and markings as a prerequisite for subsequent assembly processes (accurate manufacture). Plate edge milling machine (CNC or DNC). Plates up to 16 m length are milled to finished size here with the corresponding preparation of welding edge (tolerance ±0.2 mm on 16 m plate length). 3 driverless transport vehicles connect a total of 23 work-stations for the material transport of plates, profiles and completed individual parts on pallets by remote control (control computers for driverless transport system). (1)

Any standard profiles of 16 m maximum length are cut in the **profile cutting hall** with a profile edge milling machine, two robots with air plasma arc cutting plant in DNC operation and corresponding interlinking equipment. Panels and subassemblies are manufactured in the **hull component production**. The construction of small units from plates and profiles cut as well as small panels serves as a prefabrication station for voluminous unit construction. (1)

Straight and bent panels are manufactured on flow lines in the **panel construction hall**. Panels and double skin sections up to 240 t weight and 16 m x 22 m dimensions can be manufactured on the 16 m panel line. Voluminous units of flat and bent panels up to 320 t weight are welded in the **voluminous unit construction hall.** The hall is equipped with a unit turning and transport crane of 320 t as well as 2 assembly cranes of 32 t each and has flexible building sites. (1)

All hull sections are coated to a large extent in environment-friendly procedures in the **coating hall**. The coating of sections comprises separate boxes for blasting and for cleaning/coating/drying of panels and voluminous units. (1)

The centre-piece of the "Compact Shipyard 2000" is the **dock hall** with the building dock. The block assembly and final assembly are carried out here. It consists of the areas for block construction and the Building Dock for final assembly work. Block units are assembled in the Block construction area from voluminous units and panels up to a maximum weight of 800 t. In the building dock vessels are assembled from block units equipped beforehand. The trestle crane operating above the building dock has a lifting capacity of 800 t (1,000 t max.) with 143 m span. The dock can be subdivided into two by an intermediate gate. (1)

The basic philosophy of the compact shipyard concept is:

- Innovative ship's concepts
- Roofed production areas
- Short manufacturing times with precise observance of dates
- Short distances in production
- Minimum intermediate storage of sub-products
- Automatic production lines
- Manufacturing meeting quality standards

Planning and control of manufacturing processes, inclusive design and technological process planning, as exact as possible is a pre-requisite to make use of the high potential of manufacturing ships in a compact shipyard. Basis for that is an integrated planning and control system, which covers the preparation and performance of the whole value adding process. The goal is to reach an overall optimisation of the order processing.

Therefor the Aker MTW initiated in 1998 a joint project to develop a prototype of such an integrated planning and control system for the compact shipyard 2000. The project named "PLUS" (Planning and Control System for an Compact Yard) is supported by the German Government. Because of available results of previous work was it possible to keep the project duration within 20 month.

The project partners are:

- Aker MTW shipyard GmbH, Wismar
- Fraunhofer Institute Manufacturing Engineering an Automation (IPA) Stuttgart
- Project Group Rostock of IPA, Rostock
- unique information logistics GmbH, Bremen
- IMAWIS GmbH, Wismar
- Scheller System Technique GmbH, Wismar

The system will serve as common planning and control instrument for all parties involved in the process from strategic company planning to shop floor control. It will also cover the integration of suppliers and co-operating partners. To enhance reliability of plans and to accelerate the planning process a set of measures will be created using as-is-values as input for planning and simulation tools.

The projects main objectives are:

- Development of an integrated planning architecture for the compact shipyard
- Integration of all planning activities for both shipyard recourse planning and customer order management.
- Definition of a hierarchical set of so called reference models for planing, simulation and control.
- Specification and implementation of methods to capture and calculate the actual data for the reference models.
- Specification and implementation of a software tool to manage reference models and their application in planing and simulation.

Integrated planning and control architecture

The PLUS planning and control architecture consists of 5 levels.

Table 1. PLUS Planning Level Description

Level	**Area**	**Object**	**Horiz.**	**Goal**	**System**	**Methods**
Perspective Planning	Yard	Ships consisting of big units (Rings)	1-3 Years	Meet customers needs (especial delivery date); Dock utilisation Balanced load of resources	GIGROS	Multi-project planning using Reference Modules, Spatial resource Planning
Rough Planning 1	Halls	Shipbuilding: Sections; outfitting: Big units, systems	1 Year	Meet customers needs (especial delivery date); Dock utilisation Balanced load of resources; Minimise Costs	GIGROS	Dynamic, hierarchical network scheduling; Multi-project planning using Reference Modules
Rough Planning 2	Manufacturing Areas incl. Indirect Areas (e.g. design)	Shipbuilding: Units; outfitting: Systems	½ to 1 Year	Minimise Performance Costs	GIGROS	Dynamic, hierarchical network scheduling; Multi-project planning
Detailed Planning	Work-Shops; Design Groups	Shipbuilding: Units; outfitting: Systems	3 Weeks	Meeting internal delivery dates; Balanced Capacity Load	GIGROS; 4PM; SMC-"Paneel-Line"	MRP; capacity planning; Material Flow Simulation
Control	Foreman Areas	Materials from Single Part up to Sections	1 Week	1. Meet schedule, 2. Balanced Capacity load	unique Production Control; SMC "Paneel-linie"; 4PM	MRP; capacity planning; Material Flow Simulation; Network scheduling

The approached architecture basis on the concept of hierarchical production co-ordination. In this concept, the harmonisation of production and its activities with respect to time and content is given priority. Customer orders are handled as projects and the simultaneous production processes, their activities and the resource requirements are harmonised with multi-projects in mind. Based on this, a two-layered production co-ordination and control concept was developed. The concept consists of to layers. A decentralised independent planning and control layer for autonomous manufacturing areas (in this case Rough Planning 2 and Detailed Planning). A centralised planning and co-ordination layer (here Perspective Planning and Rough Planning 1). This integrated concept enables the devolution of decision making to the lowest possible level of operational responsibility. Therefore, the planning activities of autonomous production areas are supported by decentralised shop floor monitoring and control (SMC) systems.

Harmonization / Harmonization by	horizontal	vertical
constrains		classical approach
feedback		centralised co-ordination
negotiation	decentralised synchronization	co-operativ co-ordination

Figure 1. Co-ordination concepts for one-of-a-kind production

The vertical integration of the planning levels is realised by joining the co-ordination level with the production area control level via closed control loops. This integration process is supported by a revolving planning concept. That means, planning and control activities will be started not by time intervals but by events. In figure 1 the chosen harmonisation approach is arranged in the order of co-ordination concepts for one-of-a-kind production.

Supporting IT

The planning system architecture is build out of the following systems (see Table 2).

Table 2. Elements of the System Architecture

System	Task	Platform
"GIGROS"	Object oriented, hierarchical network planning, Resource planning in a multi-project situation, Project monitoring and control	NT; SQL-Server
"PS-System"	Material Requirement Planning, Bill of Material, Feedback	UNIX; ORACLE
"Production Control"	Detailed Scheduling, As-is-data (due dates), Physical work progress, Checklists	NT; SQL-Server
"Paneel-Linie"	Material flow simulation and optimisation	NT; MS-Access
"RAMOS"	Resource and activity management on shop floor level	NT; SQL-Server

The planning and control of complex manufacturing processes such as shipbuilding requires support of the reuse of knowledge during order processing. Therefor all involved planning systems are designed to handle this kind of information. The concept to gather and reuse such experience and feedback data is called reference modules.

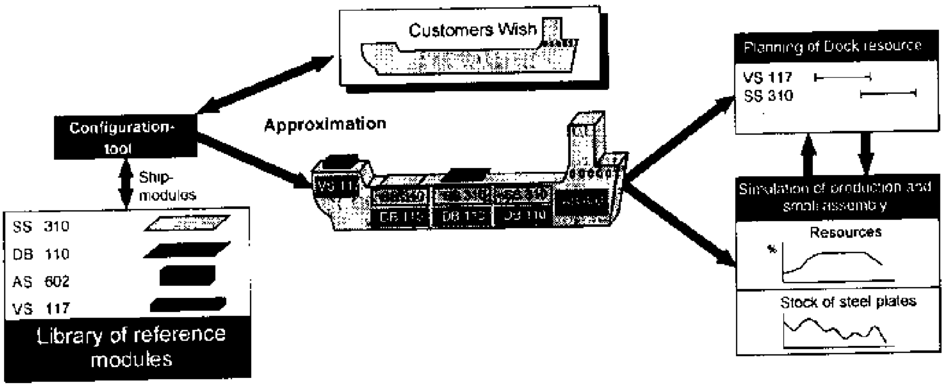


Figure 2. Approach of Reference Modules

The reference modules support the planers creating a new planning model by providing them with standard objects. The classification of a reference module is done according to certain product attributes which lead to different process parameter. The reference modules library is a hierarchical set of modules with respect to the common product structure

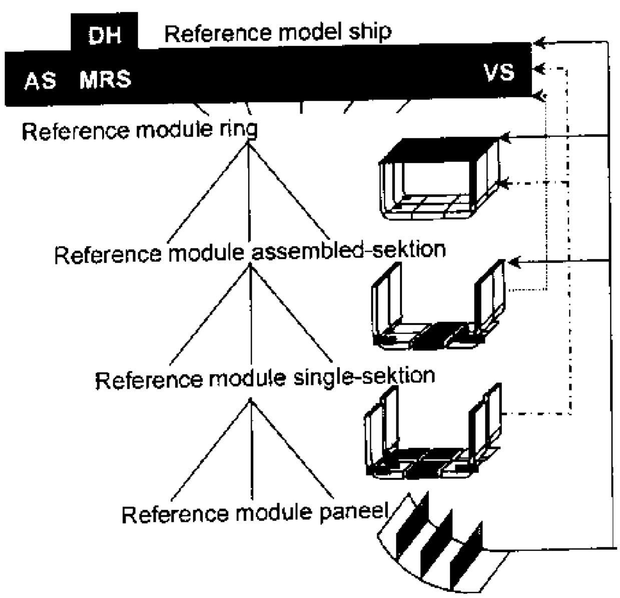


Figure 3. Reference Module Hierarchy

Therefor the system architecture must also follow an hierarchical approach to be able to make use of the different reference modules of different levels.

The definition of a reference module for a new project is done by, first selecting one from the library according to selection attributes, and then describing it using parameterise attributes. Each reference module is linked to standard planning elements in one or more of the planning systems.

Table 3. Matching of Reference Modules and Planning Elements

System	Standard Planning Element	Reference Module
Project co-ordination system "GIGROS"	Project-Version; Capacity Demand; Network; Sub-Network; Part-Network; Activity; Type-Curve	Ship; Ring; Assembled Section
MRP-system "PS-System"	BOM; Work Order; Accounting structure	Single Section; Paneel
Production Control System "Production Control"	Product-Activity Relations; Work Order Responsibilities	Assembled Section; Single Section
Shop floor monitoring and Control system "Paneel-Line"	Processes	Paneel
Shop floor management system "RAMOS"	Task; Type-Curve	Assembled Section; Single Section

As an example how to use reference modules we would like to describe the scheduling and resource allocation on the perspective planning level of a new ship to build. The responsible planer selects an appropriate reference module for the ship in question using the configuration management of the GIGROS system. This leads to related big units (ring), which then have to be parameterised. According to the selected reference modules the rough planning system creates a project with all its elements.

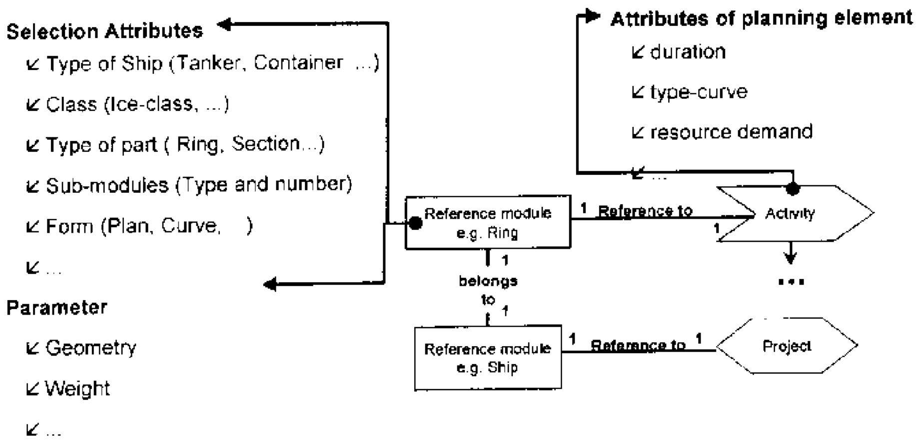


Figure 4. Reference Modules and Project Items

The GIGROS system is designed to handle this kind of hierarchical approach because of its internal data structure. Product data, process data, and resource data are structured in such a way to support the Top-down approach which is typical for one-of-a kind-production.

The next step is to adopt the automatically created project according to the actual shipbuilding project. Order processing in the shipbuilding industry requires project models with at least hundreds production activities and their corresponding technological and chronological inter-dependencies. Therefor the GIGROS Network Editor allows the representation and the definition of a selected project in an activity-oriented network.

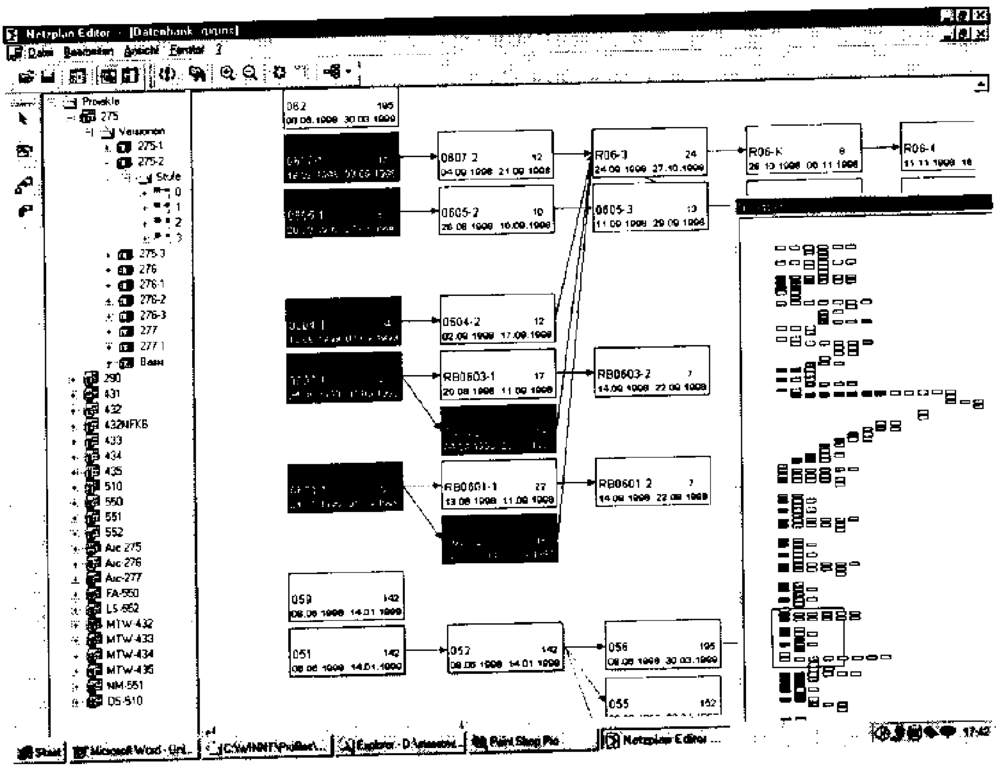


Figure 5. Networkplan Editor

In order to guarantee a clearly arranged representation (essentially defined by the number of crossings of interdependencies) an heuristic for the arrangement of activities has been integrated through which the number of crossings are minimised. (2)

With the help of this module the planer adds and removes activities and activity relations or creates new sub-nets to detail the reference project.

After building the project structure it must be scheduled. To cope with complex one has to determine the start- and due-dates for activities. This is supported by our hierarchical network scheduling approach. Activities on higher levels provide time-frames for the sub-nets on a lower level. Scheduling of sub-nets is possible only within this given frame. Thus a support of partially autonomous and decentralised production areas is provided.

The deterministic scheduling is a prerequisite to the consideration of a project in the decision supporting modules. In order to achieve an event-driven production co-ordination, the function "scheduling" can be performed both project oriented and under consideration of the feedback data. If there are project inconsistencies i.e. not allowed loops or time limits the function "fault analysis" undertakes the error search process. (2)

The planning of the new ship also have to consider the resource availability and allocation in the multi-project situation of a shipyard. Capacity demand as well as available capacities are more or less subject to change in course of time. The task of the resource planning is to align capacity demand and available resources. GIGROS supports to determine the availability of resources as well as the estimation of demands and its temporally distribution with its interactive capacity planning module. (3)

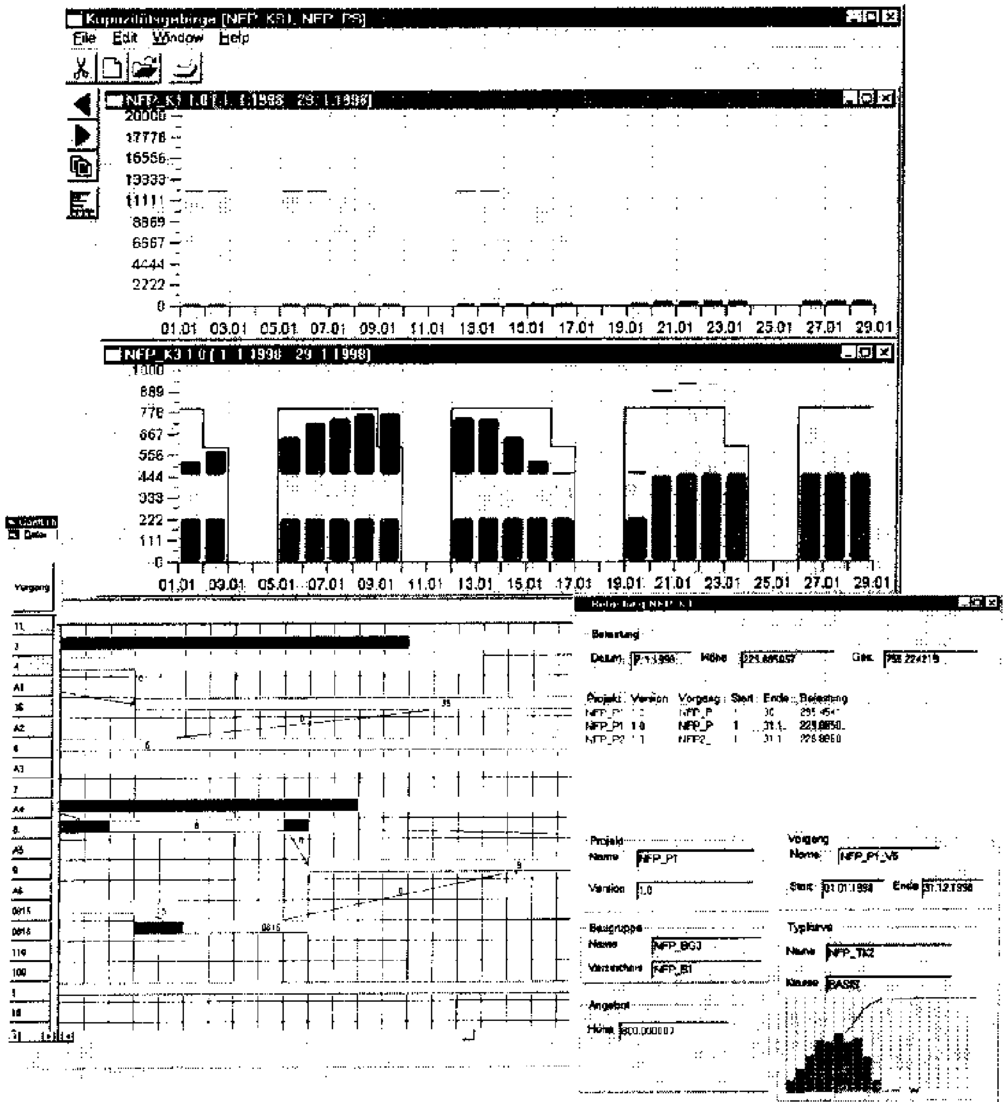


Figure 6. Interactive Capacity Planning

The planner uses the resource demand profiles derived from the reference modules and adopt them according to the actual situation (e.g. decreasing budget because of the learning effect). Analysing functions identify time periods (days, weeks, or months) were critical over- or underloads are occurs.

Whereas the Capacity Diagram supports capacity-oriented overviews and analysis the Dynamic Gantt Diagram offers functionality for time-oriented decisions. The diagram presents each time the production activities of the momentary capacity level, their current time positions, "current" buffers and total buffers. The diagram assists graphically and interactively the shift of activities and the modification of activity duration.

The short example showed, that the planning expert is able to integrate a new shipbuilding project into the yards master plan in short time and reduced effort when using the reference modules.

Conclusion

The described approach and the tools developed the PLUS project are currently in the status of pilot installation and testing and the Aker MTW Shipyard. The results so far are very promising because the time to prepare plans and their reliability has increased since the implementation of reference modules. The shipyards management expects the planning quality to be enhanced through this experience and knowledge feedback method.

The crucial point is the clustering of the reference modules and the definition of related planning data. The suitable mechanisms for that are implemented and the quantity of relevant data and therefor the quality of the reference modules will increase in the course of time.

The PLUS project focuses on the scheduling and resource planning. The potential of the reference module approach should be used to enhance the cost budgeting and controlling also.

The same goes for spatial resource planning. In contrast to human and machine resources, production activities can have a two-fold assignment to spatial resources (i.e. production premises such as halls). Between spatial resources and production orders time assignments are necessary such as those between orders and machine resources. An additional assignment is required to fix the location of the order within the production premises. In order to support these kinds of planning activities the Spatial Resource Management Module was developed and unique will enhance it with reference module functions in the near future.

References

1. Aker MTW Shipyard GmbH: The Compact Shipyard; WorlWideWeb publication http://www.akermtw.de/Seiten/Art08/MTW_T.htm
2. Kuhlmann, T., Marciniak, Z., Massow, C. (1994): Integrated Coordination Modules for the Shipbuilding Industry; in: J. Brodda; K. Johansson (Editors): 8th International Conference on Computer Applications in Shipbuilding (Volume 2); Malmö, Sweden 1994
3. Kuhlmann, T., Lamping, R., Massow, C. (1998): Build and operate logistic chains — An approach to synchronise distributed production activities; in P. Schönsleben, A. Büchel (Editors): Organizing the Extended Enterprise; London, 1998

A SIMULATION BASED SCHEDULING FOR AN INTEGRATED HULL, OUTFITTING AND PIPING WORK AT DOCK STAGE

Duck-Young Yoon, J W Park, I Y Yang

College of Engineering, Chosun University, Kwang-ju, Korea

Abstract

A simulation based scheduling algorithm for pre-erection stage and erection stage has been proposed, in which three dimensional spatial scheduling problem is converted to two dimensional one and thereby two dimensional nesting algorithm can be utilized.

Introduction

In shipbuilding industries, design, process planning and scheduling works are performed concurrently and are highly inter-related as shown in Fig 1.

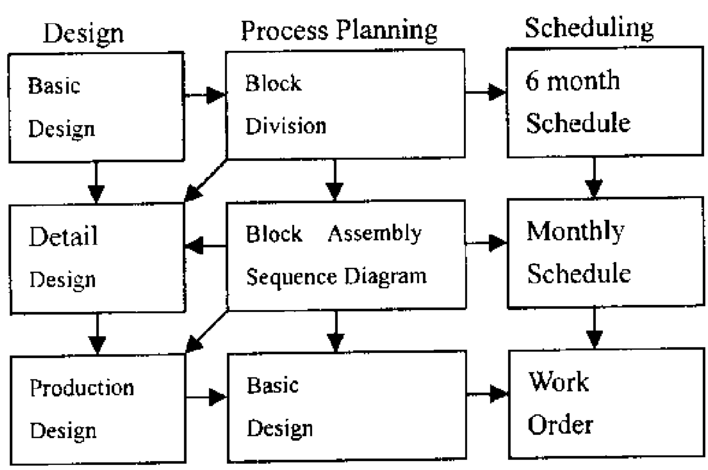


Fig 1. Inter-relations among Design, Process planning and Scheduling

In order for the dates of key events such as keel laying, launching and delivery to be kept, the master schedule should be evolves to schedules at dock stage, pre-erection stage block assembly stage and cutting/fabrication stage as soon as practical. Especially those at dock stage and pre-erection stage are necessary to set due dates for assembly blocks. Schedules at dock and pre-erection stage are prepared based on initial design information and the block division.

Scheduling at dock and pre-erection stage involves spatial scheduling problem and has three decision variables:

- The locations of pre-erection blocks in the pre-erection area (Spatial Allocation Issue)
- The start times and completion times of pre-erection work (Temporal Allocation Issue)
- The resources used by work (Resource Allocation Issue)

Since these three decision variables are highly interdependent, it is necessary to deal with these variables in one framework.

The objectives these scheduling system persue are due-date satisfaction, maximum utilization of spatial and non-spatial resources and minimization of waiting time.

Typical constraints include crane availability, man-hour availability, physical adjacency of coupled objects for operational efficiency and minimum required clearance between blocks

Network representation of erection sequence

In our model, nodes represent erection activities of blocks and arcs are used to represent physical joints between blocks. Therefore the attributes of arcs are joints length and possible status of joints such as fitting, welding or inspection.

Some of the arcs used to represent the sequence of erection are directional and pitch, which is the minimum period of time to accommodate next erection block, is added to attributes additionally. Keel laying activity and launching activity will be added to block erection nodes.

Erection sequence generator

The role of erection sequence generator is to determine the directions of directional arcs and to calculate the earliest starting time (EST) and the latest starting time (LST) of erection of each block. See Fig 2.

Spatial scheduling at pre-erection stage

Most of pre-erection block which consist of cargo holds have same length and relatively short working period of time compared to those which consist of engine room, after body and fore body. In addition, their shapes are rectangle. Furthermore their orientations are fixed since the Goliath crane has no rotational function. This facts hints us to design bays with same widths to accommodate the length of pre-erection blocks and the spatial load of each bay at a given time can be represented by one variable, occupied length. This simplifies spatial scheduling from three-dimensional problem to two-dimensional one. See Fig 3. As we can see, spatial scheduling problem now converted to two-dimensional nesting problem of rectangles on the rectangular working plates, one dimension is length and the other is time.

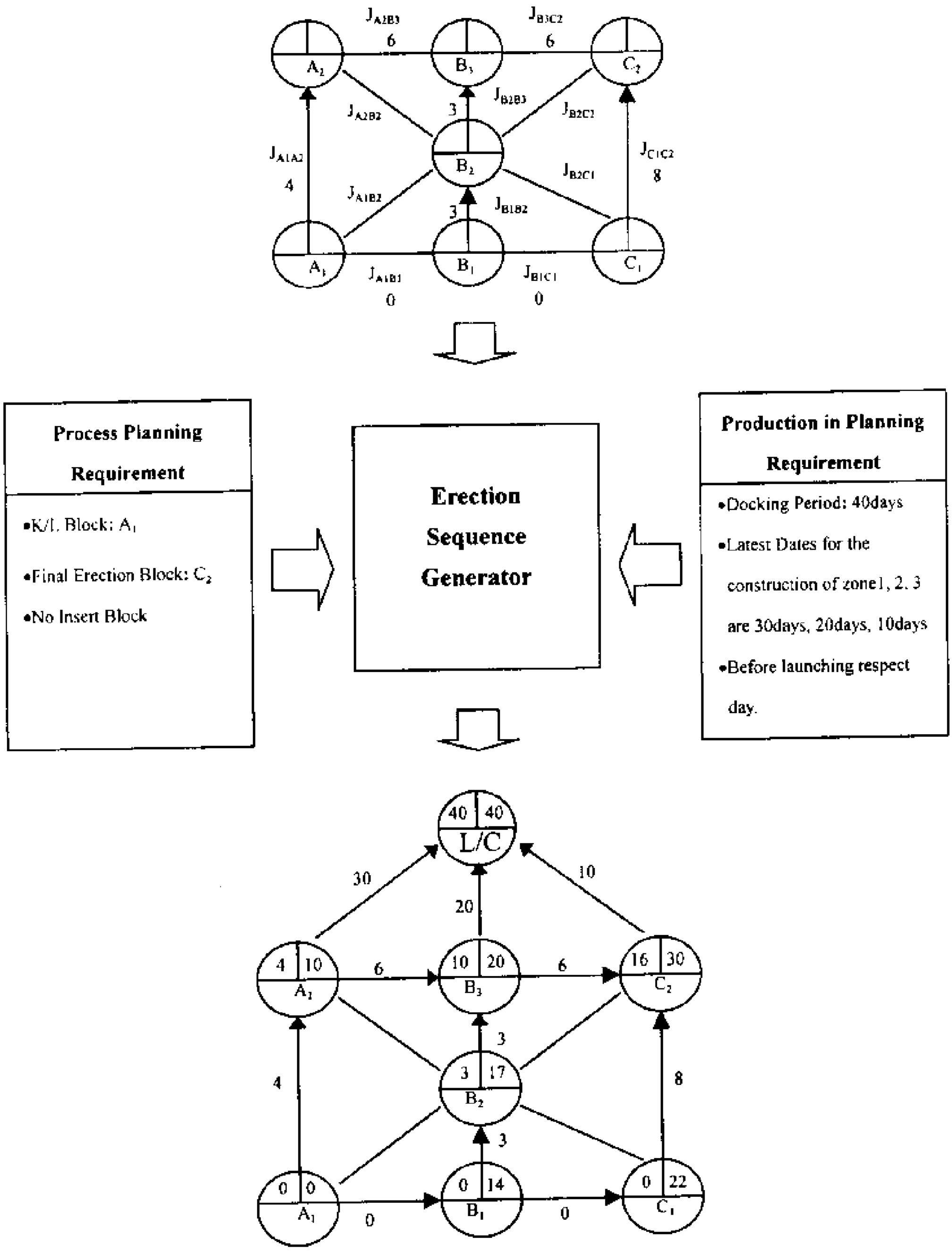


Fig 2. Erection Sequence Generation

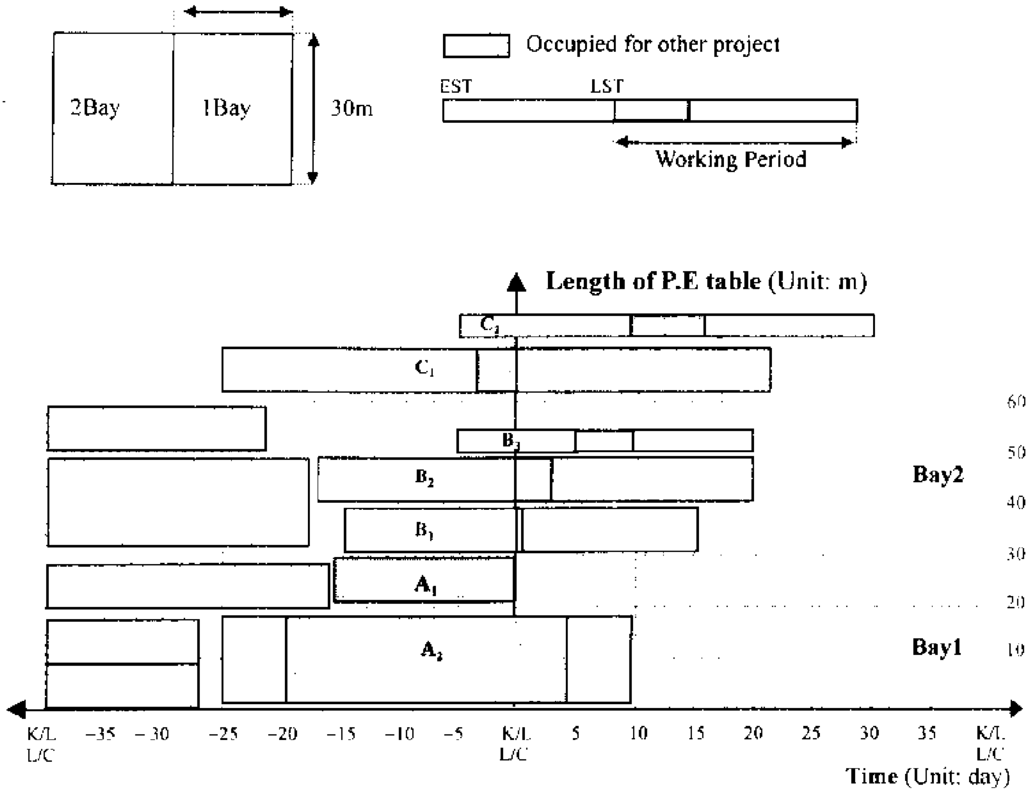


Fig 3. Two Dimensional view of spatial scheduling

Nesting strategy

Nesting strategies we adopt are:

-Nesting order is based on not time but area (width x time). It is natural to place a bigger rectangle first.

-Maximize free rectangular space. The concept of the free rectangle set is similar to the prime convex area used for finding a good path in robot path planning(1, 2). For the simple understanding of maximal free rectangular space, see Fig 4, where maximal free rectangular spaces are defined as (R1+R2+R3), (R2+R4+R6+R8), (R2+R3+r4+R5) and (R7+R8).

Scheduling procedure

Scheduling procedure we propose involves the following steps:

Step 1

Generate a erection sequence based on process planner's requests and major mile stones imposed by production planner.

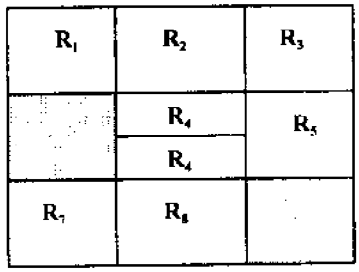


Fig 4. An Illustrative Free Rectangular Space

Step 2

Calculate ENT and LNT for pre-erection work based on those for erection and standard working period of each pre-erection block at pre-erection stage.

Step 3

Assign rectangles to the appropriate bay

Step 4

On each bay, nest the rectangles according to the nesting strategies.
Whenever the placement has been occurred, ENT and LNT of each shall be updated.

Example

In order to explain the algorithm proposed, let's take a simple example.
The process planner determines a block division shown in Fig 5.

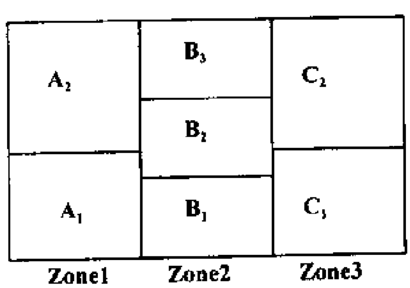


Fig 5. Block Division

In addition, A_1 and C_2 are assigned as the keel laying block and the final erection block, respectively. Fig 2 shows the initial input network model, where seven erection blocks are modeled as nodes. Joints J_{A1A2}, J_{B1B2}, J_{B2B3} and J_{C1C2} are to be directed arcs and their directions are pre determined due to the topological relations.

Joints J_{A1B1}, J_{B1C1}, J_{A2B3} and J_{B3C2} are also directed arcs since they represent erection method, which directions will be determined by erection sequence generator.

The constraints imposed by process planner are:

- The docking period is 40 days.
- The latest dates for the constructions of zone 1, 2 and 3 are 30 days, 20 days and 10 days respectively.

Fig 2 shows the proposed erection sequence, where EST and LST of each erection blocks are assigned to each erection block.

Table 1 shows information for pre-erection scheduling.

Fig 6 shows a final result.

P.E Block Name	Size (L × B)	P.E Period	
		Shortest	Optimal
A_1	22 × 10	13	15
A_2	22 × 18	26	29
B_1	22 × 10	13	15
B_2	22 × 10	17	20
B_3	22 × 4	13	15
C_1	22 × 10	22	25
C_2	22 × 4	17	20

Table 1. Input data Pre-erection stage for Scheduling.

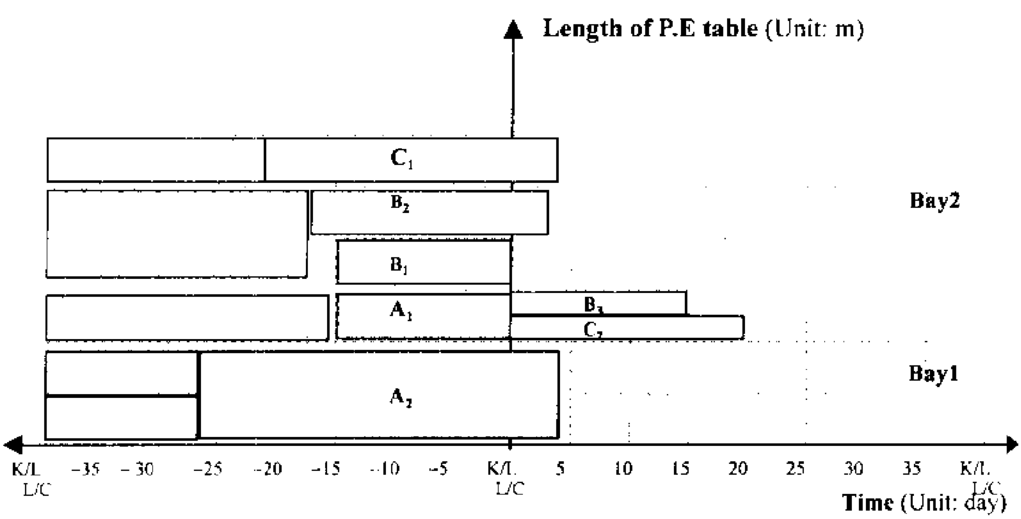


Fig 6. Final Scheduling Result

Conclusion

This paper proposes the way of converting three dimensional spatial scheduling problem to two dimensional one and thereby provides the possibility of minimizing scheduling failure.

References

1. Singh, J. S. and Wangh, M. D., Robot Path Planning using Intersecting Convex Shapes: Analysis and Simulation, IEEE Journal of Robotics and Automation, vol. RA-3, no 2,pp.101-108, April 1987.

2. Hayashi, A. and Kodama, S., Free Space Decomposition and Path Planning in 3D, Proceedings of '92 The second Pacific Rim International Conference on Artificial intelligence, pp.294-299, 1992

3. Lee, K. J., Development of Spatial Scheduling Expert Systems, PhD. thesis, Dep't of Industrial Management Korea Advanced Institute of Science and Technology

A PLANNING AND CONTROL METHOD FOR SHIPYARD PROCESSES: A SHIPREPAIR YARD CASE STUDY

George M. Chryssolouris[1], *University of Patras, Patras, GR*
Kyriakos A. Bechrakis[2], *University of Patras, Patras, GR*
Sotiris J. Makris[2], *University of Patras, Patras, GR*
Stathes D. Xeromerites[2], *University of Patras, Patras, GR*
Dimitris A. Mourtzis[3], *University of Patras, Patras, GR*
Nikolaos V. Papakostas[2], *University of Patras, Patras, GR*

Abstract

This paper describes a planning methodology and its application to a repair shipyard. The method includes an hierarchical model of the shipyard's resources and their workload. The model including an assignment logic of the workload to the shipyard's resources has been implemented in a software system. The system simulates the operation of the shipyard and produces a schedule for the resources and a set of performance measures, which enable the user to evaluate the created schedule. A set of scheduling experiments with data coming from a shiprepair yard have been conducted in order to validate and test the approach under different conditions.

Introduction

A major difficulty of planning the operation of shiprepair yards is that between the start and the completion time of the repair work there are typically many work changes which include added, cancelled work and priority changes. Ship repair jobs are often done to a fixed budget and is therefore a "Fixed Cost" exercise. Therefore the major requirement for the planning of a shiprepair yard is the ability to produce quickly a good schedule with a system that is flexible and adaptive to changes of production data.

The most commonly adopted approach to the planning of repair shipyards, is modeling the problem as a *resource constraint project scheduling problem*[4-9]. This paper proposes a different approach which is based on an hierarchical manufacturing model, adapted to the characteristics and the requirements of repair shipyards. This approach is proposed as an alternative which may be more flexible and configurable than the rigid mathematical models used in the literature.

The Planning Model of the Repair Shipyard

The work described in this paper is based on the operation of an actual repair shipyard. The Production System of the shipyard to be modeled consists of six sections: the Supervisor Engineers, the Mechanical, the Naval Works, the Riggers & Painters, the Technical Support and the Quality Management Section (Figure 1). The Supervisor Engineers supervise the shiprepair activities. The Mechanical Section involves two groups of fitters with different responsibilities, one group responsible for the interior of the ship and one responsible for the exterior of the ship. This section also includes a group of workers specialized in chemical cleaning, and the two floating docks. The Naval Works section consists of five groups: the platers, the welders, the pipers, the boilers and the carpenters. The

1. Professor, Director of the Lab for Manufacturing Systems, University of Patras, Greece
2. Research Assistant, Lab for Manufacturing Systems, University of Patras, Greece
3. Project Manager, Lab for Manufacturing Systems, University of Patras, Greece

Riggers & Painters Section includes three groups: the riggers, the painters and the vehicle drivers and operators. The Technical Support Section includes the electricians, the firemen and the people working on the piers of the dockyard and the tugboats. Finally, the Quality Management Section includes the chemist. the foundry of the shipyard and the Quality Control section.

From a planning point of view the important parts of the sections of the shipyard are the docks. the fitters. the platers, the welders. the pipers the riggers, the painters and the electricians. Therefore the modeling and planning approach has been concentrated on these parts which are shown in boldface characters in Figure 1.

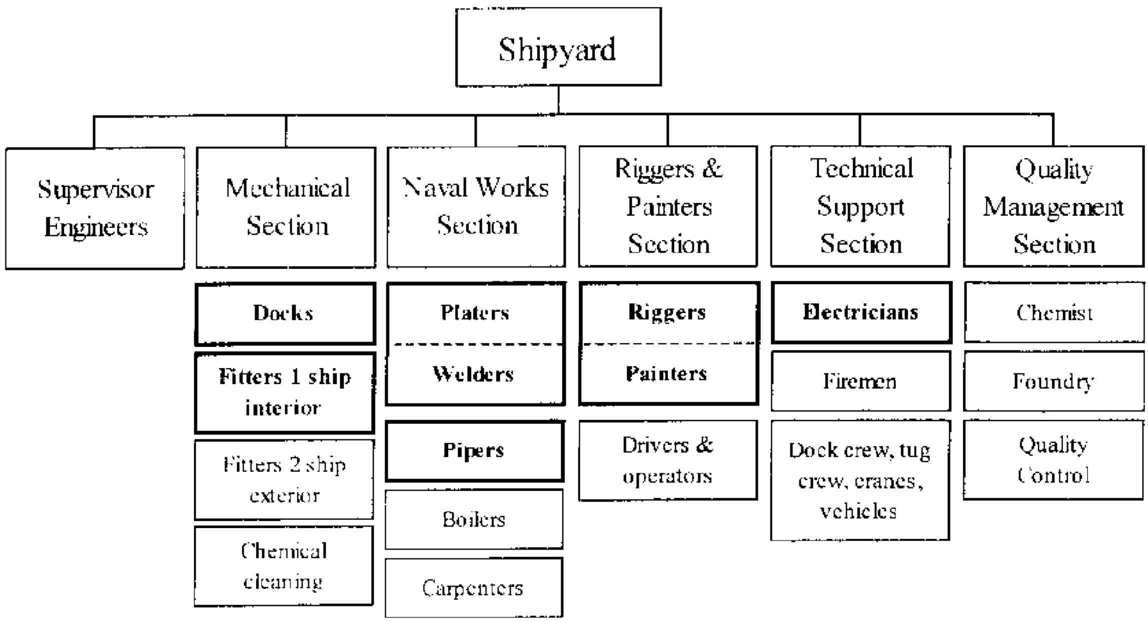


Figure 1. The sections of the repair shipyard

In this work an hierarchical model with four levels has been adapted to the repair shipyard's planning problem (Figure 2).

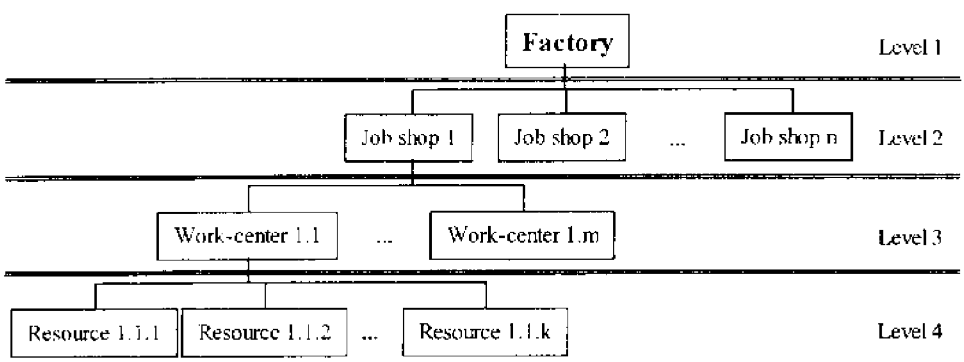


Figure 2. The four-level hierarchical model

The Factory corresponds to the entire shipyard and includes a number of Job Shops. Each Job Shop consists of a number of Work Centers, which in turn consist of a number of Resources. Job Shops correspond roughly to the sections of the shipyard, while Work Centers correspond, to some extend, to departments of the sections. The Resources included in each Work Center are a sort of "parallel processors", namely they can "process" identical Tasks. Depending upon the assignment logic or dispatching rules, a Task is assigned to one of the Work Center's Resources. In this particular application the term Resource is used for a group of workers who are typically working, according to the shipyard's rules, to a particular Task, from the beginning to the end of the Task.

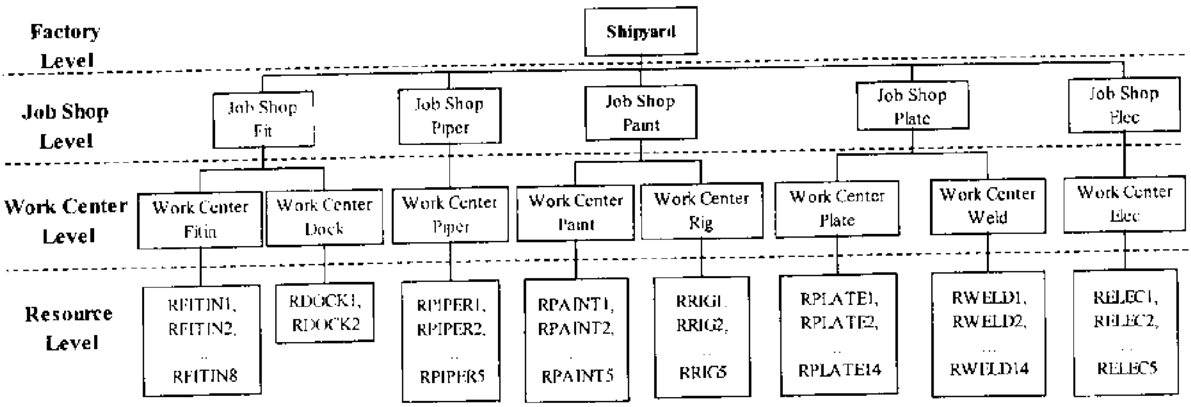


Figure 3. The model of the shipyard

Table 1. Elements of the shipyard model

Job shop	Process Description	Work Centers	Resources
Fitters	Fitting Jobs (interior of the ship)	Fitting	8 (each of these 8 resources includes 5 workers)
		Docks	2 docks
Pipers	Piping Jobs	Piping	5 (each of these 5 resources includes 4 workers)
Painters - Riggers	Blasting-Painting-Rigging Jobs	Blasting-Painting	5 (each of these 5 resources includes 6 workers)
		Rigging	5 (each of these 5 resources includes 12 workers)
Platers - Welders	Plating-Welding Jobs	Plating	14 (each of these 14 resources includes 6 workers)
		Welding	9 (each of these 9 resources includes 4 workers)
Electricians	Electrical Jobs	Electrical	5 (each of these 5 resources includes 5 workers)
5 Job Shops		**8 Work Centers**	**53 Resources**

Work Release and Assignment

Corresponding to the facilities' hierarchy there is also the workload's hierarchical breakdown. The Orders consist of Jobs which in turn consist of Tasks. The Orders correspond to the Factory and they are divided into Jobs which are released to Job Shops. A Job, based on its specification, can be processed only by one Job Shop and is thus released to the proper Job Shop. The Tasks that are included in a Job can be again processed only by one Work Center and are therefore released to the corresponding Work Centers. However, the Tasks can be processed by more than one of the Work Center's Resources and the assignment of a Task to a Resource is done with the help either of a complex decision making logic[1-3] or a simple dispatching rule.

An Order corresponds to the entire work that has to be done for the repair of a ship. The release of Jobs to Job Shops is based on the Job specifications. Similarly Tasks are released to the Work Centers (Figure 4). For example, all the blasting and painting work on the ship is one Job which is assigned to the blasting-painting Job Shop. The Job consists of a number of specific blasting and painting Tasks, such as sand-blasting the ship's hull, painting the hull, repainting the ship's name etc. The blasting Tasks are assigned to the blasting Work Center, whereas the painting Tasks are assigned to the painting Work Center. In each Work Center there are Resources which could perform the same Tasks. Therefore the Tasks have to be dispatched to the Resources according to an assignment logic, using dispatching heuristics or a multi-criteria planning mechanism that formulates and evaluates resource allocation alternatives[1-3]. An important constraint in releasing and dispatching of Jobs and Tasks is the precedence relationships among them.

The assignment of the shiprepair tasks to the shipyard's resources results in a schedule for each resource of the shipyard and thus a detailed plan and schedule for the critical parts of the entire shipyard is produced.

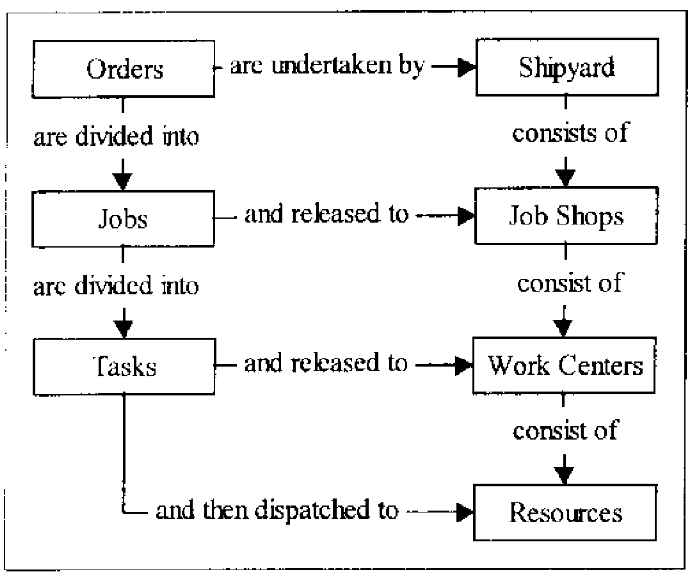


Figure 4. Work release and assignment

Results and Discussion

Software Implementation

The approach discussed above has been implemented in a software system with the help of Visual C++ version 5.0 (Win32 API) and it operates under Microsoft Windows 95 and Windows NT on a PC.

The system allows the user to construct an hierarchical model of the shipyard's facilities and their workload. The facilities model includes the definition of the Factory, the Job Shops, the Work Centers and the Resources. The user "fits" the workload model to the factory model, by specifying the Orders, the Jobs and the Tasks. Furthermore, the system allows the user to specify which Resources are suitable for performing each Task, the precedence relationships, the processing times and the set-up times. The system could include information on the cost for performing each Task and the processing quality. The graphic user interface is menu-driven, with win32 dialogues for guiding the user through the modeling process (Figures 5 and 6).

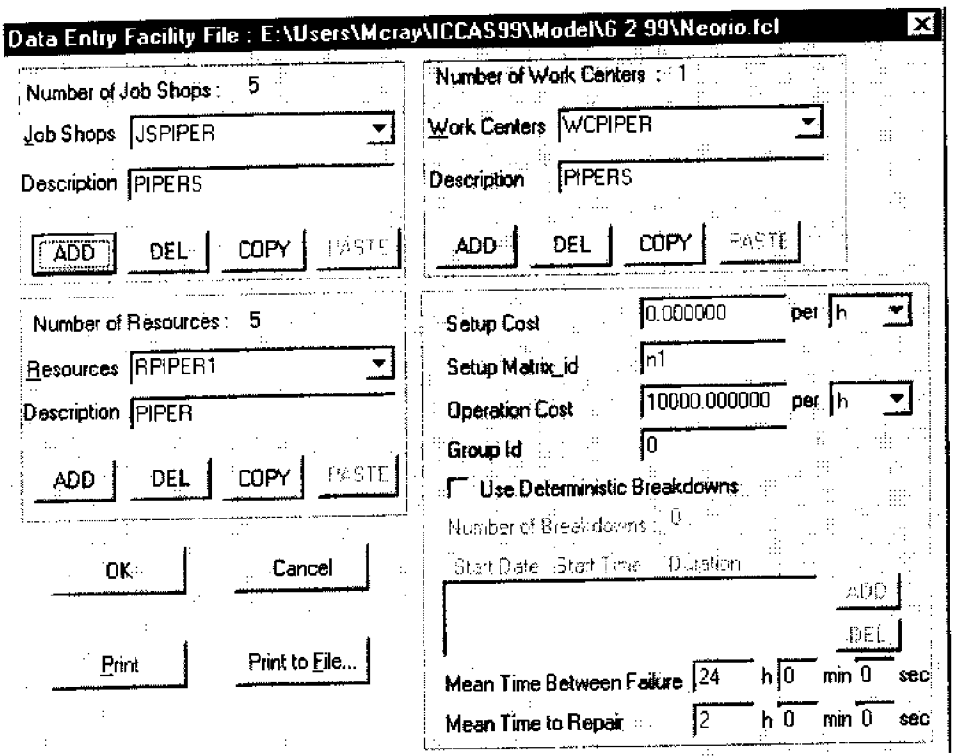


Figure 5. System's user interface: Facility data input screen.

A simple coding scheme could be used (based on the practice of the yard) for specifying the elements of the facilities and the workload model. As an example for the purpose of this work, the *Pipers* section is modeled as a Job Shop named *JSPIPER*, including one Work Center named *WCPIPER* which includes five Resources named *RPIPER-xx* where *xx* takes values from one to five. The workload for the Pipers' Job Shop consists of one Job named *JBPIPER* which includes three Tasks: *TP3110*, *TP3800*, and *TP3801*. The four-digit numbers in the task codes are the actual codes used by the shipyard modeled in this work.

The system uses event driven simulation to simulate the operation of the shipyard and the execution of the workload by the shipyard's resources. The simulation mechanism releases the workload to the Job Shops and Work Centers, respecting the precedence relationships which are defined by the user. In each Work Center an assignment mechanism decides which Task is going to be assigned to which Resource. Consequently, a dispatching decision is required when a Resource becomes available for processing. The assignment mechanism allocates the available Resource to a pending Task. The system simulates the operation of the production facilities either for a certain period of time (user specified) or until all the Tasks have been processed by the Resources. In either case, a detailed schedule for each Resource is produced in graphic or alphanumeric format (Figure 7).

The user has the option to select among a set of dispatching rules (Figure 6) and a multiple criteria decision making method. Furthermore, the system has the ability to consider planned maintenance in the production schedule and unexpected interrupts such as resource failures, which are statistically simulated. The schedule produced can be used for the planning of the activities of the shipyard.

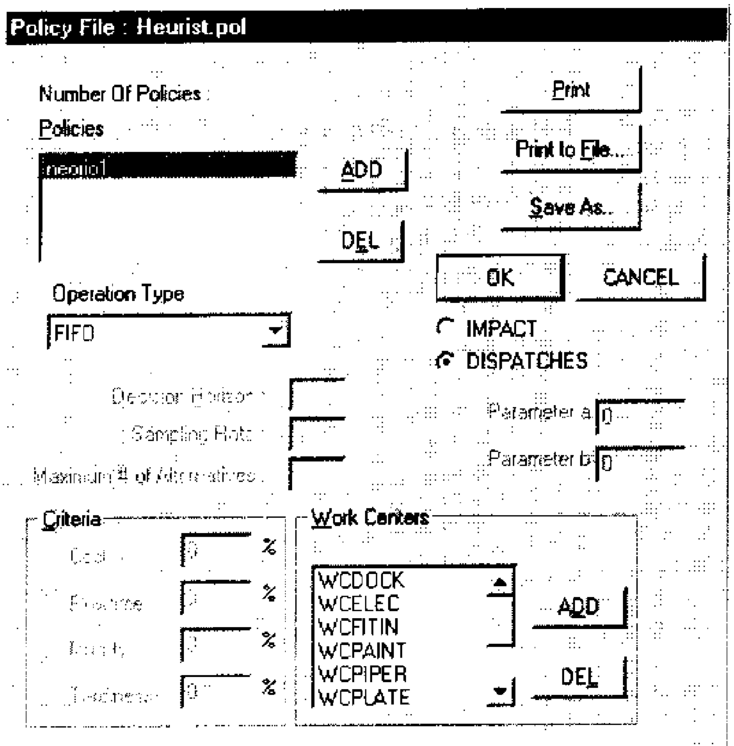
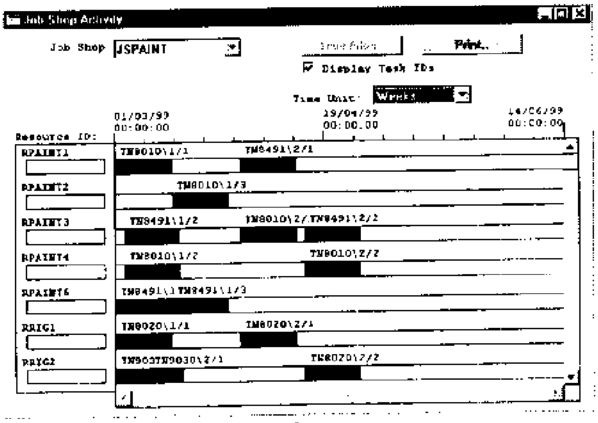


Figure 6. System's user interface: Selection of dispatching rule.

In case of a change on the workload or on the status of the facilities of the shipyard, the user can feed the system with the new data and reproduce an updated schedule and plan. Since a simulation run for a planning horizon of a few months takes a few minutes, it is very easy for the user any time that something unpredicted occurs, to reproduce efficiently an updated plan for the entire shipyard.

Resource Name	Work Center	Name of Task	Start Time	End Time
RPAINT1	WCPAINT	TN8010\1/1	10800.000	1134000.000
RPAINT1	WCPAINT	TN8491\2/2	3909600.000	5032800.000
RPAINT2	WCPAINT	TN8010\2/1	2613600.000	3736800.000
RPAINT3	WCPAINT	TN8491\1/1	10800.000	1134000.000
RPAINT3	WCPAINT	TN8010\1/3	3304800.000	4428000.000
RPAINT4	WCPAINT	TN8010\1/2	183600.000	1306800.000
RPAINT4	WCPAINT	TN8491\2/1	2613600.000	3736800.000
RPAINT4	WCPAINT	TN8010\2/2	3909600.000	5032800.000
RPAINT5	WCPAINT	TN8491\1/2	183600.000	1306800.000
RPAINT5	WCPAINT	TN8491\1/3	3304800.000	4428000.000
RRIG1	WCRIG	TN9030\1/3	432000.000	1123200.000
RRIG1	WCRIG	TN8020\2/2	3909600.000	5032800.000
RRIG2	WCRIG	TN9030\1/1	0.000	691200.000
RRIG2	WCRIG	TN9030\2/1	691200.000	1382400.000
RRIG2	WCRIG	TN8020\2/1	2613600.000	3736800.000

a



b

Figure 7. Schedule produced by the system : a. Alphanumeric format, b. Gantt Chart format

Experiments and Results

A set of experiments have been conducted in order to check the model's feasibility and produce realistic schedules for the repair shipyard. The quality of the schedule can be evaluated by the user via a set of performance measures, which are calculated by the system based on the produced schedule. The performance measures include job flow time, job tardiness, number of tardy jobs, capacity utilization etc. The job-related performance measures are calculated for each job and as mean values for all the jobs included in the entire workload. The capacity utilization is calculated for each Resource and as a mean value for each Work Center, Job Shop and the entire Factory.

For the experimentation a set of orders with different arrival times has been defined. The job due dates for the experiments have been calculated as follows:

*DD = A.T + k*P.T*

Where DD = Due Dates

A.T = Arrival Time

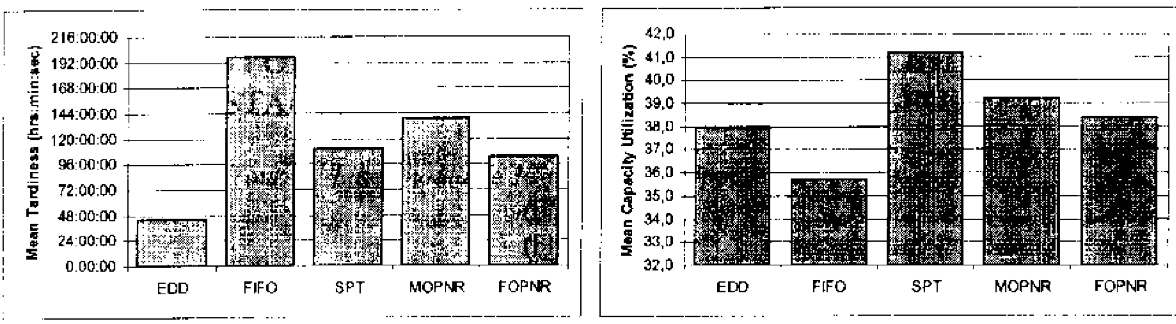
k = Constant

P.T = Processing Time of the Job

The constant k equals to k=1.3. This value of the constant results in a set of relative tight due date for the jobs. For the processing times, work and facilities related data were received from the shipyard.

A number of dispatching rules have been used for the experimentation with the model. The rules were applied for the allocation of Resources to pending Tasks for each Work Center. Each experiment includes one simulation run with the same facilities and workload data but with a different dispatching rule. Each simulation run produces a particular schedule and a set of performance measures specific to this run (Figure 8).

As was expected the EDD rule produces very good results for the mean tardiness, but performs poorly with respect to the capacity utilization. The SPT rule on the other hand, performs well with regard to the utilization. An interesting observation is that the EDD and FONPR rules produce similar results for most of the performance measures. This can be explained, considering that most of the jobs have comparable processing times and, since their due dates are calculated as functions of their processing times, the jobs with the fewer operations are the ones with the tighter due dates. Therefore the two rules result in similar assignment patterns.



Dispatching Rules

EDD: Task from the Job with the earliest due date is selected.
FIFO: Task from the Job which first arrives at the factory is selected.
SPT: Task from the Job with the shortest processing time is selected.
MOPNR: Task from the Job is selected which has the most operations remaining to be performed.
FOPNR: Task from the Job is selected which has the fewest operations remaining to be performed.

Performance measures

MEAN TARDINESS: $$MT = \frac{1}{N^{comp}} \cdot \sum_{j=1}^{N^{comp}} \max_j \left[0; T_j^{comp} - T_j^{dd}\right]$$

MEAN CAPACITY UTILIZATION: $$MCP = \frac{\sum_{j=1}^{N^{comp}} \left(T_j^{comp} - T_j^{start}\right) + \sum_{n_p}^{N^{proc}} \left(T - T_{n_p}^{start}\right)}{I \cdot T}$$

where N^{comp} : the number of completed Tasks

T_j^{comp} : the completion time of Task j

T_j^{dd} : the due date of Task j

T_j^{start} : the start time of Task j

N^{proc} : the number of in-process Tasks at time (T)

$T_{n_p}^{start}$: the start time of the in-process Task n_p at time (T)

T : the time at which the performance measure's value is calculated

I : the total number of Resources

Figure 8. Experimental results: Mean Tardiness and Capacity Utilization vs dispatching rules

Conclusions

The results of this work show that the method applied to the planning problem of the repair shipyard produces adequate and easy to use results. The method requires the modeling of the shipyard's facilities and workload with the help of an hierarchical model. The results show that depending on the dispatching rule used a schedule with different performance measures is produced. Thus the user can select the appropriate rule in order to produce a suitable plan.

References

1. G. Chryssolouris, "Manufacturing Systems – Theory and Practice", *Springer – Verlag*, 1992.
2. Chryssolouris, G., J. Pierce, and K. Dicke, "An Approach for Allocating Manufacturing Resources to Production Tasks." *Journal of Manufacturing Systems*, Vol. 10, No. 5, (1991), 368-382.
3. Chryssolouris, G. and M. Lee, "An Approach to Real-Time Flexible Scheduling." International Journal of Flexible Manufacturing Systems, Vol. 6, (1994), 235-253
4. R. Kolisch, A. Sprecher, "PSPLIB - A Project Scheduling Problem Library", *European Journal of Operational Research 96* , (1996), 205 – 216.
5. R. Kolisch, A. Drexl, "Adaptive Search for Solving Hard Project Scheduling Problems", *Naval Research Logistics 43*, (1996), 23 – 40.
6. R. Kolisch, R. Padman, "An Integrated Survey of Project Scheduling – Models, Algorithms, Problems and Applications", (1997)
7. M. Mori, Ch. Ch. Tseng, "A Genetic Algorithm for Multi – Mode Resource Constrained Project Scheduling Problem", *European Journal of Operational Research 100, (*1997), 134 – 141.
8. K. Y. Li, R. J. Willis, "An iterative scheduling technique for resource-constrained project scheduling", *European Journal of Operarional Research 56*, (1992), 370-379.
9. R. Cheng, M. Gen, "An evolution Programme for the Resource-Constrained Project Scheduling Problem", *Computer Integrated Manufacturing, Vol. 11, No 3, 274-287,* 1998

SHOP FLOOR CONTROL IN A PLATE CUTTING AREA

Hans Jørgen Lynggaard, Cell Control Centre, ODENSE STEEL SHIPYARD, Denmark

Abstract

The paper describes the information system for handling all steel plates through a cutting area at ODENSE STEEL SHIPYARD. During the last years OSS' robot cell control system, ROB-EX, has been transferred to all automated and manual processes in sandblasting, primer painting, plate marking, cutting and sorting processes including material handling between operations. Equipment in numerically controlled processes are linked to a shop floor control computer for data exchange with machine programming systems, and manual processes use wireless terminals with bar-code equipment for distribution and collection of information on the shop floor. The paper describes the information system which is located in the workshop and describes how it is used step by step in the plate cutting factory logistics. The shop floor control system handles all transactions involved in producing steel elements, i.e. it supplies production information for each element in each processing step and it registers each element as it passes through an operation. The system also automatically updates the production management system with status information from the workshops and it generates production reports about machines and parts. The implementation started a few years ago and now the complete system has been running in production for more than one year with good results. The benefits achieved is a more accurate planning and control of the operation on a horizon of one or two shifts. The results have been achieved without creating extra production management procedures, and even by eliminating some procedures. The system not only provides benefits for the area it controls and monitors, but also the assembly areas, which receives plates form the cutting area, are now able to check the status of each individual element on-line. The system has been developed by Odense Steel Shipyard and The Technical University of Denmark.

Introduction

Background

This paper describes development of information systems for the shop floor. It is a further development of the introduction of automated production facilities at ODENSE STEEL SHIPYARD. Computer applications for ship design an planning are now widely used by many shipyards and numerically controlled machines and robots are a natural part of some production workshops.

In the present context, some information systems today found in shipyards are of particular interest. Machine programming systems, or off-line programming systems, generates information for e.g. cutting machines and for welding robots. The information, including NC-programs and additional data needed for the production, is generated on beforehand, according to the production plan, and stored on data servers for later distribution to individual machines.

Another type of application of special relevance in this context is the production planning and management system. It specifies the production orders for each workshop including which input material is used for producing output parts. In addition it registers the status of all raw material, elements and assemblies.

These two types of systems have in common the dynamic exchange of information with the production activities on the shop floor. Depending on the parts being produced they deliver the data for the numerically controlled processes and they receive information about completed operations.

This paper focuses on information systems on the shop floor. One of the tasks of the information systems on the shop floor is to establish the interaction with the two types of systems described above. In addition the information systems on the shop floor shall support shop floor operations, i.e. support the production activities in the workshops.

The need for shop floor information systems

The need for information handling on the shop floor includes two main functions. First each workshop needs to interact with its surroundings and secondly the operation of each workshop can be supported by computer applications for managing and carrying out activities. Operators and foremen must retrieve the data they need for the current or upcoming production shift and they need to distribute data to individual machines. During the production run they need to collect information and generate reports for local use or for registration in a central production management system.

In a large production facility like a shipyard, and with a large number of unique parts being produced by different workshops, it can be difficult to obtain the actual status regarding each part. However, in many situations it is crucial for the planning and operation of one workshop to know the status of material produced by other workshops.

The approach for accommodating the needs

At ODENSE STEEL SHIPYARD, over a 6 year period, we have introduced so called cell control applications in different areas of the production. A system called ROB-EX has been developed originally for welding robots but has now also found its use for cutting machines and other automated and partly automated processes. Each cell controller is connected to the production management system. Since not all processes are numerically controlled, a bar-code system has been introduced for manual production processes. This is complementary to the cell controllers in order to collect data from all processes in the plate cutting area. Today the result is, that the status of all elements is collected by cell controllers and bar-codes and made available on computer terminals in the workshops which will be using the elements to produce assemblies.

The ROB-EX Cell Control system

The shop floor control system described in this paper is constructed by several cell control systems. Each cell control system serves an area of the production and thus includes functions which are mainly used locally. These functions will be briefly outlined in the following. However, the main topic in this paper is the effect reached by introducing several cell control systems in the production which will be described after this section.

Functionality of a ROB-EX cell controller

Each cell controller can link operators and machines to other areas and departments in the company. It provides information which is needed in the workshop. Also functions are needed in the workshop in order to enable local decision making. As an example, the production during one shift involves many NC-programs to be executed on several machines. In this case the cell controller assists the operator to decide the allocation and sequence of jobs to the individual machines and it transfers the right NC-programs in the right sequence to each machine. In the event that it is necessary to change the allocation and sequence, this is easily done on the cell controller.

During the operation the cell controller automatically registers events by each machine. A machine log is created for extraction of different reports as well as some of the event may immediately be forwarded to e.g. the production management system.

The technology elements in a cell controller

A cell controller is constituted by several different technologies. The cell controller is connected to different host systems and to the production equipment. For the communication, local area networks are used as well as different types of production equipment interfaces.

The cell controller uses a database to manage large amounts of production data. Often it also connects to other databases, e.g. it can be directly connected to the database of the production management system.

The cell control system is used by operators and foremen in their daily work. Therefore the user interface is based on either graphic terminals placed in the workshop or handheld computers which are wireless connected to the network.

The cell controller may contain calculation programs for e.g. machine calibration. In general it can be used for any type of information processing or analysis which is needed on the shop floor. Typically the operator needs some analysis or calculation in connection to the daily work, or a large amount of data is collected and analysed on the shop floor, and only the analysis result is forwarded to other departments.

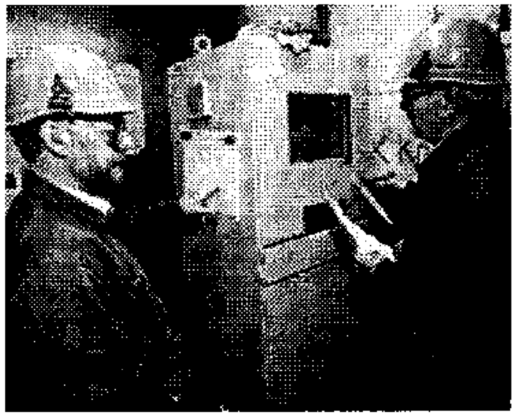
The cell controller can also be used for manual and automatic control of production activities. E.g. some operations for setting up a machines and downloading the NC-program may be completely automated or the cell controller can be used for manual remote control of several production machines.

Not only production equipment is found on the shop floor. Different types of identification and measurement systems may also be found. Therefore the cell controller also integrates e.g. bar-code systems and vision measurement systems.

The users of cell control

The philosophy of ROB-EX cell control is to make a system for the shop floor workers. It is a system specifically made for operators and foremen, see Figure 1.

Figure 1 The ROB-EX cell control systems is used by operators in the workshops



Results acheived with cell control

The ROB-EX cell control system has been in use at ODENSE STEEL SHIPYARD and other shipyards for several years. The first version was installed in production in 1993. Since then new functions have been added as well as new computer platforms and production processes are supported.

The system has brought several benefits to the users. First of all the cell control systems have enabled local decision taking and increased the possibilities for local action taking. By collecting information automatically the operators use the cell controller to form a general view of the situation very rapidly. In effect they are able to make decisions right when it is needed and based on the right information. Furthermore, the functions on the cell controller also mean that carrying out the new actions, following the decision, is done very rapidly. The cell controller assists the workers in the workshop to achieve high responsiveness to changes and disturbances. As a result, they are able to make the reality meet the goals due to the increased manoeuvrability.

The result can also be seen on the increased utilisation of production equipment. In a production environment where many different parts are produced, a significant fraction of time is used for changeover and setting up for a new production run. By quickly arranging and starting the work, the cell controller is able to make more efficient use of the equipment. A cell controller may in some degree be an alternative to investment in new machines and production space due to increased utilisation of existing equipment.

Finally, one of the consequences on introducing cell control on the shop floor is that the operators become more involved in the complete production process - which in itself has a positive effect on productivity and quality.

Further developments of cell control

The ROB-EX cell control system is a result of development by ODENSE STEEL SHIPYARD and by a research collaboration with The Technical University of Denmark. This approach combines a practical side with a theoretical side - and we continue to use an approach where academic research and industrial development joins efforts in the realisation of new systems. ODENSE STEEL SHIPYARD works with universities on improving the cell control applications and also in new areas such as the coordination between multiple production cells.

Deployment to other processes

The development of ROB-EX cell control started in connection with welding robots. ROB-EX means: robot execution system. Its main task was to support the execution phase of the robots in the production when the planning and programming had been performed in the technical departments.

The new processes

These so-called new processes was only new with respect to cell control. The production had been using numerically controlled equipment for plate and profile cutting for years. From 1995 to 1998 the ROB-EX system was introduced in the plate and profile cutting area at ODENSE STEEL SHIPYARD.

However, this area contained also a number of manual processes for post-cutting operations and material sorting. Therefore a new bar-code system was introduced in all non-numerically controlled operations for data collection.

Material flow management and monitoring

The introduction of cell control and bar-codes in the plate cutting area took place over a period of time. During this period the systems were mainly used for distributing NC-data to each machine or machine group and for generating production reports. Similarly the manual operations used the bar-codes for reporting progress of the production.

The manual operations and the numerically controlled operations are interconnected in terms of the material flow. The steel passes through several of these operations. Once each operation is equipped with either cell control or a bar-code system it is possible to monitor the progress of the steel on its way through the operations.

The plate cutting facility at OSS

The plate cutting area is producing steel plates and profiles for several assembly lines, see figure 2. It is receiving raw plates and profiles from the steel stockyard and it produces elements which are delivered in a sorting area or directly to the assembly lines.

The elements produced in the plate cutting area

The elements produced in this area are all contours which are cut out of the raw steel plates from the steel suppliers. Out of each raw steel plate one or several elements are cut. Figure 3 shows a typical element. Note that the element also contains a marking consisting of an identification code including a bar-code.

Figure 2 Plate cutting area

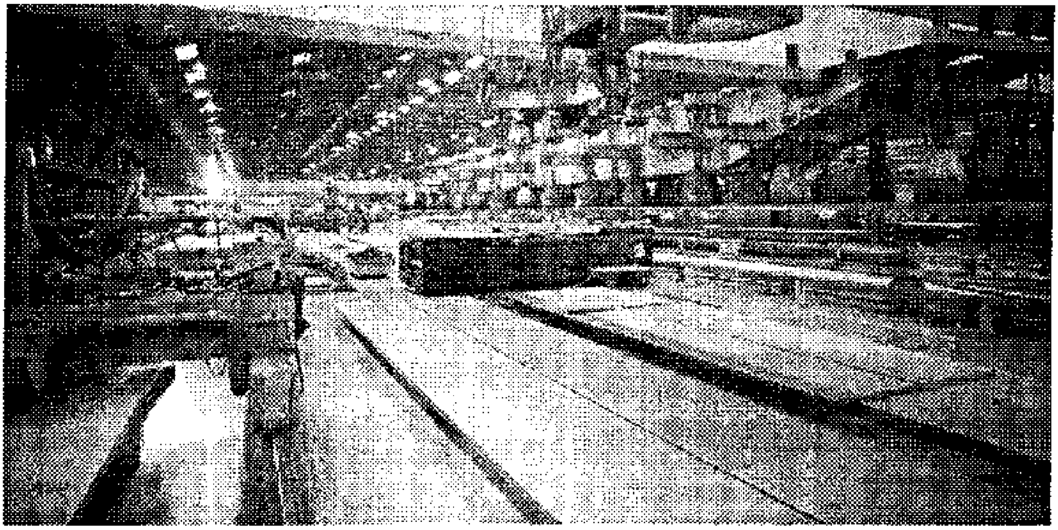
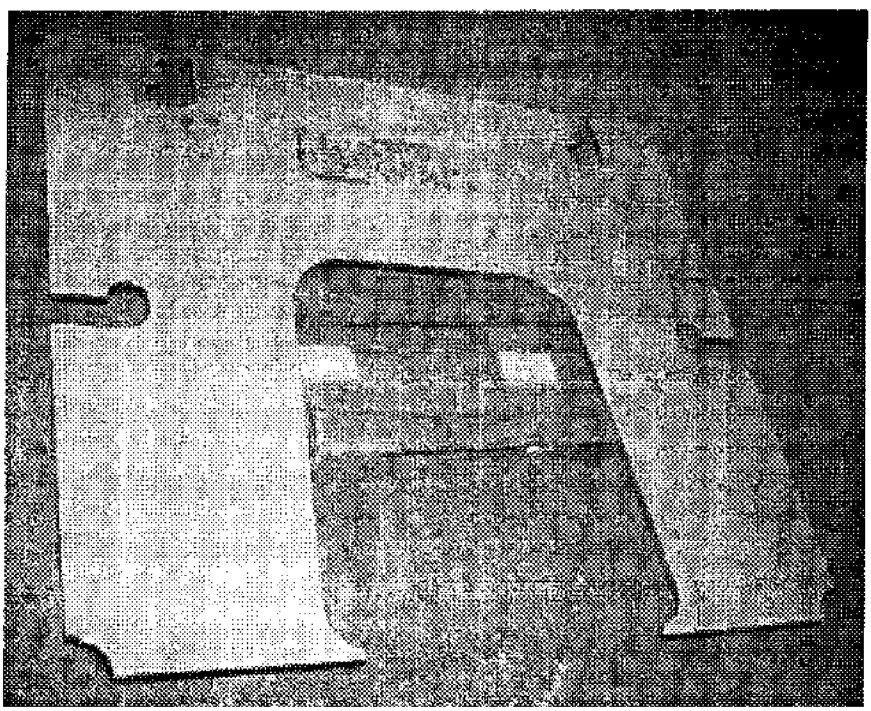


Figure 3 A plate element produced in the cutting area



The processes in the plate cutting area

The processes and the material flow of the plate cutting area are shown in figure 4. The diagram shows the main processes only. In total the area includes some twenty processing stations.

Figure 4 The main material flow and processes of the cutting area

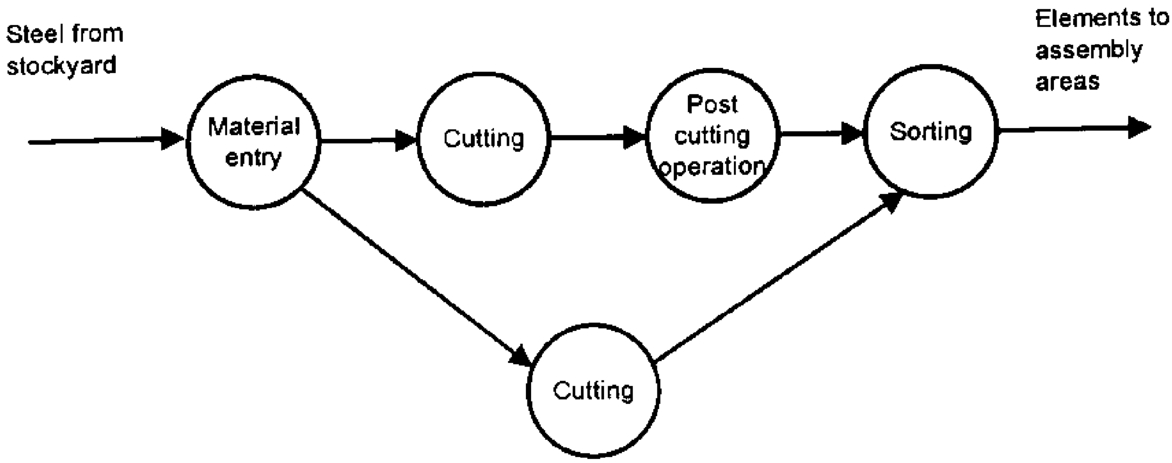


Figure 4 shows how the steel from the stockyard passes through a steel entry line and continues through one or more cutting machines. Some elements must pass through a post cutting operation. Eventually the elements ends up in the sorting area. From the sorting area the material is delivered to the assembly lines.

The steel entry line consists of several steps. First the steel plates are sand-blasted and surface treated with primer paint. Following that each plate is marked with identification code and a bar-code. Finally the plates are transported and stacked automatically to an area where they can be picked up by cranes depending on the cutting process the plates must undergo.

The cutting operations are manually or numerically controlled and the process used is either plasma or flame cutting. During the cutting operation the elements are marked with new unique identification codes including bar-codes. Figure 5 and 6 shows the cutting machines.

Some elements have to undergo post cutting operation, either because the cutting machine was unequipped to perform the complete process or because the process failed. The post cutting operation includes additional cutting or machining. Finally the elements are gathered and sorted by assembly in the sorting area.

Figure 5 Numercally controlled plasma cutting of plates

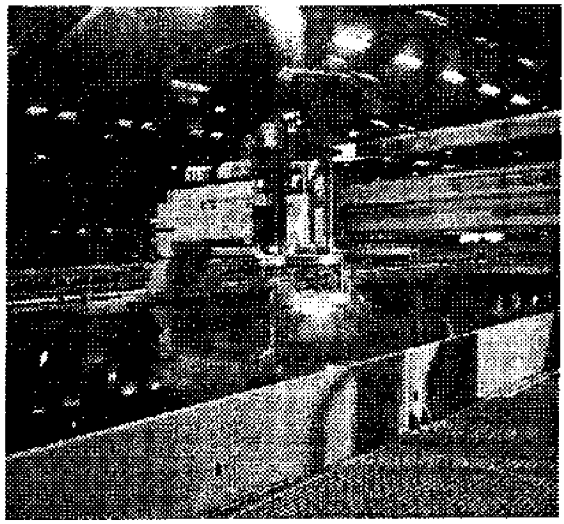
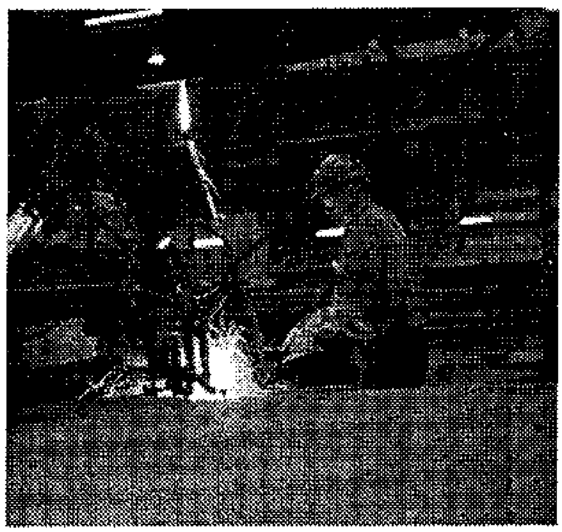


Figure 6 Manually controlled cutting machine



Information system overall architecture

This section describes the main systems involved in the shop floor control. Figure 7 identifies the systems and shows how they are connected.

Production management system

The production management system holds a record on each element and assembly of the ship. It also defines that elements are cut from raw plates and that the elements are joined in assemblies.

The production management system defines the process plan for each element. Similarly it can keep track of the status of each element.

Off-line programming

The off-line programming system generates NC-programs for the cutting machines. The NC-programs may be executed on different machines, however special post processing of data from the off-line programming system may be required for certain machines. The NC programs generated also contains data which controls the marking equipment mounted on the cutting machines.

PC for reporting

From any PC on the local area network it is possible to obtain production reports from the plate cutting area.

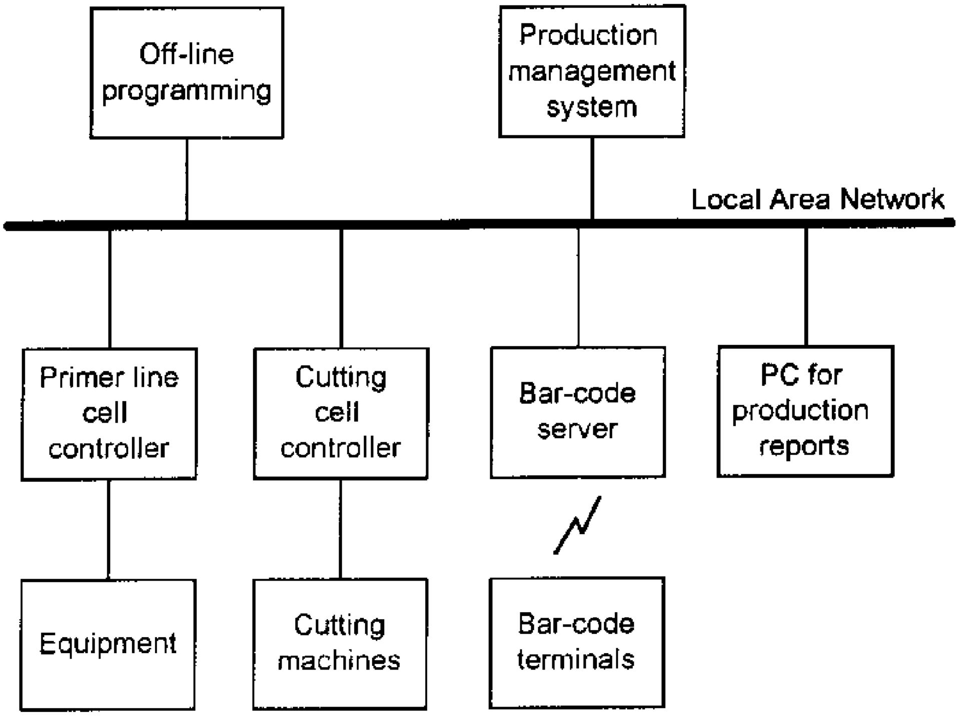
ROB-EX Cell Controller for numerically controlled machines

The ROB-EX cell control system has already been described in an earlier section of this paper. The control system for the plate cutting area described in this paper involves two cell controllers. One cell controller is used to control the steel entry line and another is used to control all the cutting machines. Each of the cell controllers are connected to the equipment for direct communication with the equipment controls.

The bar-code system

The bar-code system consists of a base station and a about a dozen bar-code terminals. These terminals are battery powered and are using wireless communication to connect to the base station. The handheld terminals consist of a visual display, a small keyboard and a bar-code reader.

Figure 7 Information system overall architecture



Operation scenario

This section demonstrates a typical operation scenario of the cell control and bar-code applications in the cutting area. It explains how the systems, described above, work in different stages of the production process, also described above. The description focuses on how different transactions are handled by the cell control and bar-code systems as the steel passes through the processing steps.

Steel entry

The foreman uses the cell controller to define a list of plates to be introduced from the steel stockyard to the cutting area through the steel entry line. The cell controller keeps track of all the plates currently being processed on the entry line.

As this stage each plate is identified and the cell controller for the steel entry line automatically notifies the cell controller for the cutting area that plates are on their way. Then, the cutting cell controller automatically retrieves the NC-programs needed for each of the plates. In this way the cell controllers collaborate and coordinate activities according the actual status on the shop floor.

As the steel undergoes sandblasting, primer painting and marking with identification code, the cell controller makes sure that the processes are adjusted to each plate and that the plate is marked with the correct identification code.

Finally the cell controller also supervises an automatic material handling system, and it controls the automatic delivery from the steel entry line to a certain position in the cutting area, depending on the further processing.

At this stage the plates are changing status from being 'in stock' to be 'ready for cutting'. This is automatically reported to the production management system which then can generate reports on the status of individual plates or on the total amount of plates entered to the cutting area over a period of time.

Cutting

The plates are transferred from a plate stack to the cutting machine by crane. The plates are identified by the code applied in the steel entry line. According to the plate identification the NC-program for cutting the plate is transferred automatically from the cell controller to the cutting machine controller. The cutting process may now begin. The cutting machine is equipped with marking equipment and during the cutting process the raw plate is marked with new identification codes for each of the element cut out if the raw plate.

The cell controller continuously monitors all the plates being processed on the cutting machines and the progress of the work. Once a raw plate has been cut into one or more elements the cell controller automatically registers the completion. At this stage, the raw plate no longer exists and may be marked as such in the production management system. Instead each of the elements cut out of the plate are now registered as having completed their first operation.

Some of the elements just produced have post cutting operations to be performed. Either because the process plan requires that they undergo manual processing after cutting or because the cutting process failed and the element must be reworked or repaired. In this case the bar-code systems is used to identify these elements and register their next operation.

Post cutting

Typically the post cutting operation is a manual process. Therefore the bar-code systems is used for data collection. Once the post cutting process is completed each element is identified and registered as completed in the production management system.

Sorting

After the elements have been produced they are gathered in the sorting area. Elements are sorted by assembly. Once all the elements for a particular assembly are checked in the sorting area by bar-code, they are registered as 'ready for assembly' in the production management system. At this stage the elements are ready to be transported to the assembly areas.

Production reports

Throughout the process described above the cell controllers and the bar-code system collect information on each element as they pass through the operations. The collected data is stored in a local database on the cell control system. Some of the information is immediately reported also to the production management system.

The collected information is used to produce production reports of different type. Typical reports contain information about the number of plates processed by a machine or a group of machines over a specified period of time. The foreman responsible for a machine group needs to know the amount produced on his machines during his shift, whereas the production manager typically want a day-by-day report with a weekly summary of the total production.

Other reports may be used to analyse the logistic in terms of work in progress levels and lead-time of the elements. Also the quality performance may be indicated by the number of redirected elements to unplanned post cutting operations.

In terms of the machine utilisation the collected data also shows the distribution of work between the individual machines and the accumulated processing time of the machines.

Finally, specific enquiries on particular elements are possible. Perhaps an operator want to locate an element or want to find out which machine produced the element.

Benefits achieved by using the shop floor control system

The introduction of the shop floor control system described in this paper has had a significant impact on the daily operation of the cutting area. In addition the system has also provided a much better service for the areas which receives the elements produced in the cutting area.

Benefits for the operation of the cutting area

One main benefit achieved is the automatic data collection and report generation. Without adding any extra work procedures (actually some manual procedures have even been eliminated) a number of high quality production reports are now created automatically. In addition these reports are generated immediately as events in the production occur. The presence of detailed information with no delay provides a good base for more efficient management of the plate cutting area.

As an example, it is very easy to keep track of work in progress, element by element, and based on that decide on how to organise the production activities within a shift. And during a shift it is possible to see the status and progress, take action accordingly and see the result in the reports.

The shop floor control system has also been able to eliminate some paper based procedures by handling the information electronically. The electronic handling of the data also mean that more flexibility is achieved e.g. in re-distribution of NC-programs between machines when production changes require cutting operations to be moved from one machine to another.

Benefits for the assembly areas

One major benefit for the area which receives elements from the cutting area is that the responsible foremen in the assembly areas can be notified immediately when the elements are produced. Elements needed for a particular assembly are reported ready in the sorting area without delay. Furthermore, the foreman can also check for each element, i.e. find out their progress in the cutting process and thus take the actual status of the material into account when planning the assembly operations.

This feature of the shop floor control system saves time for the preparation of the flowing assembly operations because they very quickly can obtain the information they need about the elements for each assembly.

Conclusion

This paper has described an information system for control and management of production activities on shop floor level in a plate cutting area at ODENSE STEEL SHIPYARD. The shop floor control system handles the exchange of information between the shop floor and the technical and administrative systems. It also supports the operation on the shop floor.

The shop floor control system is built up by several cell controllers, as used by e.g. robot cells elsewhere in shipyards. By using the same concept and open technologies it has been possible to reuse the technologies from the robot systems. For the same reason it has been possible over a period of time to build a highly integrated and homogeneous system.

By combining several cell controllers and a bar-code systems a shop floor control system covering the entire plate cutting area has been established. The results achieved are increased efficiency in the plate cutting activities and less disturbances in the coordination of the delivery of steel elements to assembly areas.

Acknowledgements

This paper has described a highly integrated information system constituted by many contributions. Therefore the author wishes to acknowledge colleagues from the Cell Control Centre of the Automation Development Department at ODENSE STEEL SHIPYARD which by their expertise in different technologies have played an instrumental role in the realisation of the cell control and bar-code systems described in this paper.

Also colleagues from the company Maersk Data who are involved in the off-line programming of cutting machines and the local area network installation, and colleagues working with the production management system are acknowledged for their role.

Finally, the users and production management who initiated and supported these developments are acknowledged for their role.

References

1. Neckelmann, Ole Knudsen (1997), "Industrial Vision", Ph.D. thesis, December 1997, Technical
University of Denmark, Odense University, Odense Steel Shipyard Ltd.

2. Lynggaard, H.J., "A Generic Cell Control Technology for Automated Shipbuilding", Proceeding of the ICCAS '97, Edited by K. Johansson and T. Koyama, October 13-17 1997, Yokohama, Japan

PIPEFAB
WORKFLOW MANAGEMENT FOR THE PIPE PRODUCTION

Walter H. Thomsen, CARETRONIC Ingenieurbüro GmbH, Lübeck, Germany

Rationalization in Shipyard Industry and Equipment Engineering

It can be observed for quite some time now that strict realization of technical innovations in working processes and tools contributes greatly towards rationalization, which in turn affords significant competitive advantages for both shipyards and shipyard suppliers.

In the field of Pipe production, integration of design, preparation, fabrication and installation has furthered great potentials in saving costs and finally led to „industrial" prefabrication of pipelines.

The success of such efforts has become visible now and the experiences gained can be evaluated and summarized as follows:

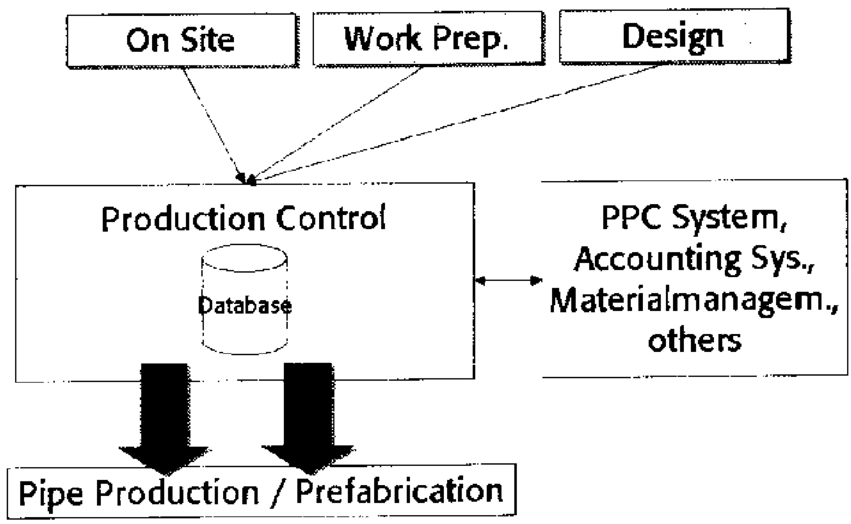
The consequent integration of virtually all working processes for realization of a pipeline system, whether it is in the field of shipbuilding, in chemical or petrochemical plants or in underground installed supply lines, is the decisive way towards rationalization. To this end, computer-aided data processing systems are equally important as direct processing tools.

Nature, extent and the time of data collection play an important role with respect to the cost-saving potentials. This applies in particular to equipment engineering, a field that is yet dominated by handicraft.

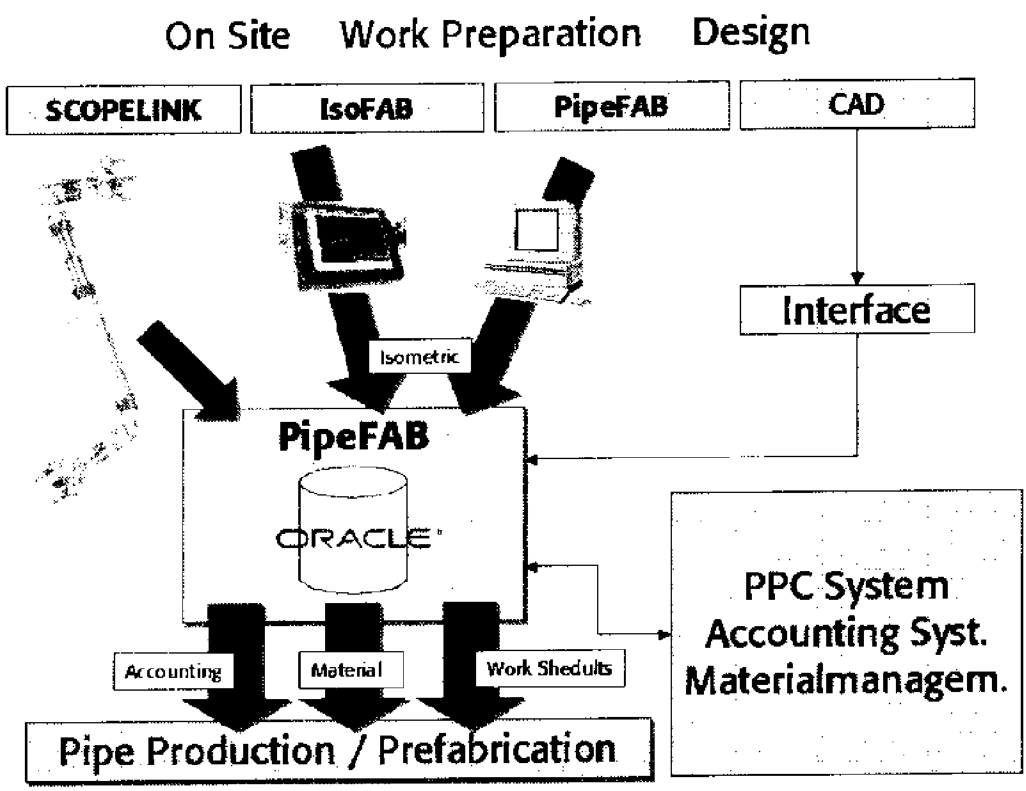
EDP systems suitable for shipyard industry and equipment engineering must:

- ☑ start early in the production process and
- ☑ be easy to operate and
- ☑ be of sturdy design and
- ☑ be able to store data strictly structured and
- ☑ be flexibly adjustable to changing requirements and
- ☑ be based upon accepted and future-oriented standards.

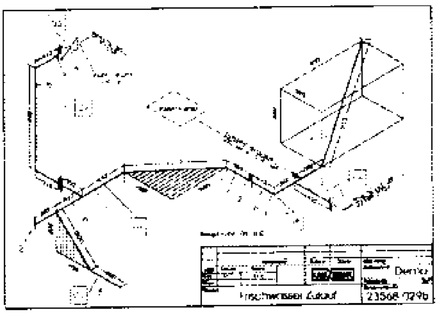
The fundamental configuration of a system can be derived from these requirements:



This system is realized by the PipeFAB- Productfamily!



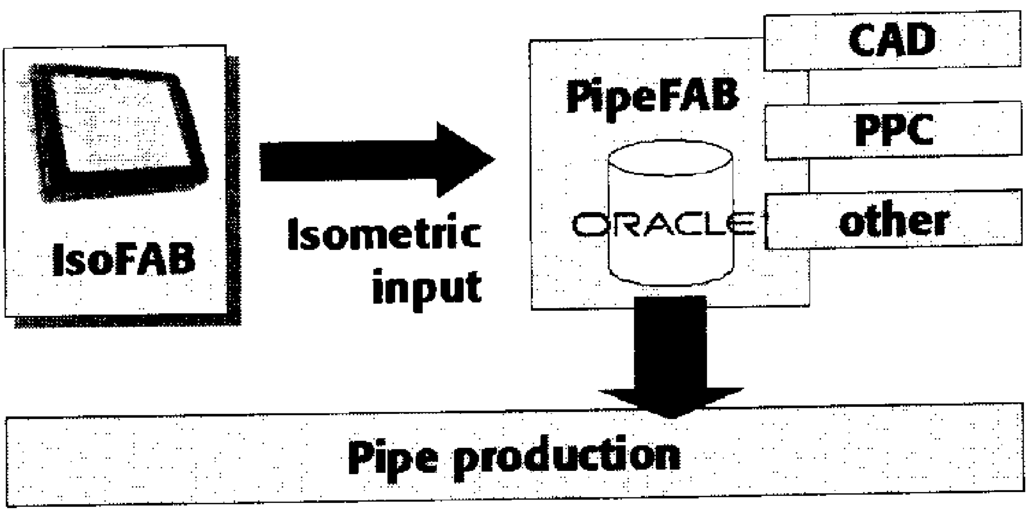
IsoFAB



Software Isometrics: Easy, fast and wherever you want!

- The Straight Way To Create Isometrics!

One way is the automatic production of pipeline isometries from a constructed product specimen by means of a CAD system and the subsequent processing of data in PipeFAB.



But the immediate computer-aided input of the Pipe course by the designer is frequently required.

IsoFAB supports the design by

- very easy and fast handling
- complete independent in use
- automation of routine functions
- direct database support
- with PipeFAB automatic creation of all fabrication data inclusive material- time- and cost information

IsoFAB - functions (Windows Version)

- Sketching of non-scaled isometry by clicking pixels with the mouse within a pre-selected drawing frame , or by direct input of coordinates.
- Assignment of Pipe class, diameter and semi-finished products by selection from the components´data base.
- Insertion of fittings and other components by selection from the data base, and clicking the point of insertion with the mouse.
- Completion of isometry by semi-automatic dimensioning of lengths and angles.

In addition

- Processing the display by Zoom / stretch and compress functions.
- Insertion of auxiliary geometry by drawing with the mouse.
- Completion of text with or without relation to individual elements of isometry
- Input of additional information (name of operator, date of preparation etc.)
- Automatic sectioning of line isometry into single isometry

IsoFAB is available with the waterproofed and shockresistent UniDAT OutdoorPC. It can be used in all conditions on site.

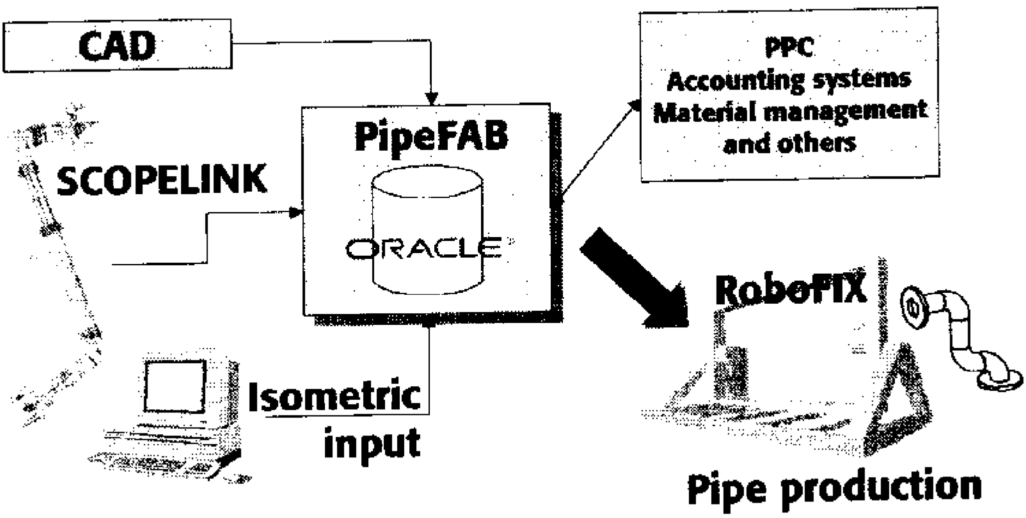
PipeFAB

Workflow Management For Pipe Production – Creation Of Isometrics, Calculation Of Production Costs And Management Information

PipeFAB was developed to rationalize the application and organization of production data for piping systems. All information are held in a relational databases. Pipe geometries are keyed in or taken directly from a CAD system: everything else is done virtually automatically. Connections to NC machines are available.

PipeFAB organises your materials and working times, calculates costs and provides our workshop with all the production documents required.

PipeFAB can not only be integrated into existing CAD and PPC systems, but also functions fully independently.



Fabrication in the part lists, sawing and flame cutting lists, bending lists, working schedules, manufacturing lots, disposition lists etc. are made available at the outputs. A bending collision check is carried out. All outputs are also available as machine-readable files.

PipeFAB makes the main functions, and in particular those for preparation and output of manufacturing data available in a complete system.

A calculation of all costs of material and productioning time is done !
While information on material and geometry is first linked in a data base with IsoFAB or an integrated CAD system, these linkages are evaluated by PipeFAB.

IsoFAB is useable being a stand-alone systemand offers all funktionalities for creating isometries founded on the database. It also includes the datatransfer to PipeFab and printing of the bills of material.

PipeFab is useable being a stand-alone system too an includes the entire functionality of IsoFab. In addition to that PipeFab offers all logical conditions for fabrication (working processes and organisational processes, advance times, prices of material and times, available machinery, order and optimization criteria, information on tools etc.)

PipeFAB contains all logical requirements of the production process (workflow information, working times, avialable maschines, criterias for optimazation, tools information and many more). All needed results are created automatically by calculations on the base of these informations and cataloques.

As a result all required data are to be printed out as lists or NC programs.

The Data Base

According to program technology, all modules of the FABRICATION product family are based upon a relational data base (preferably ORACLE), where all required information is stored.

The PipeFAB products are prepared to be integrated with all other ORACLE-based applications. This is the more important, the more comprehensive the existing EDP environment (PPC, CAD, accounting software etc.).

Utilization of existing data

If a relational data base if already available, it is possible to connect all PipeFAB modules directly. Any missing partial information can be subsequently added if required.

It is a rule that almost any enterprise holds great parts of required data available: They are stored in different formats (ASCII, text files, Excel tables, Word documents etc.) are distributed over a number of departments/workplaces, or they are stored in different data bases.

Experience has shown that by far the greatest part of these data can be automatically transferred into the relational data base by means of minor auxiliary programs.

Such auxiliary programs (scripts) are, in part, already available, or they can be easily established and adapted to individual requirements by CARETRONIC or by a staff member of the enterprise (system administrator).

Every kind of material cataloque can be transfered to the database.
Customers pipe class definitions can be intergrated.
All information of customers cataloques (material, fittings, indentification numbers, prizes and many more) are useable.

PipeFAB is prepared to read the IsoFAB data and to calculate the complete range of fabrication data of these isometrics.

3D Measuring And RoboFIX Are Parts Of The Overall Strategy Of The PipeFAB System

Advantages of the method are:

- Considerable savings in time because repeated transports between plant and workshop are no longer necessary
- The preparation of production models (basket, box, wire) can be dispensed with
- Saving of the entire material for the preparation of models (avoidance of scrap)
- Continuous and consecutive processing of any number of pipes on board ship and direct data transfer to the workshop
- Inclusion of the fitting pipe data into the shipyard's general EDP processing (material management, time, costs, etc.)

Result:

1. Considerable rationalization in the complete pipe production process
2. A speedey return of investment realizable already after having built less than 3 ships

3D Pipe Measuring

Treatment Of Fitting Pipes

As is the case with any type of fabrication, it can happen in the fabrication of piping systems that the manufactured and assembled system differs from the theoretical design. This may result from dimensional inaccuracies in pipe production (sawing, welding, bending, flanging, etc.), installation, as well as from the environment of the system (walls, ceilings, foundations, etc.).

As a rule, so-called "fitting pipes" are provided to compensate the overall dimensional inaccuracies by summing up all individual tolerances. Their task is to compensate differences between the design and the reality of a piping system. To this effect, their geometry is determined taking into consideration the real conditions in the ship. This means in fact that the course of the fitting pipe is only determined after the remaining part of the piping system has already been manufactured and fitted.

In principle, there are several methods available for the processing of fitting pipes.

The pipes already designed are:

- manufactured on site taking real conditions into account. Should documentation be required, it will be prepared subsequently on the basis of the sketches and charts established on site,
- manufactured in the workshop according to a model made in the plant, and documentation is prepared subsequently,
- prefabricated following manual determination of the situation in the plant and input of the data into the EDP, and being provided with extra length, adapted in the plant and completed. Here, the "as built" documentation is prepared subsequently as well,
- directly corrected to the dimensions which are automatically determined by means of 3D measuring and integrated into the normal process of workshop fabrication. The required documentation is prepared automatically.

State Of The Art

Until now it has been common practice for a steel model (either wire model or box model) of the pipe course to be made, which also demonstrates the positions of the flanges which are to be connected.

These are transported from the plant to the pipe workshop for further processing and/or manufacture. Or individual pipe components (partially prefabricated) are transported to the plant where they are cut, bent, if required, and completed by tack welding.

Both methods have a great number of common negative characteristics:

- Welding within the plant,
- transportation of welding equipment and residual materials,
- increased expenditure due to work regulations (flying sparks, ventilation, accident prevention, etc.),
- partly multiple transport of the material,

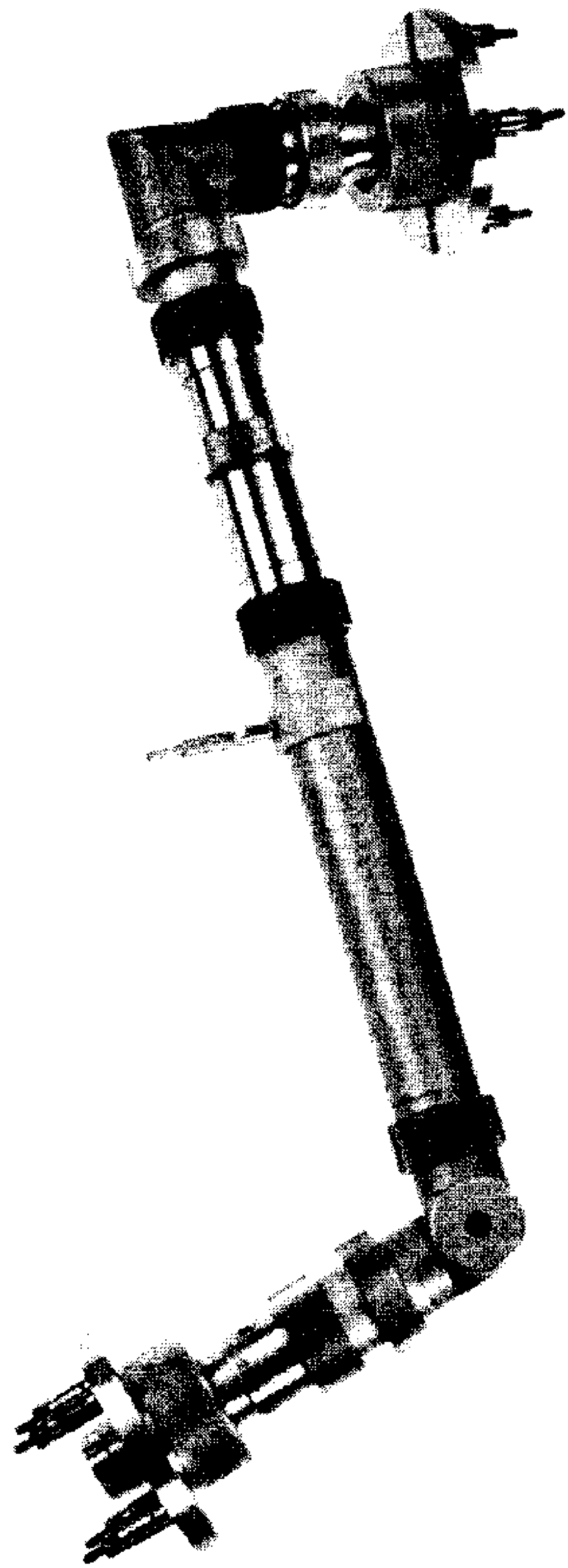
- accumulation of waste material (scrap),
- hardly avoidable distortions of the pipe, missing "as built" documentation.

Altogether, these methods are therefore too time-consuming and too expensive.

SCOPELINK

SCOPELINK is an electromechanical device for the determination of pipe course coordinates. It allows automatic measurement of all pipe coordinates within a plant, manufacturing of the pipe in the workshop and preparation of the "as built" documentation with reproducible accuracy.

To this end, the course of the pipe to be manufactured is modeled directly in the plant between two connection points (e.g. between two flanges), using plug-in components. This model is then evaluated on site by means of a portable computer.



Special features of the patented system are:

- easy three-dimensional modeling of the piping course immediately on board ship by means of plug-in components,
- simple shaping of fitting pipes of different nominal widths using the same components, including direct control of the actual outer pipe diameter,
- universal use of the system for differing pipeline courses by different combination of only three types of components (fixed and variable length components, variable angular components),
- quick connection to existing fixed points (flanges or the like) by means of universal adapters,
- easy handling by locking the components with bayonet catches,
- extremely sturdy design of all components as well as of the portable computer for use on board ship (humidity, knocks, temperature, etc.),
- undisturbed integration of the measuring transducers for lengths and angles into the device and direct output of the processed values to the computer,

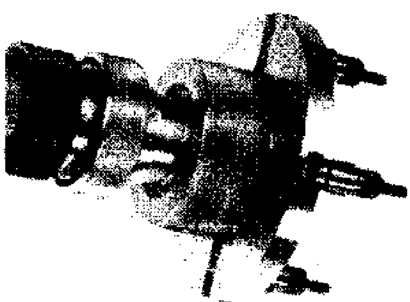
- immediate judgement of the measured values directly on site in a connected computer, including a suitability test for construction (usability of the available tools, bending cllision, selection of materials, etc.)

This ensures that a simultaneous visualization of planned Pipe courses and computer-aided measuring of the associated Pipe coordinates can be realized directly within the plant to be piped.

The systems to be piped can thus be defined right within the ship taking into consideration the existing marginal mechanical conditions and manufacturing tolerances in terms of suitability for fabrication, and the required manufacturing data can be transmitted to the workshop.

Brief Description Of The System

The measurement of lengths and angles is performed by means of integrated measuring transducers which are mechanically connected with the device, and transmit the required individual values for section lengths and individual angles via an internal bus system to a portable computer for evaluation.

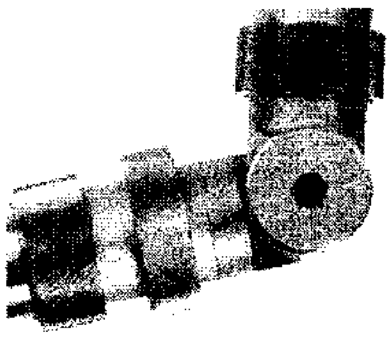


The connection of the device to the fixed points (fixed Pipe ends, machine connections, etc.) to be considered in the plant in each case is made by means of universal adapters, which are rigidly connected to the existing fixed points (e.g. flanges or sleeves). To this end, twisting angles (flange position) and angles of departure # 90° within the range of ± 5 degrees are taken into consideration.

The Pipe course between the fixed points is built up by facility segments.
The following types of segments are available for Pipe laying: transition flanges for the range DN 40 to DIN 400, straight segments of fixed lengths, straight segments of variable lengths and bent segments with adjustable angles.

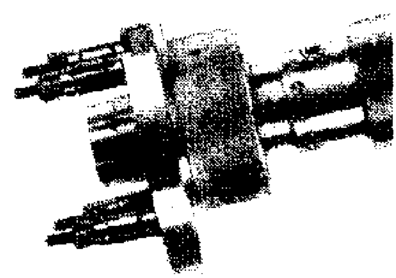
The segments in each case are plugged together as required by the planned piping course. During plugging, the segments are mechanically interlocked by means of quick-acting couplings. This is performed in defined positions to each other, so that unambiguous values are measurable both for lengths and angles.

Connection of the necessary segments and adjustment of the bent segments to the required bending angle are made manually.



The measurement of the lengths and angles is performed automatically by means of the integrated measuring transducer. Variable straight segments are fitted with linear displacement transmitters and variable bent segments with angle transmitters. Each of the segments used processes the measured values by calibration and transmits the actual lengths and angles via a bus system to the standard output interface.

The required hardware is integrated in the individual segments ("intelligent segments") and is also electrically interconnected by simple plugging of the components.



Clip-on segment rings facilitate a direct collision check of the planned piping course by comparing its actual outer diameter with the surrounding pipes, components or other parts of the plant.

Consideration can be given to branching pipes (fittings) by using additional devices.

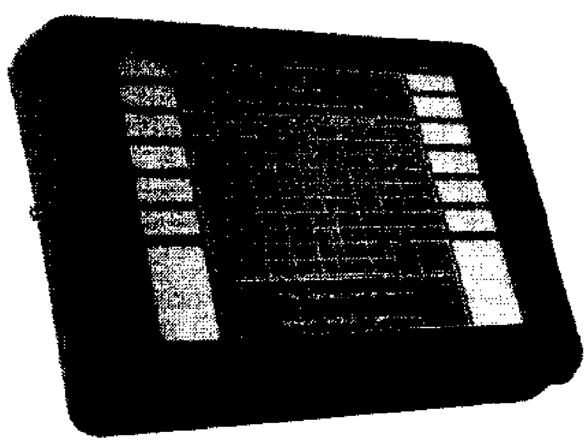
Once the measurement is complete, the device is dissembled and is again available for the next measurement.

Further Measuring Methods

Further measuring methods can be used, varying and depending on the individual case. These are 3D laser scanner for model wires or model pipes or photometric methods, which can be supplied by Caretronic as well and be integrated into the shipyard's system.

UniDAT OutdoorPC

The further processing of the actually measured Pipe data is performed by a connected portable computer, the outdoor PC UniDAT. This computer combines the functionality of an efficient office PC with the protection from humidity and knocks and blows occurring during harsh daily operation (work environment in the plant and/or ship).



The computer is protected according to IP 56 and can be operated without keyboard and mouse. The database as well as the respective measuring software is installed on the operating system Windows 95 and/or

Windows NT. Isometries can be prepared everywhere, independently and directly supported by the database

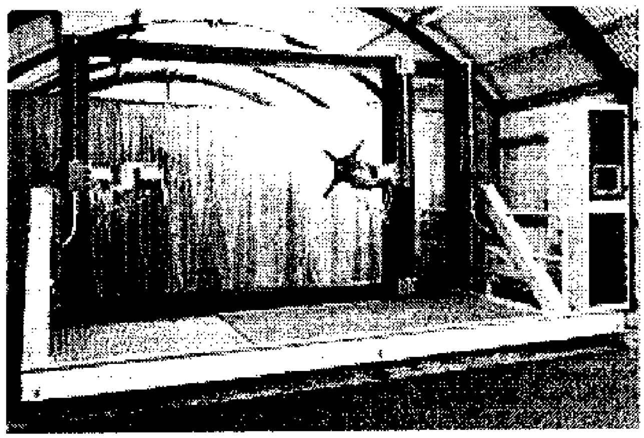
The data are transferred via network and radio telephone into the company's own system environment.

Availability of the PipeFAB software in the shipyard's EDP system is the prerequisite for the calculation and output of data meeting the manufacturing requirements and for triggering the machines

RoboFix Tacking Device

General

It is the task of the tacking device to define the flange positions required for Pipe production, so that prefabricated Pipe sections can be connected in the device with the flanges by means of tack welding. In addition, the tacking device allows checking of the Pipe course.



The tacking device can directly use the Pipe coordinates calculated by means of PipeFAB or determined by one of the 3D measuring systems.

Working Process

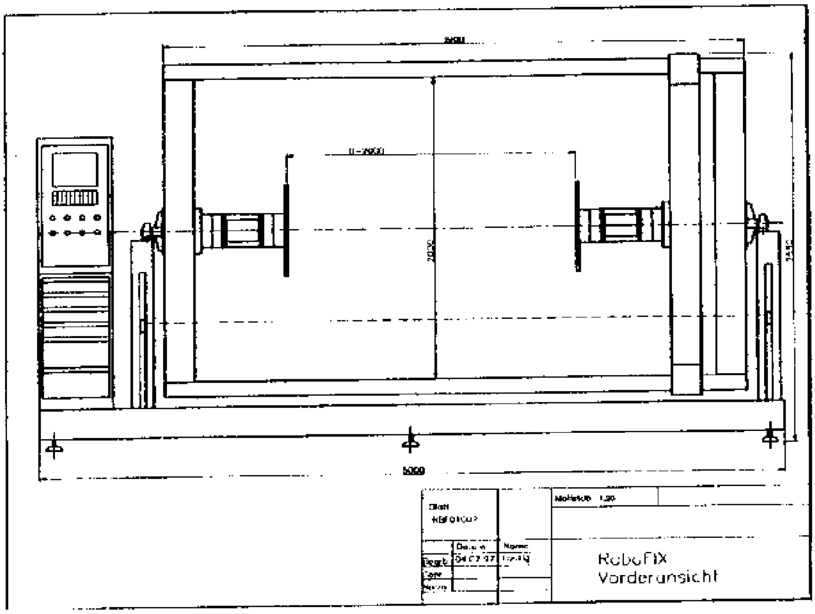
The Pipe data are loaded into the control computer of the tacking device either by means of a disk or via network connection. The control computer verifies that the permissible geometrical data are observed and performs a collision check in comparison with the device.

The 3D flanges of the as yet unloaded tacking device (without Pipe and real flanges) are then moved by means of an electric motor into such position to each other as if measured in the ship / at the building site. At the same time, the screen of the control computer shows the course of the fitting Pipe in a 3D display.

THE SCREEN DISPLAY SHOWS:

- The spatial course of the Pipe in the device
- The designation and position of the Pipe sections
- Reference dimensions to verify the Pipe course in the device

The "real" flanges of the Pipe are now screwed by means of the quick acting couplings onto the 3D flanges of the device. Subsequently, the prefabricated Pipe sections are positioned and fixed according to the screen display. The Pipe course can then be checked against the reference dimensions and corrected if required. The weight of the Pipe sections is held by external supports so that the accuracy of the system is not impaired.



The final step is to connect the Pipe sections by tack welding.

In order to make for better accessibility, the tilting frame can be tilted manually by ± 30°.

In case of overload of the 3D flanges, e.g. due excessive overhanging lengths, the flanges give in without being destroyed, and a message is given. Once the overload has been removed, the 3D flanges are repositioned by the control electronics.

Technical Description of the Components

Basic Frame

The basic frame is intended to accommodate the tilting frame and the electrical enclosure as well as the tool cabinet. It is a welded steel construction. The passable areas are covered with checkered plates or gratings. Alignment is effected by means of leveling feet.

The working area before the tilting frame is safeguarded by side rails. The front and rear side of the workroom are fitted with light barriers which signalize the presence of persons in the danger zone to the control and activate the safety stop as soon as the area of the device is entered.

Tilting Frame

The tilting frame accommodates the 3D flanges as well as the linear guides with the movable cross beam. The actual working area is located within the tilting frame.

In order to make for an unhindered access to the workroom, the tilting frame can be tilted by ± 30° from the vertical. This ensures that the fitting Pipe sections can be positioned in the device, even though geometry may be unfavorable. Fixing is made by means of two locking bolts, the position of which is monitored by the control. The tilting frame is supported by means of ball bearing pillow blocks on the laterally arranged stays.

The U-shaped horizontal carriers of the tilting frame hold the linear guides of the cross beam. The channel section is open on the side away from the workroom, so that the guides are protected from mechanical impact.

The control ensures that the positioning of the flanges and cross beam as well as entry to the working area are possible only when the tilting frame is fixed.

A safety circuit ensures that both the tilting frame and the flange can be readjusted only while in unloaded state.

3D Flanges

Essentially, the 3D flanges serve two purposes:

- Alignment of the flange level in each direction of an imaginary hemisphere
- Location and centering of the flanges to be tacked

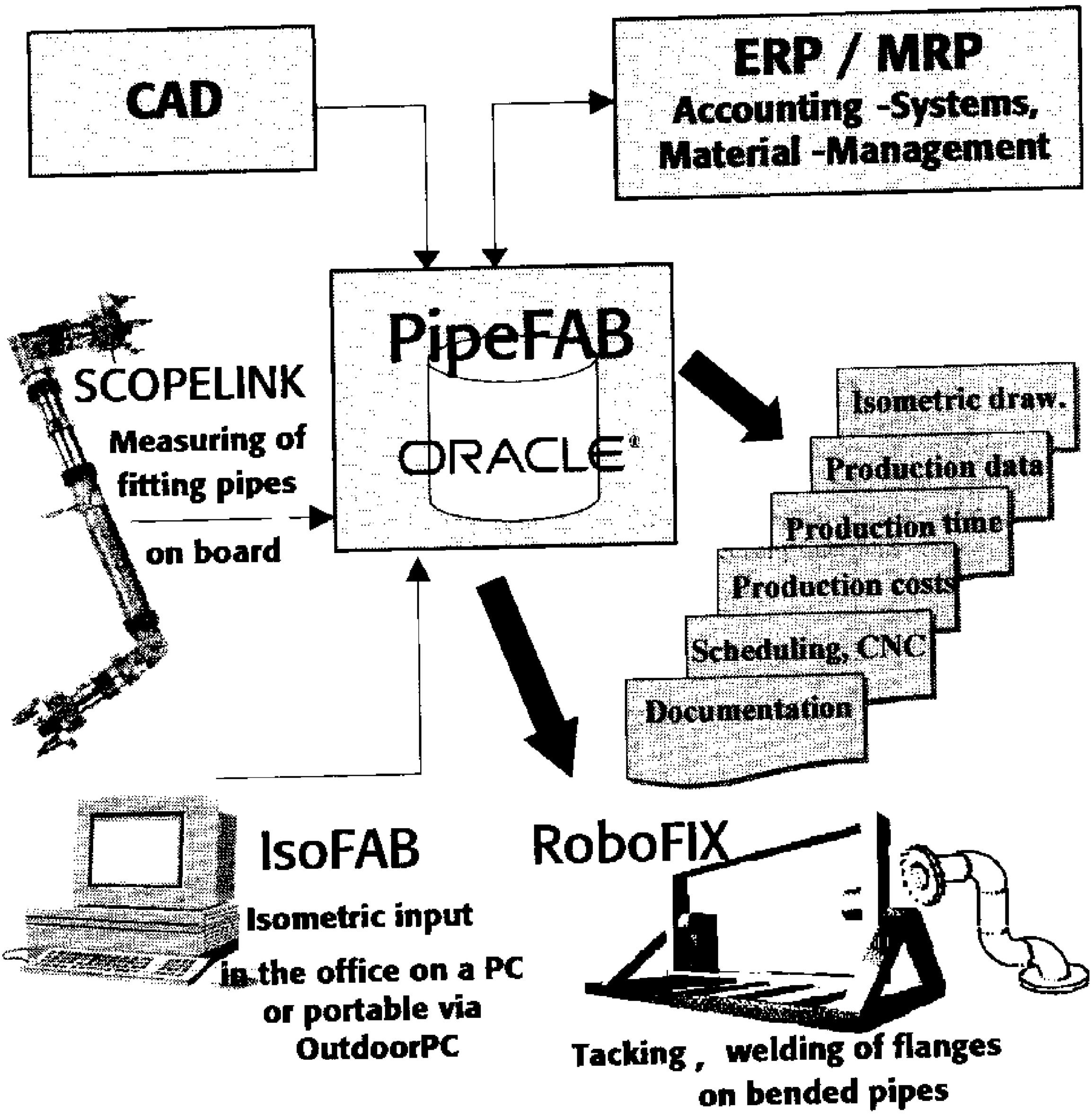
The 3D flanges consist of rotary plates mounted one behind the other. The rotary plate on the Inlet side is connected with the tilting frame and/or cross beam, while the rotary plate on the outlet side is fitted with the adapter plate which receives the flanges.

The rotary plates are equipped with drive mechanism for the rotary movement as well as with angle transmitters. The drive mechanism use automatic overload protections and are self-locking.

The control ensures that

- positioning can only be effected when the brakes are released
- the workroom can only be entered when the brakes are applied (otherwise alarm)
- the brakes are applied in case of failure of the AC mains supply.

PipeFAB -
The Production Management System



A HEURISTIC APPROACH TO OPTIMIZATION OF SHIPYARD MASTER SCHEDULE

D. S. Jeong, S. T. Yun, M. H. Jeong, Y. J. Heo, J. H. Cho
Hyundai Heavy Industries Co., Ltd., Ulsan, Korea

C. H. Lee, H. R. Choi
Donga University, Pusan, Korea

C. H. Park, K. R. Ryu
Pusan National University, Pusan, Korea

Abstract

The shipyard master schedule is a base plan for three to five year term shipbuilding processes that occur in all shipyard docks. The master schedule consists of a master action schedule and a master plan schedule. The master action schedule encompasses all projects that are already contracted as well as those in the process of being contracted. The master plan schedule plans for long term candidate projects based on the forecasted market demand, the company's production capability, and dock capacity. In this paper we present a heuristic approach to optimizing the master plan schedule. The master plan schedule deals with a large number of ships of various sizes and types to be constructed in docks of different capacities, with the primary objective of achieving maximum production efficiency within certain constraints. To cope with the computational complexity of this scheduling problem, we hierarchically decompose the problem into three subproblems of product-mix generation, dock arrangement, and dock-term adjustment. These are then solved heuristically by a genetic algorithm, a convex polygon algorithm, and a tabu search, respectively.

1. Introduction

The shipyard schedules are categorized, by their scheduling time and period, into the master schedule, the long-term schedule, the mid-term schedule, and the operation schedule. The master schedule is a base plan for coordinating all shipbuilding processes, based on which all the production process plans and the total work amount are determined. The master schedule also serves as a target indicator for setting up the long-term marketing strategies and managerial policies. The long- and mid-term schedules are mainly used for part ordering and part management, and for balancing the monthly workloads among various shops. The operation schedule is an executable schedule made up at production time by taking into account the workloads and the shop status at that time.

The master schedule consists of a master action schedule and a master plan schedule. The master action schedule encompasses all the projects that are already contracted as well as those in the process of being contracted. The master plan schedule plans for long term candidate projects based on the forecasted market demand, the company's production capability, and dock capacity. In this paper, we present a heuristic approach to optimizing the master plan schedule.

Master plan scheduling involves a determination of the types and sizes of candidate ships to be contracted, the respective docks where those ships are to be constructed, and the time period required for each of those constructions. Thus, a carefully-devised master plan schedule plays a critical role in maintaining stable production and achieving the company's sales goals. Yet, it is difficult to obtain an optimal master plan schedule because we must deal with so many ships of various sizes and types being constructed in many docks of different capacities, while attempting to maximize dock utilization and minimize load imbalances among the different docks and shops and also satisfy various constraints[6].

There have been attempts to optimize the master plan schedule using mathematical models[1,7]. These attempts were not successful because they demanded an excessive amount of computational resources due to the problem complexity. Consequently, the master plan schedules continue to be developed through time consuming manual processes relying on the domain knowledge of experts. If we can make available an automated scheduling system that runs in a reasonable time, we can use it to easily derive several different plan schedules based on different possible market forecasts and with different emphasis on various objectives. The comparative results thus obtained would make a significant contribution to optimal managerial decision-making. Therefore, the aim of this research is to develop a master plan scheduling system that can produce a near optimal schedule in a reasonable amount of time. We cope with the problem complexity by decomposing the problem into hierarchically organized subproblems and solving each of them heuristically in separate stages for computational efficiency. The developed system is expected to be used effectively and flexibly under various conditions, thus providing increased competitiveness in the shipbuilding business.

2. Problem Decomposition

To develop a master plan schedule, we should consider the target sales volume, the load distribution status among docks and among lower level assembly shops. Since there are several docks and assembly shops in the shipyard, the master plan scheduling problem is very complex. Therefore, we hierarchically decompose the problem into three subproblems: the product-mix generation at the top level, the dock arrangement simulation at the next level, and the dock term adjustment at the lowest level. Each of these problems is solved heuristically in separate stages for computational efficiency.

A product-mix is a combination of ships to be constructed in a certain dock during a certain dock term. The dock term of a dock refers to the period from the keel-laying date to the launching date in the dock. The dock terms are determined by experience based on the company's past production records of each dock. A product-mix is assigned to a dock term of a dock only when its constituting ships can all be constructed within the duration of the dock term. The objective of the product-mix generation stage is to assign product-mixes to the dock terms of all the docks in such a way that the assignment guarantees both a maximum sales volume and an even load distribution among the docks and assembly shops.

Sometimes, the ships of a certain product-mix for a certain dock may be impossible to be arranged in the corresponding dock. The objective of the dock arrangement simulation stage is to verify whether or not the ships of each product-mix can all be arranged simultaneously in the corresponding dock. If it turns out that the ships of a certain product-mix cannot be arranged in the respective dock, usually one of the constituting ships is replaced by a smaller ship so that the new mix fits into the dock. This replacement is currently made by the experienced expert.

The last stage of the master plan scheduling is the dock term adjustment. Its objectives are to smooth the load distribution among the docks and also among the preceding assembly shops, and to keep the number of launch time duplications within certain limits.

3. Master Plan Scheduling System

3.1 System Overview

Figure 1 shows the flow diagram of the overall system. First, the system starts by loading necessary data containing standardized ship types(crude oil tankers, bulk carriers, pure car carriers, container carriers, etc.), and ship classes. Each ship type has distinct work characteristics and resource usage patterns. The ship classes define ship sizes, such as e.g., 30,000 dead weight tonnage or 100,000 dead weight tonnage. For each ship type and class, we keep a database for standardized dimensions and resource requirements.

Then, the system determines dock terms for each dock and generates the product-mix candidates based on the company's past production records. A product-mix candidate consists of a group of ships that can presumably be physically arranged in a dock simultanesouly. Using those candidates and the standardized ship data, each dock term is assigned a product-mix. In this stage we apply a genetic algorithm[4] to find an optimal assignment which maximizes annual sales volume and at the same time minimizes load imbalances among the various shops.

Next, the system checks if the ships of each product-mix can really be physically arranged in the corresponding dock. For this purpose we use a convex polygon algorithm[5,8,10]. This algorithm effectively checks whether the priority rules of arrangement are followed and the technical constraints are satisfied. When there are product-mixes found that cannot fit into the corresponding docks, they must be replaced by other product-mixes which can maximally satisfy our objectives of sales volume and workload balance. However, since the search for optimal replacement at this stage is too demanding, we fix this problem in an easier way although not optimal; one of the ships of a problematic product-mix is replaced by a smaller ship but still considering the sales goal and load balance. This replacement is currently not done automatically by the system but by interaction with the human expert.

Finally, after the product-mix is determined, the system makes an adjustment of the dock terms in order to further smooth the load distribution among the shops and to keep the launch duplications under a certain limit. In this stage we use a tabu search algorithm[2,3] to find an optimal adjustment of the dock terms.

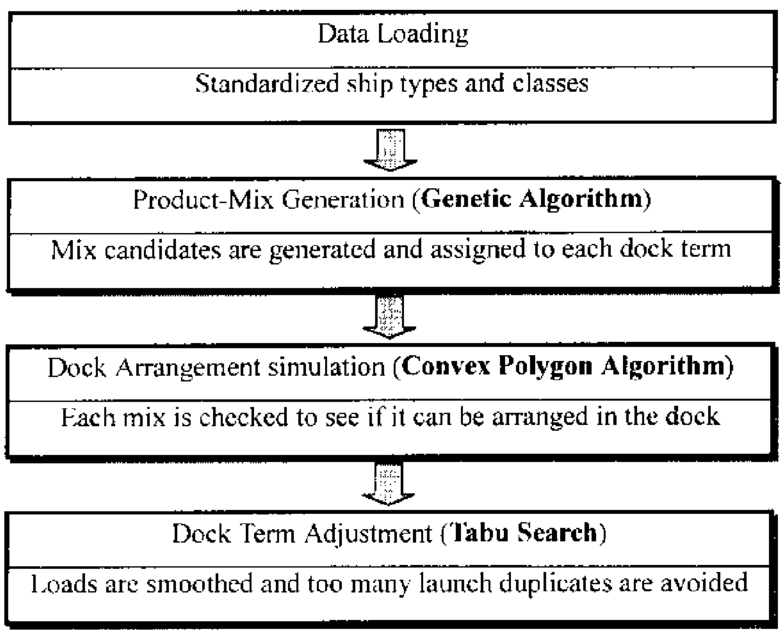


Figure 1. Flow of the overall system.

3.2 Product-Mix Generation

The product-mix problem is how to determine the list of planned projects. The annual product-mix is not easy to define because the construction period of a project spans more than one year. We assume that each project is keel-laid at the cycle start. The number of dock-term of each dock in the planning horizon should be determined. Other key events are determined by the standard lead times from the keel-laying date and the launching date for each project type and class. The number of projects for a year is counted on the basis of the keel-laying dates. The dock-term can be determined from the actual data, such as No. 1 dock is 60 net days or No. 3 dock is 55 net days. In this system, we are able to determine the period by a planner and calculate the calendar dates of each dock-term. Once the number of dock-terms is calculated into the planning horizon, the product-mix generation is to assign a combination of projects in a dock-term which can also meet the managerial performance objectives, such as the annual sales target or market demands. For each type and class of projects, we wish to have a desirable product-mix in terms of technical feasibility of construction in the yard, demand and price, and resource requirements and capacity.

In this paper, we use a genetic algorithm as an optimization tool to find a product-mix of each dock term. Genetic Algorithms(GAs) are heuristic search and optimization techniques that imitate the natural selection and biological evolutionary process. GAs combine the notion of survival of the fittest, random and yet structured search, and parallel evaluation of points in the search space. GAs operate by maintaining and modifying the characteristics of a population of possible solutions over a suitable number of generations. A generation algorithm consists of a string representation(genes) of points in search space, a fitness function to evaluate the search points, and a stochastic assignment to control the genetic operators. A solution is represented by a string of symbols, and is analogous to an individual. An objective function value called fitness is associated with each individual. The GA assimilates knowledge about the problem domain by iteratively producing and evaluating the individuals. In each generation there exists a subset of solutions referred to as the population. Some of these solutions are selected to be parents by a selection function. The selected parents are then recombined by the genetic operators of reproduction, crossover, and mutation to create an individual called the offspring. The set of all offsprings together with a subset of the parents constitute the next population. One of the advantages of the GA over traditional searching techniques is that it searches many points in the search space in parallel. The size of the parallel search is called the population size, which is equal to the number of strings in every generation. The population size is typically problem dependent and needs to be determined experimentally[4,9].

P_{11}	P_{12}	P_{13}	. . . .	P_{mn}

Figure 2. Encoding scheme of a candidate product-mix assignment.

To solve any optimization problem using a genetic algorithm, a coding scheme is necessary to encode the parameters of the problem into a string. In our product-mix problem, an individual is represented by a series of candidate product-mix for dock terms as shown in **Figure 2**. In the figure, P_{ij} is an index number that implies a product-mix which is constructed at the j-th term in dock i. And also, the number indicates when and where a product-mix is constructed. The count of the index that can be made in dock i is determined by physically counting possible arrangements of the ships that make up the product-mixes. In each generation, individuals are decoded to the workload and the annual sales by the resource data connected to the ships that make up the relevant product-mix P_{ij}, such as standard man-hour, working duration, model-curve, and etc.. Results of this stage, a product-mix and a distributed man-hour for each shops, are provided to dock arrangement and dock-term adjustment. To search for the optimal solution in rational time, we decided to use 40-populations and 200-generations experimentally.

We evaluate the suitability of an individual by annual sales target, workload, and market demands. The annual sales target is a managerial goal to achieve. The workload is levels of hull assembly, outfitting, and painting man-hours, and the market demands is a strategically estimated value that is determined for ship classes and types. Our objective is to minimize the workload imbalance among different docks and shops, and to maximize annual sales while satisfying market demands. We implement this multi-objective function using weighted factors within each objective. The annual sales target and market demands are reflected in this stage and the workload is optimized in the dock-term adjustment stage. A suitable solution for annual sales is evaluated by the deviation of the annual sales target and the system result sales. The lower the deviation, the better the solution. Of course, giving a penalty to an individual solution should not violate market demand conditions.

3.3 Dock Arrangement Simulation

It could not be guaranteed that all the ships composing each product-mix generated in the previous stage can actually be located in a corresponding dock. The dock arrangement simulator tries to search for a feasible locatable space satisfying both the physical constraints and the efficient production constraints. The physical constraints refer to the constraints such as the location of crane and related equipment, the location of already arranged ships, the minimal free space for operation, and the capacities of dock and equipment, etc., which should be satisfied. Efficient production constraints guide production efficiency. All the constraints related to the productivity such as the number of floating, the existence of a pre-erection area and the minimal dead space are included in the classification.

We have adopted the convex polygon algorithm that is used for spatial scheduling. To find the feasible locatable positions of a ship within a dock that does not overlap with already arranged ships, we can utilize the notion of configuration space[5,8,10]. Configuration space is the space through which the reference point of an object with fixed orientation can possibly pass without colliding with obstacles that are present. There are two kinds of configuration space: the obstacle-avoiding space and the inner locatable space. In the obstacle-avoiding space, the reference point of a ship can be located without colliding with the already located ships that are regarded as obstacles. In the inner locatable space, the reference point of a ship can be located within the boundary of a dock. Thus, the feasible locatable space can be derived by intersecting the previous two kinds of space.

Figure 3 illustrates the spaces where the object A_j is to be located within a rectangular space, in which two objects B_i and B_k are already located. Because the feasible locatable space is a continuous space, it is impossible to find all the points in it. To extract a set of meaningful discrete points out of the continuous space, we define the distinctive locatable point set, which consists of the vertexes of the feasible locatable space. Theoretically speaking, the distinctive locatable point set does not guarantee finding an optimal location. However, the points have empirically provided satisfactory locations with the advantage of computational efficiency[5].

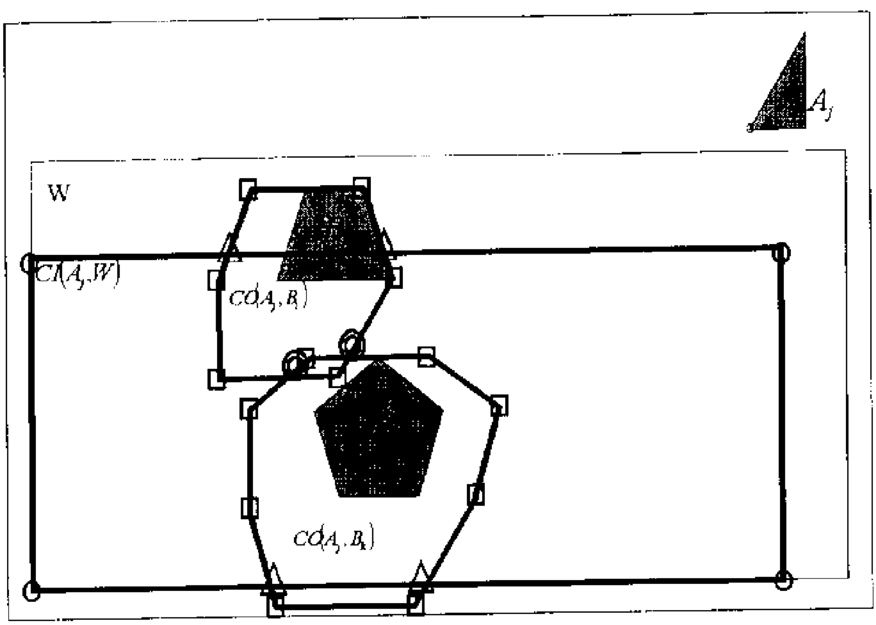


Figure 3. Feasible locatable space and distinctive locatable points.

The distinctive locatable points can be classified into four categories: 1) the union of the feasible vertexes of each obstacle-avoiding space; 2) the feasible vertexes of the inner locatable space; 3) the union of the feasible intersection points between the boundaries of each obstacle-avoiding space and the inner locatable space and; 4) the union of the feasible intersection points between the boundaries of each obstacle-avoiding space. These points are illustrated in **Figure** 3. The categorized points can be used to reduce the search space by identifying a special category of the distinctive locatable point set contingent to the situation. Among these points, we have selected one considering the various constraints mentioned earlier.

3.4. Dock Term Adjustment

The purpose of dock term adjustment is twofold. The one is to smooth the load distribution among the docks and also among the preceding assembly shops. The other is to keep the number of launch time duplications within a certain limit. When too many launches take place simultaneously in a certain period (i.e., on the same weekend) they cannot all be finished with the given manpower and equipment. Too many simultaneous launches are also likely to entail workload bursts in certain preceding assembly shops. Therefore, there arises the need for adjusting the start and end dates (i.e., the keel-laying date and the launching date, respectively) of the original dock terms through careful examination of the launch events and the overall workloads imposed by the product-mixes determined in the previous stages by the product-mix generator and the dock arrangement simulator.

We adopted a tabu search algorithm to search for an optimal adjustment of the dock terms. Tabu search, like the usual local search, begins from one candidate solution and moves to its neighborhood, proceeding iteratively until a given termination condition is met. However, tabu search goes beyond local search by employing strategies for avoiding being caught in a local optimum. When it moves from one solution to another in its neighborhood, it always takes the best move available even though the move gives a solution worse than the previous one. This is a diversification move for getting out of a local optimum. But since the search might quite possibly get back to that local optimum in the subsequent moves, there must be a means to force diversification for a few more moves. For this purpose, tabu search records its last *m* moves in a short term memory called the tabu list and prevents them to be repeated while they remain in the list. The tabu list is usually maintained on the first in first out basis[2,3,9].

E_1	E_2	E_3	. . . . . .	E_n

Figure 4. Representation of a candidate solution for dock term adjustment.

In our dock term adjustment problem, a candidate solution is represented by a series of dock term end dates as shown in **Figure 4**. In the figure, E_i is the positive or negative amount of adjustment made to the original end date of the dock term of the i-th product-mix P_i. Since launch events can take place only at weekends, the amount of adjustment is in the number of weeks. When the end date of the dock term of P_i is changed, the start date of the product-mix that is scheduled next to P_i in the same dock is accordingly changed. The maximum amount of adjustment that can be made is determined by referring to the maximum and minimum possible dock durations of the ships that make up the relevant product-mixes. Initially, the search starts from all zeros which means no adjustment. It is a reasonably good start point because the original dock terms of the entire product-mixes are determined based on the company's past production records. From that start point, each of the n neighboring candidate solutions is generated by replacing each E_i by a random amount of adjustment within the allowed limit. Then, the best of these n candidates is selected as the next current solution, and that move is recorded in the tabu list. From this new current solution, the search moves again to its best neighbor and continues iterating in this way until a good enough solution is found or an iteration limit is reached.

Although the tabu list provides a means to diversify the search to get out of a local optimum, we can further improve the search performance by introducing a stronger diversification strategy based on a so called frequency-based memory[3]. The frequency-based memory reflects a long-term search history by recording the number of changes made to each E_i in the moves selected so far during the search. When the rate of finding new best solutions falls below a certain threshold, the frequency-based memory is referred to in order to make a jump or a compound move so that a relatively less investigated part of the search space can be brought into attention. In our implementation of making a compound move, the least frequently changed 30% of all the E_i's are selected and then 60% of them are randomly selected. Those selected E_i's are then changed simultaneously to get the next candidate solution; this solution is used to restart the search. Our experiments show that the quality of the candidate solution reached by this jump usually worsens. Quite often, however, the proceeding tabu search finds a new best solution in a relatively short period of time.

The evaluation of each candidate solution is made by measuring the degree of load balance among the docks and the preceding assembly shops as well as by examining the number of launch events that are duplicated. The appropriate weights to be given to these two objectives are determined experimentally. Whenever a new neighboring candidate solution is generated from the previous one, the evaluation can be made incrementally rather than from scratch because in the new solution only a single E_i is different from the original solution. This incremental evaluation saves a lot of computation time and thus makes the tabu search much faster than is possible using other search methods such as genetic algorithms.

4. Implementation Results

We have implemented a master plan scheduling system using the algorithms explained in the above sections. During the implementation of each algorithm, we have performed various experiments to choose adequate value of parameters.

In the product mix generation stage, we have tried to find the proper parameter values of a genetic algorithm. **Figure 5** shows the results of experiments to compare fitness values of different combinations of population size and number of generations, the execution times of which do not exceed 2 hours. In the figure, for example, the case 'p100g50' seen around the middle of the horizontal axis indicates an experiment where the population size was set to 100 and the GA was allowed to run until the 50-th generation. With that setting, the graph roughly tells us that it took 800 seconds to give the final population in which the evaluation of the best individual was 395, the average 415, and the worst 440. We have chosen the combination of 40-population and 200-generations(p40g200) since the fitness value of that combination is fairly good while the execution time is reasonable.

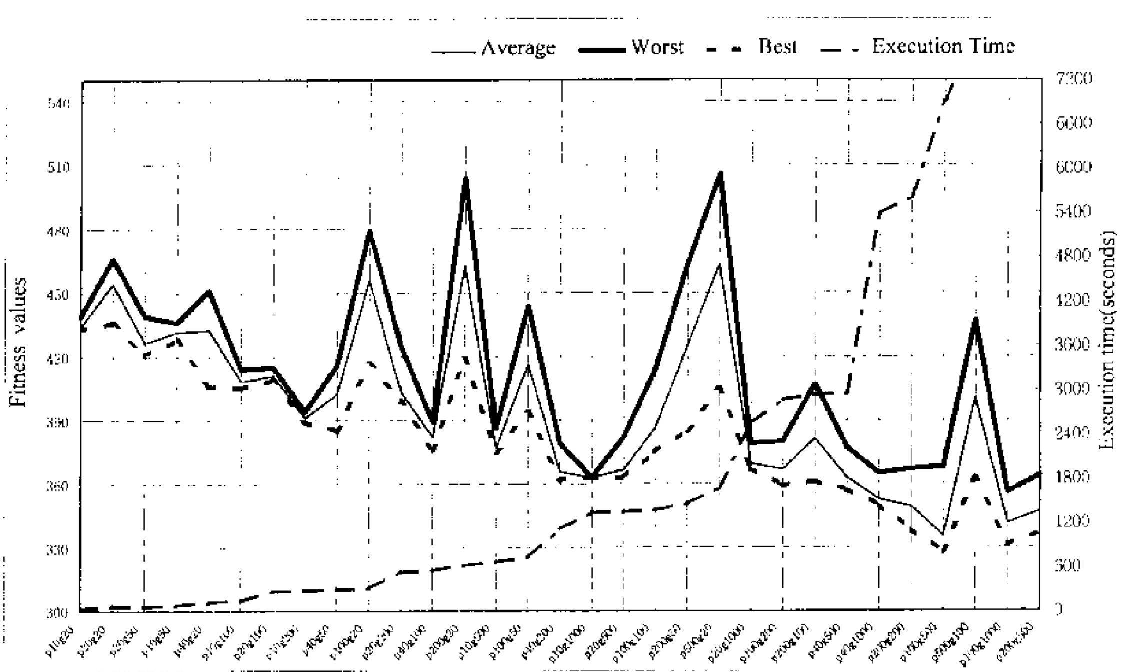


Figure 5. Effects of different population sizes vs. the number of generations.

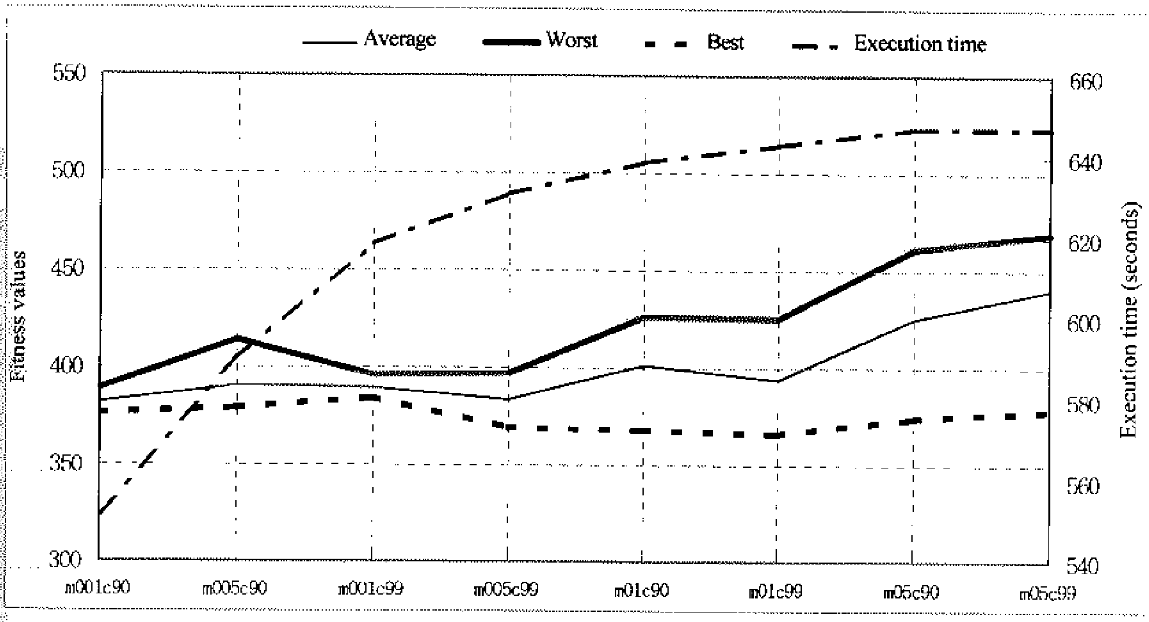


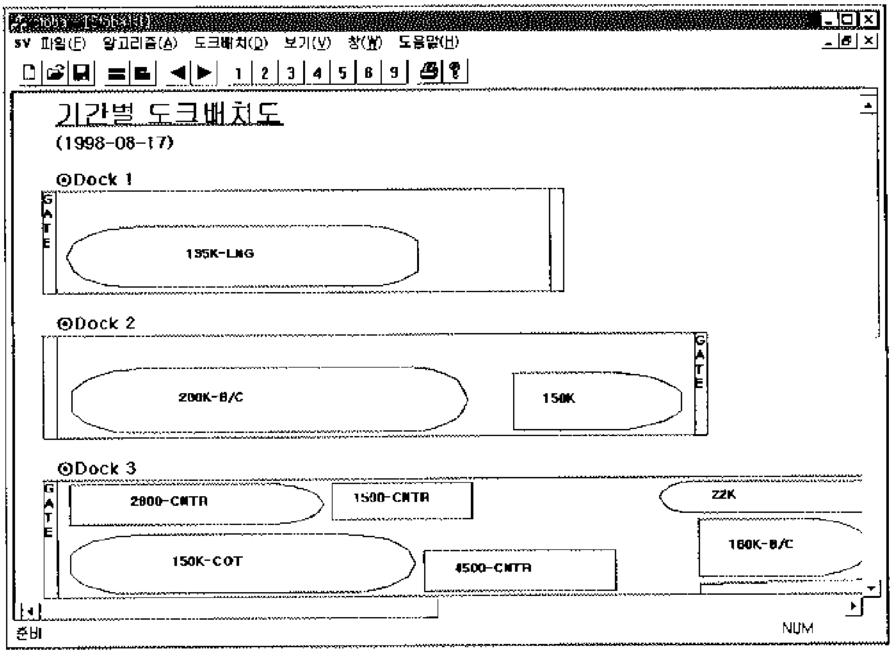
Figure 6. Effects of different combinations of crossover and mutation rates.

Figure 6 shows the results of experiments to compare fitness values of different combinations of crossover rate and mutation rate, the execution times of which do not exceed 650 seconds. In this figure, for example, 'm005c99' means the combination of mutation rate 0.005 and crossover rate 0.99. Among these combinations, we have chosen the combination of 0.99-crossover rate and 0.01 mutation rate since the fitness value obtained from that combination is satisfactory considering the execution time.

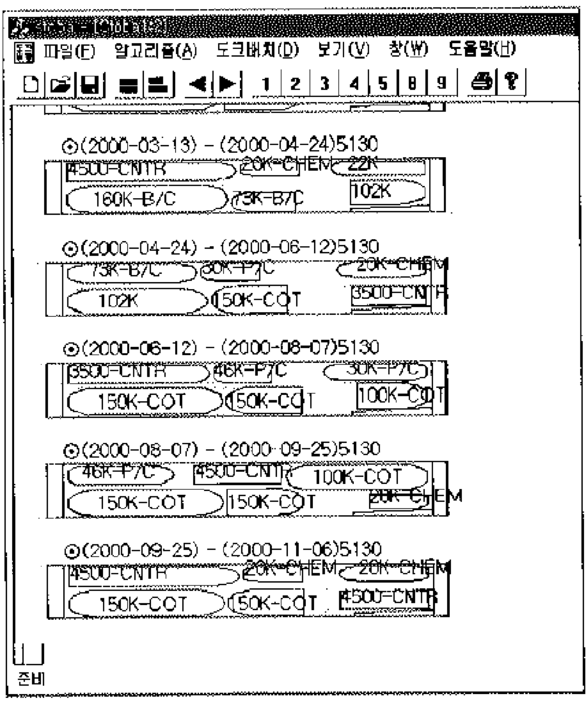
Figure 7 illustrates the results of dock arrangements: (a) shows a snapshot of the spatial layout status of ships in each dock for a certain day and (b) shows a snapshot of the spatial layout status of ships in a dock during a time interval. Ships that fail to be arranged can be replaced by other ships of similar size, price, and assembly loads.

For the dock term adjustment stage, we have derived proper values of tabu search parameters through various experiments. **Figure 8** shows how the load balance and launch duplication improves as the search proceeds. We can see that the evaluation significantly worsens at times due to the search diversification but it quickly discovers new best solutions.

The master plan scheduling system has been tested using real world data and evaluated by the experts in the yard. The experiments have revealed that the output of our system is satisfactory and the processing time is also reasonable. **Figure 9** is an exemplary snapshot showing a Gantt chart of the master plan schedule.



(a) Arrangement of a few docks on a certain day.



(b) Spatial layout status of ships in a dock during a time interval.

Figure 7. A screen snapshot of dock arrangement simulation.

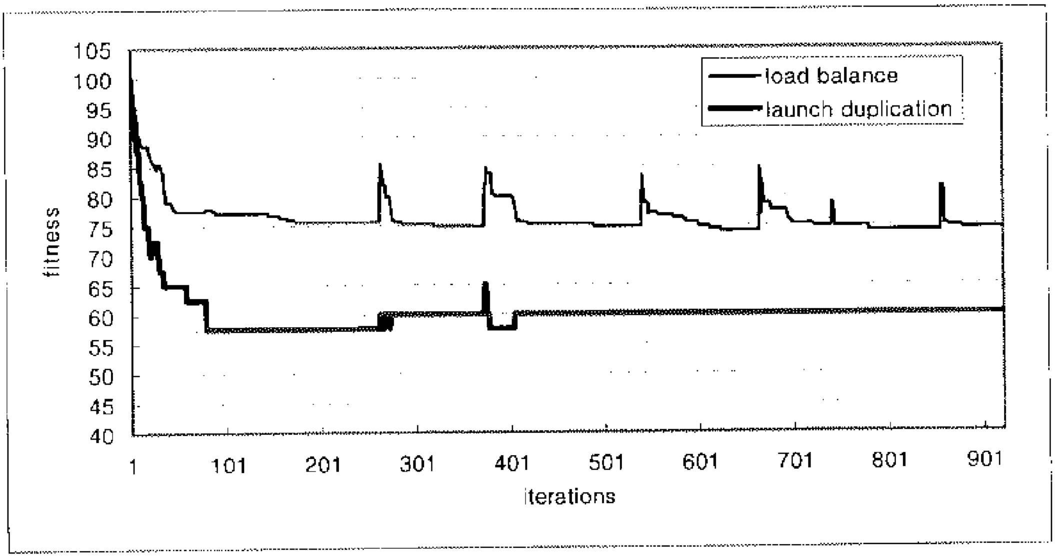


Figure 8. Improvement of load balance and launch duplication as the tabu search iterates.

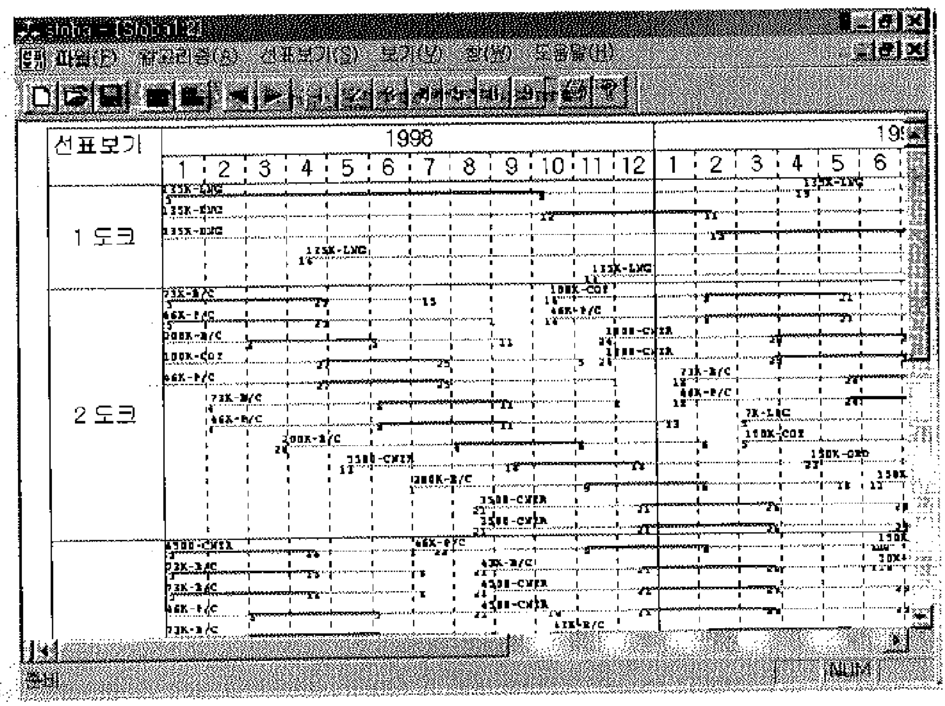


Figure 9. A screen snapshot of a master plan schedule.

5. Conclusion

The master plan scheduling is a problem of high computational complexity. To cope with the complexity, we have systematically decomposed the problem into hierarchically organized subproblems solving each of them heuristically in separate stages for computational efficiency. The master plan scheduling problem is composed of three subproblems: the product-mix generation, the dock arrangement simulation, and the dock term adjustment, each of which are solved heuristically by a genetic algorithm, a convex polygon algorithm, and a tabu search, respectively. We have developed a master plan scheduling system and it has been tested using the real world data. The experiments reflecting various management strategies have revealed that the output of our system is satisfactory and the processing time is reasonable. By using this system we can significantly reduce the planning efforts and planning lead time. This makes it possible to easily derive several different schedules with different emphasis on various objectives, and to use the comparative results to make an optimal managerial decision.

References

1. Chang, H. M., Choi, W. J., Park, H. K.(1993), "A Study On The Application Of Mathematical Methodology And Computerization To The Dock Arrangement", *Hyundai Engineering Review*, Vol. 13, No. 2, pp. 18-23.
2. Glover F.(1993), "Tabu Search : Part I", *ORSA Uournal on Computing*, pp.190-206.
3. Glover F. (1994), "Tabu Search Fundamentals and Uses", *Graduate School of Business, University of Colorado, condensed version published in Mathmetical Programming : State of the Art*, pp. 64-92.
4. Goldberg, D.E.(1989), "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley Publishing Company.
5. Lee, K. J., and Lee, J. K.(1995), "The Dominance and Preference of Search Space in Dynamic Spatial Layout", *Working Paper*, KAIST-MIS-WP-9503-01, Department of Management Information Systems, KAIST.
6. Lee, K. R., Moon, D. W.(1992), "A Study On The Optimization Of Main Schedule", *Hyundai Engineering Review*, Vol. 12, No. 4, pp. 73-85.
7. Lee, T. E., Song, J. S., Im, J. C., Park, J. C., Jeong, D. S., Lee, K. R.(1996), "Search-Based Heuristic Algorithms for Basic Planning in a Large Shipyard", *Journal of Ship Production*, Vol. 12, No. 4, pp. 211-219.
8. Lozano-Perez, T.(1983), "Spatial Planning: A Configuration Space Approach", *IEEE Transactions on Computers 32(2)*, pp.108-120.
9. Morton, T. E. and Pentico, D. W. (1993), "Heuristic Scheduling Systems", John Wiley & Sons,

Inc.
10. Zhu, D., and Latombe(1991), J.C., "Mechanization of Spatial Reasoning for Automatic Pipe Layout Design", *AI EDAM*, 5(1), pp.1-20.

THE ROLE OF ADVANCED COMMUNICATIONS IN SHIP ENGINEERING

J. Torroja, Sener, Ingeniería y Sistemas, S.A., Tres Cantos, Madrid, Spain
P. Gómez, Sener, Ingeniería y Sistemas, S.A., Tres Cantos, Madrid, Spain
M. Moreno, PYMAR, Madrid, Spain
A. Rodríguez, Sener, Ingeniería y Sistemas, S.A., Tres Cantos, Madrid, Spain

1. The Cycles of Ship Engineering

As is well known ship design is performed in cycles. In the case of merchant, fishing or service vessels the following cycles can be identified:

- **Contract Design** is intended to support the Building Contract technically. It is based on the Owner's requirements and on the Yard's Bid and defines preçisely the performance, quality and cost of the ship.
- **Class Design** is intended to obtain the approval of the Regulatory Bodies and the Owner. It is based on the Contract Design and on basic documents from suppliers. Its output is a set of approved documents and of main material specifications.
- **Detail Design** is intended to procure, fabricate, erect and test all the elements and systems of the ship. It is based on the Class Design and on detail documents from suppliers. Its output is a set of additional specifications and of production documents.

From the point of view of their relative importance on the cost and performance of the ship, the rating of the above three cycles coincides with their development in time. Contract Design is the most critical, followed by Class Design and Detail Design.

However, from the point of view of the amount of information, the rating is reversed. Detail Design contains an enormous amount of information; Class Design contains much less and Contract Design contains only a small amount of information.

As communications problems increase with the amount of information to be handled, this paper deals mostly with Detail Design and, to a much lower extent, with Class Design. Contract Design is not important from the communications point of view.

2. Shipbuilding CAD/CAE/CAM. Integrated Systems

Since the time when computer applications to shipbuilding only addressed naval architectural calculations and structural piece cutting, there has been a tendency towards **integrated systems**, either specifically developed for shipbuilding or adapted from general purpose systems.

Integrated systems should fulfil the following requirements:

- To cover most ship engineering activities, including concept and contract design, as well as class and detail design for hull, outfitting and electricity.
- To have one single database connecting all application programs, with a management software controlling simultaneous access of users and information integrity and security.
- To have a common user interface for all applications.
- To be open, or able to be enhanced with new applications or improvements, while remaining compatible with databases created with former versions.
- To accept a large number of simultaneous users, possibly in different sites.

2.1 The 3D Ship Product Model

Integrated systems are based on a 3D Ship Product Model in which the geometry and the attributes of the elements of the ship are stored. The model, which is built as an essential part of the engineering work, can be visualised at all stages and can be exploited to obtain information for material procurement and for production. The main characteristics of a typical ship product model are discussed in the following paragraphs.

2.1.1 Hull Structure

The basis for all ship product models is the definition of the main hull surfaces: shell, decks and bulkheads. The shell is defined by means of line or surface formulations, while other surfaces are defined by means of simple mathematical formulations. There is a recent trend to use NURBS formulations both for the shell and for other surfaces.

The basis for definition of the hull structural model is the set of standards for materials and for construction details.

Plate elements contained in the main hull surfaces are defined by their material quality, thickness and contour geometry, while profile elements connected to main hull surfaces are defined by their scantlings and by their landings on said hull surfaces.

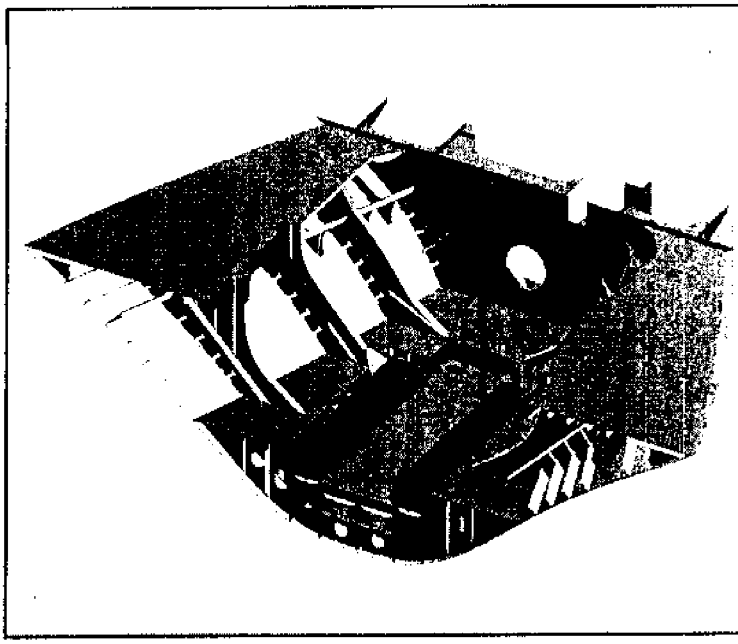


Fig. 1 - Hull Structure Model

Internal elements are defined by their internal contours and by the intersections of their own planes with the main hull surfaces. Intersections between internal elements take into account which elements are continuous and which are intercostal.

An important characteristic of integrated systems is the extent to which structural elements are defined topologically rather than geometrically. The main advantage of topological definitions, where geometrical data are not stored, but calculated on-line, is that changes in the main hull surfaces are automatically incorporated in the modified elements, just by reprocessing them.

An additional advantage of topological models is that the amount of information to be stored in the database is much less than is the case for geometrical models. This means lower requirements in storage capacity of hardware and, what is more important, lower bandwidth requirements in Local Area Networks and, above all, in Wide Area Networks.

2.1.2 *Machinery and Equipment*

The machinery and equipment concepts refer mostly to ship systems distributed through piping or ducting. Its Class Design work is mostly conceptual. Graphical documents include symbolic representations, without geometrical content, and may also include logic content, such as compatibility checks.

In systems extending over a large part of the ship, items of equipment may be placed, in piping and instrumentation diagrams (P&ID's), at their approximate position on plan, profile or transverse views.

In integrated computer systems, machinery and equipment Detail Design work is basically done in 3D. Previous to the creation of the 3D model is the introduction of piping, ducting and fitting standards, including technological and geometrical attributes.

Introduction of machinery and equipment data, which starts in the Class Design phase with the most important technological attributes, continues in the Detail Design phase with the geometrical data.

Once equipment items are modelled, including their anchoring points and pipe connections, each equipment model is placed in position with reference to the hull structural model or to the main hull surfaces. Ducting and piping are then modelled, usually starting with the largest elements.

2.1.3 *Electrical Systems*

Electrical and instrumentation engineering basically involves defining electrical equipment items, creating one-line diagrams, creating and managing cable catalogues, modelling cableways, routing cables and managing cable terminations.

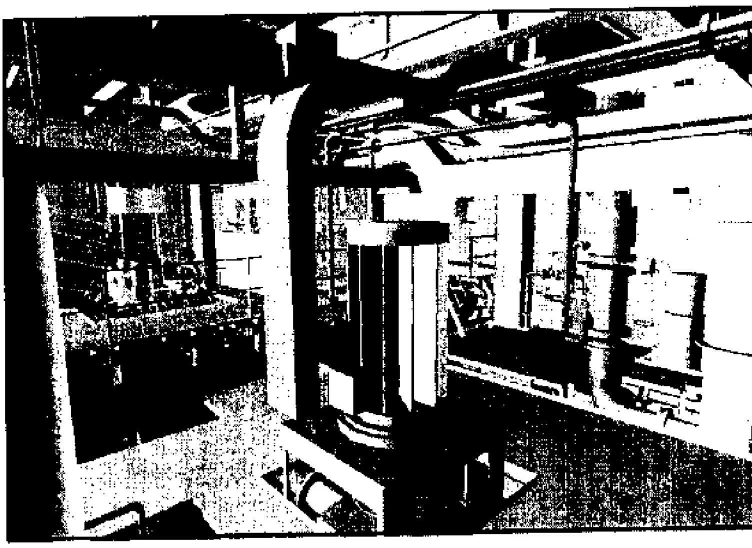


Fig. 2 - Hull and Outfitting Model

Creation of one-line diagrams and selection and specification of cables are usually done as part of the Class Design, while the remaining activities are part of the Detail Design.

Cable selection includes definition of sets of conductors and of insulation and protection layers, and is based on permissible voltage drop and on allowable temperature under standard and short-circuit conditions.

Although traditionally piping and ducting design has been loosely co-ordinated with cableway design, the use of a 3D model offers an excellent opportunity to improve the coordination, as duct, cableway and piping design can be done simultaneously.

Cable routing is usually based on the geometrical configuration of the cableways and on the permissible use of transverse area and maximum number of layers at each cableway segment. Due to the large number of alternatives, automatic and semi-automatic algorithms are used to select the optimum solution.

2.1.4 Accommodation

Accommodation design involves basically three types of activities:

- General design activities, to define the appearance of each space and the distribution of its elements.
- Material management activities, to specify, order and manage the specific accommodation materials.
- Piping, ducting and electrical design activities.

General design activities are largely intended to give the Owner a clear idea of the aspect of each space. The best way to do this is to create a 3D model of each space, during the Class Design work. This does not call for special modelling requirements but, contrary to machinery space 3D models, material textures are needed in accommodation 3D models.

Material management activities can be done in 3D or in 2D, as in most accommodation spaces the vertical co-ordinate is nearly always constant for each tweendeck. These activities are based on sets of material standards, and on material accounting.

As regards piping, ducting and electrical designs, which are basically performed in spaces between the hull structure and the accommodation ceilings and non-structural bulkheads, the tools and procedures are the same as those used in machinery spaces.

2.2 Construction of the 3D Model

Construction of the 3D ship product model can start as soon as the Contract Design is complete, the Class Design is quite advanced, the final hull lines are almost faired, the breakdown of the hull into sections is defined and a section erection schedule is available.

Most yards incorporate as many outfitting elements as possible into the prefabricated structural sections, before they are welded together in the building berth or dock. In this way on board outfitting work is reduced, thus reducing labour and delivery time. This means that the time lag from hull structure modelling to outfitting modelling in each section or area must be very short, or even zero in areas such as double bottoms.

A great advantage of 3D ship product models as compared to conventional methods is the fact that all the people involved in a design have real time access to the complete model (structure and outfitting) in each area. In this way, provided a set of rules of precedence is established, interference control is fairly simple. This is even more so with systems with on-line clash control capabilities.

A number of shipyards now obtain some advantages of series production through the use of similar processes for different ships. To achieve this the ship and its production processes must be developed simultaneously by using group technology, which basically consists in splitting the ship into geometrical "zones" and time "stages".

Each combination of zone and stage defines an "Interim Product" with its own production process and its own process line. Interim products are grouped into families characterised by size, material and individual production operations. Each family of interim products is produced in a work station or process line for all ship types.

When designing each element, during construction of the 3D Ship Model, its production process must also be defined. Thus, if the ship is to be built by zones and stages, its detail design must also be developed by zones and stages.

2.3 Exploitation of the 3D Model

The ship model is exploited by extracting from it the information and documents necessary for material procurement and for actual production. Although some documents are extracted from the model in the class design phase, most of them are extracted in the detail design phase. This documentation includes:

- Information for material management.
- Production drawings for interim products.
- Material lists for interim products.
- Information for fabrication of structural parts.
- Information for fabrication of pipes, ducts and cableways.
- Information for pipe, duct and cableway penetrations through structural elements.
- Information for fabrication of interim products.
- Information for estimating man-hours.
- Information for painting.
- Information for testing.

Due to the enormous amount of information to be produced, it is most important to be able to generate this automatically, with a minimum need for editing. It is also essential to keep the amount of information in each document to the absolute minimum, while maintaining the formats sufficiently flexible for all the possible alternatives.

2.4 Auxiliary Tools in Ship Engineering

In recent years there has been a shift of interest among shipbuilders toward product development, trying to improve their capabilities as they have been doing in the past with "day-to-day" operations.

Integrated Product Development (IPD) is the result of applying Concurrent Engineering techniques, and reflects the participation of shipyard, and other downstream industries in ship development.

The infrastructure technology for IPD is becoming simpler and less costly. Standardisation in platforms, networking software and database software by system integrators is an important part of this trend.

In general, IPD requires integration of information technology from three traditional IT areas: Technical, "Office" and Management applications.

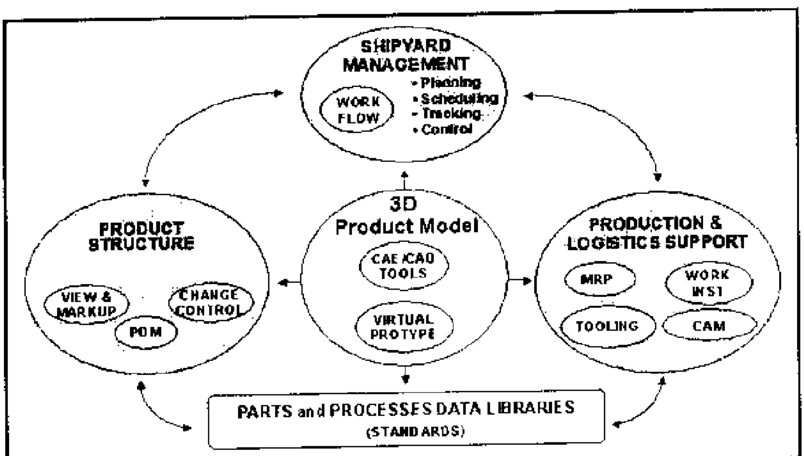


Fig. 3 - Integrated Product Development

Typical technical applications have been referred to before, and a large part of them are included in shipbuilding integrated systems.

Office applications cross "vertical" disciplines and are usually selected at the department, division or enterprise level. Office applications include word processing, business graphics and presentations, document creation, spreadsheets and e-mail.

Management applications used at the department to company levels include workflow, project management, product data and document management.

To support requirements for communication and data sharing, technical, office and management applications should be linked together within their respective domains and across the three domains where appropriate. Applications may be linked by interfacing or integration. Both forms of connection are useful and both have their strengths and limitations.

Interfacing provides a direct connection between two applications at a particular point in time, translating data between applications which have different local data models and formats.

A shared, single database between some of the applications containing the "official" data is the only route to more complete integration.

Product development is a creative, iterative process requiring human intervention and decision-making at many points. This is especially true of product development cycles for highly complex synthesis products, such as ships.

3. Characteristics of Integrated Systems

3.1 Databases

While the first shipbuilding CAD/CAE/CAM systems used files to store information, the increasing complexity of the systems, covering more areas of the design, called for new tools to handle great amounts of information in a fast, flexible way.

At the same time the increase in the number of users working simultaneously on the same ship called for concurrent access to the ship's information. In response to these requirements databases started to be used to store the information, partly or totally.

In the first instances relational databases were used to store the attributes of ship components, while in some of the second instances proprietary databases were developed to store geometrical and topological information of the ship model, as well as attributes of its components.

During recent years there has been a trend to replace the proprietary databases of integrated CAD/CAE/CAM systems with commercial databases with standard access languages and database server capabilities. The main reason for this change has been the ease with which new applications can be developed by the users.

Another reason for this change is the possibility to reduce the amount of information transferred between each user and the database. This is not very important in a LAN, but it is so in a WAN, where users work concurrently on a project from different sites.

The use of object oriented databases, proposed in the last few years, has not had wide acceptance, as the standardisation, concurrency and access capabilities of object oriented databases are considerably behind those of current commercial relational databases.

3.2 Topology

According to the usual terminology, the definition of an element in a 3D ship model is said to be geometrical when its co-ordinates are not referred to the ship lines or to other elements. On the other hand an element is said to be defined topologically when its co-ordinates are referred to ship lines or to adjacent elements.

Although, in the last instance, most data used for fabrication and erection are geometrical, integrated CAD/CAE/CAM systems tend to make extensive use of topological definitions, particularly in the hull structure.

One main advantage of topological definition is the ease with which design changes can be handled. For instance, changes in the hull form are straightforward. By replacing the original data with the new data the geometry of all elements concerned can be automatically recalculated. This permits to progress in the definition of the 3D structural model before the hull lines are completely faired.

Another important advantage, which has become apparent only recently, is the drastic reduction in size of the database. This involves a strong reduction in bandwidth requirements for WAN's, facilitating concurrent engineering work from various locations.

3.3 Local Area Networks

The development and use of integrated systems able to create ship product models, has been closely connected with the development of LAN's and, more recently, WAN's with sufficient bandwidth to cope with the flow of information.

This flow of information in a LAN depends on a number of factors including:

- The **complexity of the 3D model** and the **number of simultaneous users**, which vary throughout the development of each project. While at the beginning the number of users is small and the model has few elements, near the end the number of users is still near the maximum and the model is almost complete.
- The **volume and frequency of transactions** between the users and the 3D model, which depends on the structure of the database (topological models need smaller databases) and on the frequency of 3D model updates. This, again, depends on software design, on workstation power and on user working practices.
- The **location of the application software** when most of the process is done at the workstations. If the software resides in the server the network traffic is increased.
- The **switching capabilities** of the network. The traffic decreases as the switching capabilities increase.

From the above considerations it follows that the bandwidth necessary in a LAN and, even more, in a WAN, is an important consideration when selecting the hardware configuration. If engineering work is to be done concurrently from two or more locations, the application must be configured in such a way that information traffic is compatible with the bandwidth obtainable at an affordable cost.

4. Evolution of the Ship Engineering Environment

Important changes have occurred during the past decade, both in the industrial environment, in which ship engineering is performed, and in the field of information technologies, in which the engineering tools originate.

4.1 Industrial Environment

The following main changes have occurred in the environment in which ships are designed and built:

- **Delivery times have shortened**, thus reducing the time available to perform the engineering work. To make up for this time reduction, overlaps between Class Design and Contract Design and between Detail Design and Class Design are necessary.

- **The amount of required engineering work has increased**, as a consequence of the general acceptance of the engineering for production concept, which takes into account production requirements to a much larger extent than heretofore.
- **Yard payrolls have been strongly reduced**, which means that much engineering work needs to be subcontracted, particularly in Detail Design. This involves the need to co-ordinate several groups of people working at different sites on the same project.

4.2 Hardware and Operating Systems

During the last three decades the average cost / performance ratio of hardware has been halved approximately every one and half years.

During the late eighties a trend started in engineering applications to abandon proprietary operating systems in favour of UNIX. This trend was accentuated in the early nineties and was maintained well into the mid-nineties.

In the last years the combination of high capacity INTEL Pentium processors and Windows NT (or, more recently, Linux) operating system has allowed personal computers, until recently restricted to limited engineering applications, to enter the area of integrated shipbuilding systems. This alternative is having a wide acceptance and will undoubtedly have a deep influence on the evolution of ship engineering work.

The success of the INTEL alternative has not only been due to its very favourable cost / performance ratio (in the region of 50% of other UNIX alternatives). Other important factors have been the familiarity of many people with the PC environment, and the possibility of using the same hardware for engineering and for general purpose applications: word processors, spreadsheets and graphical editors.

4.3 Telecommunications

During recent years communications have improved at a very fast pace, mainly in the following aspects:

- The spread of Internet has put at the disposal of everyone a vast amount of technical information and the possibility of transmitting it in a reasonable time and at a very low cost.
- The introduction of digital communications technologies has made possible new techniques such as remote software maintenance and remote supervision of engineering work through public communication channels (such as ISDN or Frame Relay) in a Wide Area Network (WAN) environment.

On the other hand the deregulation of communications has introduced, at least in Western Europe, a higher level of competition, with the consequent reduction in costs. For this reason we believe it will be possible, in the near future, to perform concurrent engineering work from various remote sites, as we will discuss later.

5. Ship Engineering in Spain

A common circumstance of all Spanish shipyards is the inability of their engineering departments to cope with their requirements, particularly in the detail engineering phase. Many times the yards have to use external engineering resources and very often these resources are situated far away from the shipyard.

To cope with this, the shipyard has to break down the work in zones to be distributed between the shipyard engineering department and one or more external contractors. The shipyards must coordinate very carefully the geometric interfaces between these zones, which must be as small as possible and very clearly defined.

Another important characteristic of the Spanish Shipbuilding Sector is that most of the Shipyards (private and public) are using the same integrated system (FORAN) for ship design and production. This fact can be used to facilitate the use of external engineering companies, provided they use the same tool as the shipyards.

5.1 Peculiarities of Large Shipyards

All large commercial shipyards in Spain are operated by Astilleros Españoles, S.A., a State-owned concern. In general, contract and class designs are done in-house, with some yards doing work for others and, in some cases, using contracted personnel working at the yard's premises.

On the contrary, a considerable part of detail designs is contracted outside, mostly to local contractors, but also to contractors remote from the yards. If possible, outside work is done with FORAN. As engineering for production is widely used by large yards, the amount and complexity of detail engineering are considerable.

The breakdown of detail engineering work between the yard and its contractor or contractors is done on a geometrical basis, in such a way that physical interfaces are reduced to a minimum. Work coordination is done by the yard. Each contractor builds and exploits its partial 3D model, using standards defined by the yard. Upon completion of the work the partial 3D models are transferred to the yard.

The above situation involves complex problems of coordination, which would be considerably reduced if tools were available to perform remote concurrent work from various locations on a single 3D ship model at a reasonable communications cost.

5.2 Peculiarities of Medium and Small Shipyards

The main peculiarity of medium and small yards is their **diversity** of company size, market and type and size of ships built. While the largest yards in this group operate in a similar manner to large yards, the smallest have resource limitations which make them almost totally dependent from outside engineering contractors.

As a consequence there is a wide range of situations as refers to the yards' ability to perform contract, class and detail engineering. Almost two thirds of the yards perform class design, at least occasionally. Most yards perform detail design, at least partly, but are not self-sufficient to cover their engineering needs.

Some yards have their engineering resources scattered geographically. Some yards perform engineering work for each other, and there are some recent initiatives to share engineering resources among several yards.

Most yards are evolving towards increasing requirements of drawings and information for production, as well as to more exhaustive exploitation of the 3D ship model for management purposes.

In summary, medium and small yards need more and more to supplement in-house engineering resources with outside personnel working on a single 3D ship model containing more and more information. For this purpose the availability of tools allowing remote concurrent work from various locations on a single model, using relatively cheap hardware and with affordable communications cost, is extremely convenient.

6. Telecommunications in Spain. The PISTA Initiative

The following paragraphs describe the environment of telecommunications in Spain, as well as a recent Government initiative which is expected to have an influence on the future use of telecommunications in a number of sectors, including Shipbuilding.

6.1 The Spanish Telecommunications Environment

As in other European countries, telecommunications have been, until very recently, a monopoly of the State. Currently, data transmission is mainly done via Frame Relay or ISDN, depending on the bandwidth required. The number of Internet users in growing steadily; currently it is slightly more than 6% of the population.

Two years ago ISDN and Frame Relay solutions had a very high cost per Mbs of leased line. The progressive cost reductions in recent years have put the required bandwidths within the reach, not only of large yards, but also of medium and small yards, at moderate prices.

6.2 The PISTA Initiative.- PISTA NAVAL

The Spanish Administration, through the General Secretariat of Communications of the "Ministerio de Fomento", has promoted several pilot projects to use new advanced communication services. In the industrial area three sectors have been selected: Aerospace, Construction and Shipbuilding. This initiative is called PISTA.

The PISTA NAVAL project is oriented to take advantage of the new communication technologies in the Spanish Shipbuilding Sector. As already said, this Sector is characterised by the fact that both the shipyards and the engineering companies working for them are using FORAN as their main design tool.

As FORAN is based on the definition and exploitation of a 3D ship model, the coherence and compatibility of the work performed by different users is obtained by using a single database to store one single 3D ship model for all the users. This fact guarantees that the quality of the engineering work is far superior to that which can be reached when a 3D ship model is not used.

This advantage can be obtained when all of the users are working in the same local area network (LAN) to access the common database containing the ship model. When the workstations are geographically distributed, it would be necessary to access the model through a wide area network (WAN) using public resources for data transmission and the necessary bandwidth at a reasonable cost.

Until recently, several factors have contributed in not allowing the extension of the use of integrated systems such as FORAN to this environment in a proper way. One is the high communication cost associated with the required bandwidth, which is very high in the case of most integrated systems. Another is the fact that the database management software used to manage product models has not been designed to reduce data transmission ratios.

In the case of FORAN the bandwidth requirements are considerably lower than in the case of other alternatives, and this for two reasons:

- Because of its largely topologic character, the volume of the database is lower by an order of magnitude. Consequently, the amount of information to be transferred and the required bandwidth are also much lower.
- Due to the reasons mentioned above, the proprietary database of FORAN is being replaced with a relational database with SQL and database server. According to comparative tests already performed, this reduces the required bandwidth also by an additional order of magnitude.

Because of the above reasons FORAN is believed to lend itself particularly well to be used concurrently from remote sites.

The main objective of PISTA NAVAL is thus to take advantage of the PISTA initiative to extend the FORAN capabilities in order to be used effectively in these environments. In this way, a large number of users could be working concurrently from remote locations in the same 3D ship model. The coordination performed by the shipyard, which is the owner of the ship model, would be reached in a very easy and efficient way.

In order to reach this objective, FORAN is being revised within a re-engineering process specially oriented to those areas related with the communication between the modules and the database. An optimal data flow must be reached to reduce bandwidth requirements.

To complete the set of tools necessary to facilitate collaborative work, the integration of FORAN with other programs like NetMeeting, chat, shared whiteboard, voice transmission and others are also considered. In this way, distributed work can be organised in a very similar way to that in a local environment.

Two database architectures are being considered within this project. The first one is strongly preferred because it will work in a WAN exactly in the same way as current FORAN versions are working in a LAN. This architecture requires a single database to which remote users will be connected via Frame Relay or ISDN, depending on the number of users.

If the number of remote users is large enough, the second architecture can be used as an alternative to increased bandwidth. This architecture is based on a distributed database, with replication between the shipyard (master location) and the remote locations. In this case, the modifications performed by the users are updated in real time in the local database and with a certain delay (or even without it) in the other database(s).

The advantage of this architecture is that, most of the time, the FORAN modules are reading from the database (local read) and only a few times are these modules writing to the database (local and remote write). So the network traffic is reduced drastically.

The disadvantages are the complexity (specially in the management) and the software cost (because replication requires additions to the basic database software).

The Spanish Administration is very confident that the PISTA Initiative will facilitate the use of advanced communications in the above mentioned industrial sectors. For this reason, it has insisted that potential users be involved in the development of the corresponding projects. In PISTA NAVAL this role is being performed by AESA, representing the public sector, by PYMAR, representing the medium and small shipyards, and by CNP Freire, representing the private sector.

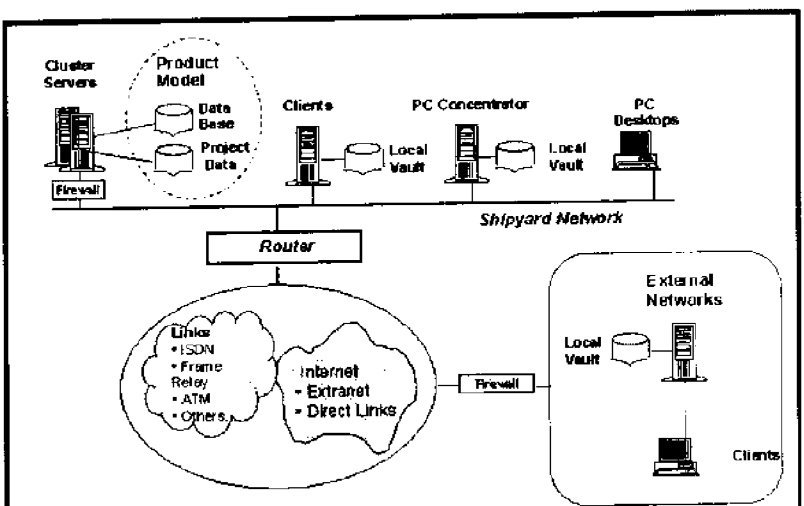


Fig. 4 - Wide Area Network Architecture

The objective of the shipyards, both large and medium or small, is to have the tools necessary to provide CAD/CAE/CAM applications, particularly FORAN, with the capabilities to enable users to work locally and remotely on a 3D ship models in the same manner in which they can work locally.

This would permit them, without significant investments in hardware, software or personnel training, to share engineering resources among yards, without physically displacing personnel, duplicating information, increasing data processing resources or "breaking down" the 3D ship model.

It would also permit

concurrent work on the same 3D ship model from various engineering groups of the same company or from outside engineering contractors at different locations.

6.3 Solutions for Large and for Medium / Small Shipyards

Although only actual experience with remote operation of FORAN will tell which of the two above mentioned architectures is better, both the large yards and the medium and small yards in Spain have shown their tentative choice for the first period of operation.

Large yards have indicated a preference for installing a master database at the shipyard where each ship is to be built, with replicated database(s) at other yard(s) or engineering contractor(s) working on the project. Remote access will be done via Frame Relay.

Medium and small shipyards are in favour a single database, to which remote users will access via ISDN.

7. The Future of Ship Engineering

The possibility to do remote concurrent engineering work on 3D ship models will lead shipyards and engineering contractors to a different manner of co-operation in the future.

As there will be one single 3D ship model, coordination of the various engineering groups, either at the yard(s) or at engineering contractor(s) will be enormously simplified, and the risk of incompatibilities will be almost zero.

When circumstances make it advisable, permanent communications, via Frame Relay or ISDN, will be established between shipyards and engineering contractors.

An interesting possibility will probably arise for small engineering organisations not near to the shipyards, and even for individuals working at home, which, up to now, have not been a real alternative to perform detail engineering work for shipyards, as their small contribution does not justify the coordination work necessary on the part of the shipyard. This type of organisation will be able to work remotely from PC's, via ISDN and using the yard's software, on 3D ship models in the yard's database.

Due to the tendency of shipyards to purchase complete packages rather than individual items of equipment, many equipment suppliers will act as turnkey contractors, with their own engineering capacity. This will lead them to use the same design tools as the yards, and to work remotely on the shipyard's 3D ship model.

Regulatory Bodies, which up to now are making relatively little use of telecommunications in their work for shipyards, will probably make a more intensive use. In this case bandwidth will not probably be an issue.

References

1. Gómez, P., Telematics: A Challenge to Concurrent Engineering, ICCAS 97
2. Conway, J., Integrated Product Development: The Technology, Winners Consulting Group

3. Shina, Sammy G., Successful Implementation of Concurrent Engineering Products and Processes, Van Nostrand Reinhold.
4. C. Gillmer, T., Modern Ship Design, Naval Institute Press, Annapolis.
5. Keane Jr., R.G., and Capt Tibbitts, B., Re-engineering the Warship Design Process: Towards Total Quality Development, INEC-96.
6. Keane Jr., R.G., and Capt Tibbitts, B., The Warship Design Process: Making Design Everybody's Job, Naval Engineers Journal 1994.
7. Jons, O.P., Ryan, J.C., and Jones G., Using Virtual Environments in the Design of Ships, Naval Engineers Journal 1994.
8. Huthwaite, B., Strategic Design: A Guide to Managing Concurrent Engineering, The Institute for Competitive Design, Rochester, MI, 1994.

THE APPLICATION OF TECHNOLOGY TO IMPROVE DESIGN, PRODUCTION AND THROUGH LIFE PROCESSES FOR WARSHIPS

Colin Beames, Director of I.T., Vosper Thornycroft, Southampton, Hampshire, UK

Introduction

Vosper Thornycroft is a designer, integrator and builder of high specification, high quality naval vessels. These products range from 30metre patrol craft to frigates. This has been recently augmented by the acquisition of Halmatic who manufacture a wide range of vessels up to approximately 30metres. Within this complete range all types of materials are used including composites, aluminium and steel.

Throughout its history Vosper Thornycroft has been known as an innovator of new designs which have encompassed the available technology. Today is no exception with the arrival of our recently enhanced stealth design, Sea Wraith II and the Trimaran Demonstrator, RV Triton, being built for the Defence Evaluation and Research Agency (DERA).

RV Triton is being designed and built as part of the concept and assessment phases of the Future Surface Combatant project for the Royal Navy. The Royal Navy's interest in this concept is based around the benefits that the design provides. These benefits are slender hulls which give reduced drag figures, improved seakeeping and the ability to locate the flight deck closer to amidships, thereby increasing the envelope for helicopter operations, greater available deck area, enhanced survivability and signature reduction.

Sea Wraith II is a 135metre frigate design that incorporates a flight deck, hangar and armament. The aim has been to incorporate the design changes necessary to enhance its naval capability but at the same time maintain the low signature of the ship. The essential elements which make Sea Wraith II such a revolutionary addition to warship technology are masking of infrared signature, cooled emissions from exhaust systems, masts positioned to counter radar homing missiles, retractable mast and communications antenna enclosed in the vessel's stealth profile.

Innovation and the ability to design, deliver and support complex naval vessels has allowed Vosper Thornycroft to supply products and designs to over thirty countries in the last thirty years.

General

The use of technology is critical to any business. Technology is a term which is widely used to address the marvels that are available today in the computer world. In Vosper Thornycroft when we talk of technology we refer to any facility or resource which is being used in a new way. Therefore constructing structures in a different way, pulling cables more efficiently and the utilisation of our computer facilities are all ways of exploiting technology.

There are pitfalls that await any business. The main three of these are :-

(a) not all technologies provide the efficiency and quality that is needed today

(b) high tech computer systems can become a burden

(c) new methods not accepted by the employees which mean that they will never be successful

If these potential problem areas are not addressed then the introduction of and success of new technologies will not bring the maximum benefits. The following sections of the paper considers these three pitfalls in different areas of Vosper Thornycroft's design and production processes. A measure of success of any new method, whether it be a simple new production process or a new company wide computer system, is "does it make the jobs of those employees using it easier". If it does then it will be successful, if it does not then it is likely to encounter problems.

Team Working

Like all companies Vosper Thornycroft is continually addressing its production methods and facilities in order to meet the challenge of ever shortening delivery times and greater demands for quality. Our approach to meeting these challenges is to review continuously and evaluate our processes to identify areas where improvements can be made. The key here is the involvement of employees at all levels and from all areas in the process. Gone are the days of departmental boundaries and demarcation. The rule today is that all in the process need to be involved from day one and everyone is important no matter what role they perform. If their role is vital to the process then they are of equal value.

To address this we are now a 'team' orientated organisation which works together to meet the demands of the task to be done. Wherever possible these teams are co-located with the team members changing to suit the need of the task at any one time. An example of this is the setting up of a project office for the Trimaran project. The office currently contains all of those involved in the design and planning phases of the project. There are, under the direction of an Executive Project Manager, designers, planners, draughtsmen, procurement, production staff and through life support staff housed in the same office sharing the same information and working together to achieve the same targets. The tools they use are of course their own knowledge, communication with others and IT systems geared to providing the information needed at the right time. The scale of the project is such that the office is manned by approximately 50 staff during the design phase and located in the Design and Technical building. As the project develops the team will move to the production area so that communication with the workforce, at that stage, will be easier.

The concept of the project office is not new but this is the first time that all involved have been co located. Our experiences to date have been reviewed and improvements identified for the next project. Part of this review is considering how we can make people's jobs easier and the environment right for the task to be done.

The main benefits gained today, from the people view, are that all the relevant information and people needed are immediately available and that all have common goals which reduce the individual's frustration in wanting to get the job done as quickly as possible. The one problem that has been found is that it takes time for many different people from disparate parts of the organisation to become a team. To understand this problem we have undertaken a review of all of the issues found and are addressing the problem areas. Specific team building training is being undertaken to reduce the time taken to blend the different staff into a single team. A benefit already seen is improved communications between members.

Design Processes

The project team approach has affected the design areas to a greater extent than some others. The problem of producing the design information quickly remains but the demands to include production friendly features and to consider the requirements of minimising through life costs are growing. Whilst designers have always considered these two requirements, in today's environment they are much more critical to the overall success of the design.

There are now several areas where Vosper Thornycroft is evolving its capabilities to enable the best possible overall design to be created, these include:-

Systems Engineering
This includes requirements engineering, system analysis, system design, performance modelling, etc. These terms are fashionably associated with software projects but have a direct parallel in ship design. Some of the methods and tools transpose directly from the software world, others need to be modified to suit the ship environment.

Cost and Operational Effectiveness Assessment (COEA)
The shift of emphasis from performance parameter specification to high level operational effectiveness requirements is encouraging the development of innovative solutions. This can include the development of partnerships where the shipbuilder becomes responsible for more than the supply of the vessel, taking on support and even operational roles.

Life Cycle Costing (LCC)
Total cost of ownership appreciation on the part of owners and operators is providing more focus on LCC during the design phase. To achieve this studies such as Availability, Reliability and Maintainability (ARM) need to be performed.

The effect of these growing requirements and greater emphasis on team working is changing the role and environment of design staff. Today they need greater amounts of information in an ever shortening time to meet the agreed deadlines. To achieve this the availability of correct information and access to the appropriate tools is vital.

Through Life Support

The significant change within this area is the involvement in the project team from day one to ensure that the overall requirements of through life support are catered for. This early involvement means that information required can be generated at the same time as that for design and production, resulting in less work and improved consistency of data.. In the past this has caused significant frustration which the new approach is minimising. In addition the staff within the through life support area have operational knowledge that is aiding the designers in their tasks.

Production Processes

In the Production areas, again multi discipline teams have been formed to tackle the construction of the vessel. New methods have been employed which make for greater efficiency but at the same time make the individual's job easier. Some examples of these are:-

- The task of pulling cables into the ship is one that requires considerable planning and manual effort. We now set up a jig in the open sky environment which allows the workforce to lay the cable in the jig and dress the run prior to the deckhead being installed. This saves considerable time, dramatically reduces the manual effort required and with appropriate staging allows for the work to be undertaken in the down hand position. This new way of working came from the workforce.
- On our FRP Minehunters for the RN the outer hull is shotblasted instead of being disked prior to painting. Again this has brought significant efficiencies by reducing the man-hours required by 50% and has made the environment better for those doing the task.
- In 1992 we invested in a laser cutter for steel preparation, at the time the best method available for cutting our structure. This was connected to our CADCAM system where the parts were defined. Part definition included all the marking and numbering required operator to identify the pieces when cut and all information needed for assembly. Three dimensional keyplans were introduced to provide the necessary overall view of the unit to be constructed. This approach produced significant saving throughout the process but it also reduced the frustration of the individual doing the task as all components were clearly marked. Again all that was needed was provided.
- In our process of review we identified that Plasma technology was at a stage were it has become as good as laser as far as quality is concerned and its environment had improved dramatically from the days of sunken baths and generally messy work areas. We have installed a new plasma and have linked it with our laser and CADCAM to enhance our capabilities.

Within Vosper Thornycroft we have an Engineering Services company which undertakes project management work and production work in a variety of sectors including Offshore and Nuclear. The nature of this business is for smaller projects and shorter time scales than shipbuilding itself. This has allowed Vosper Thornycroft to learn from different market places and those lessons learnt are being introduced to shipbuilding where appropriate. Our staff are also able to gain experience on smaller projects, therefore allowing them to develop faster.

A significant part of our R&D programme is devoted to our production processes and this has led to various method and process changes being implemented. As with all areas we involve those doing the job to ensure that what is proposed will work. One such development is the use of a racking system that can be used at the outfit stage to facilitate the seating of all minor equipment within a compartment or space.

Vosper Thornycroft have been using three dimensional modelling for the detail design/ of the ship since the early 1980's. As this technology has developed greater use of pictorial presentation of information has allowed better understanding of the end product. Having the model as a database gives a greater variation of presentation, therefore, allowing all those who use the information to gain a better understanding. Traditionally, the Drawing Office produced the information that they needed to detail design the vessel. Today whilst this needs to take place, they are able to provide the right information to build the vessel as well. Their role in the actual manufacture of structure and pipes, to name two, has been transformed from producing a drawing which others interpreted to get the production information to providing the toolpaths themselves for the components to be made from. This provides the draughtsman with a more fulfilling role that gets him involved with production and the necessary high quality information which dramatically reduces the problems found in the manufacture and construction phases.

Continuous Improvement

One facility that is open to all employees is our Continuous Improvement Programme (CIP). This is a mechanism where anyone can suggest a way which will improve the way they or their colleagues work.. This scheme is aimed at the lower level activities that we all undertake but not restricted to them. Since its inception 3000 proposals have been made with over 50% being implemented. Such suggestions have been re-hanging a door to allow better access, to producing a jig for equipment assembly in the shop as opposed to on the vessel. Virtually all of these proposals are ones which simply make someone's job easier, therefore, making that individual more efficient.

Information Systems

Underlying the business processes are systems more and more of which are computer based. The role of Information Technology (or Data Processing (DP) as it used to be known) is to provide the information infrastructure that the business needs to operate efficiently. The world of the computer boffins within Vosper Thornycroft has changed as dramatically as any other part of the company. Gone are the days when individuals never stepped outside the hallowed walls of the DP department. Today IT staff are as much a part of the overall shipbuilding process as those who weld the structure. In simple terms if the IT system fails then there will not be any components to weld together, if not immediately then within a very short time.

For IT to provide the required level of support the individuals need to understand the business process as well as those who use it on a daily basis. Within the greater world of IT there is, somewhere, the technology to do whatever you want to do. The danger is that too much technology means it is great fun for the IT specialist but a major headache and, therefore, the cause of much frustration to the users of the tool. A simple rule of thumb is 'if it is simple then it usually works and does not cost the earth to use or support'.

Within Vosper Thornycroft we have been implementing a system strategy over the last three years which has been a partnership between those who need and use the system and those who provide it. We have moved away from in-house developed systems wherever possible and have purchased packages which are based on Oracle. The logic here being that we should not try to reinvent wheels and that by using packages we will stay up to date and learn from others using the package. Using a common database gives the potential of integration at the data level.

Whilst it is inevitable that some of the systems that we have and are implementing are in themselves complex the use of them needs to be simple and focussed around the business process. The challenge to IT is to ensure that all areas use the systems available and not to develop 'specials', which are just a variant to an existing system, for some areas.

Our system strategy is made up of the following components, these are:-

1. Project Management (PMS)
2. CADCAM
3. Engineering Data Management (EDM)

4. Material Management (MS)
5. Finance System (FS)
6. Office Systems
7. Supporting Systems

The role of these systems is to provide the tools and the information that the business needs. The selection of these has been a business and not an IT decision. With the exception of CADCAM and EDM, which have been ongoing upgrades of systems we have been using successfully over a number of years, the systems have been selected by a team drawn from the areas of the business who will use them once deployed. This same team has been held together to undertake the implementation of the system with those user area representatives becoming the 'expert user' in their project or department once the system has been deployed.

1. ***Project Management System (PMS)***

The software used here is Project X supplied from PSDI. Vosper Thornycroft use this system to manage all activities across shipbuilding. This includes both project and manufacture planning. The main role of the system is to provide a project management tool based on the earned value concept and the CSCS standard. In doing so it provides information both on the performance against the schedule and financial performance of a project in order that the project manager is aware of progress against the programme and budget. It also provides the business and customer with statistical progress information.

2. ***CADCAM***

Software used is predominantly CADDS supplied by Parametric Technology Corporation, formerly from Computervision. As stated earlier we have been using the system since the early 1980's and exploited its three dimensional modelling capabilities fully. The current software includes shipbuilding's specific structural package which was developed with Vosper Thornycroft's involvement. This system is used throughout the design and detail design phases and provides fly around video sequences at one end to detailed piece part and toolpath information at the other.

3. ***Engineering Data Management (EDM)***

Software used is Optegra from Parametric Technology Corporation, formerly from Computervision. The facility which is still being rolled out provides the repository for all engineering information including component details, document management, drawing management, weights, etc.

4. ***Material Management System (MMS)***

Software used is Control from Cincom. This system which has recently gone live manages all aspects of material acquisition and logistics from requirement identification and specification through to delivery to the assembly area. It is integrated into the Finance System so that all costs can be collated and passed to the Project Management System for reporting.

5. ***Finance System (FS)***

Software used is CODA from Baan. As stated above this system is integrated with the Materials and Project Management Systems so that all costs are collated and reported

6. ***Office Systems (OS)***

Microsoft Office and other Microsoft products are used.

7. ***Supporting Systems***
These include such systems as Personnel and Pay Role, Maintenance System, etc which whilst not directly used within the engineering process are vital to providing the information that is needed to support the process.

The systems above are integrated to some extent and work continues to add integration where needed. For the expert user the systems provided are as close as possible to the standard product but for the casual user we provide the limited amount of information that they may need either via an icon or accessed via the intranet.

It is always a difficult path to tread when considering customising systems to provide greater efficiency to the user and, therefore, making his life easier and the cost of providing and maintaining the change together with the implications of the change when the software needs upgrading. With the ever increasing level of PC expertise of the workforce, because of the use of PCs at home, the demand for all systems to look like and function like Microsoft Office continues to grow, but from an efficiency position and an ongoing maintenance position this is not possible. This, at times, can cause significant frustration to the users of the systems. The balance can be hard to find. A lesson that we have learnt is that whilst training in the use of the system is essential, education in the role of the system and how it fits into the overall business process is critical to its success.

We have experienced a few occasions when the introduction of a system or new techniques has been seen as a threat to the workforce. This has had the effect that the workforce has responded and made the current method so efficient that the new way was not adopted. This has proved adequate in the short term but has not been sustained, the lesson learnt here is that better education and selling of the new way could have brought greater benefits earlier.

Computer systems have the potential to remove a great deal of the frustration suffered by individuals when they are unable to find the information that they require. However, once reliant on the computer to supply the information any problems seen when the system fails can cause even greater frustration. The IT supporting organisation must understand and organise for this. If they do not then the failing system will become a burden and an excuse of all problems suffered anywhere in the process. Systems have to be seen as belonging to the users not IT.

The other issue with systems is that they must be appropriate to the business and not be the use of technology for technology's sake. The issue here is that in general terms there is technology available to do anything you can think of but if it is not appropriate to the business then it will either become a burden or will fall into lack of use. What does appropriate mean? Here we can use the analogy that we use when selecting components or equipments for our ships that is 'total cost'. In equipment terms this means that the initial cost needs to be considered together with installation and maintenance costs before a purchase is made. This can often result in not buying the cheapest item. The situation with IT systems is just the same with the added dimension that we have the ability of 'customising a system to death'. Therefore very careful use of and customising of system needs to take place to ensure that their overall cost is less then the benefits the system provides.

System customisation is people driven whether by demands from the users or by exploitation of the technology by IT. If this is wrong then the level of frustration and the reduction in efficiency of either the user area or IT staff can be detrimental to the efficiency and costs of the organisation.

Lessons Learnt

This is a continual process we must review and evaluate if we are to keep moving forward. The overall lessons learnt are:-

(i) Our people are our biggest asset. The more that we involve those who do the job then the more we will improve our processes. We must include all those involved in the process, the whole team and not the individuals. Those supporting organisations such as IT are part of the extended team.

(ii) We must use technology which is appropriate to the business and not fall into the trap of technology for its own sake. We must consider the total cost of the technology and not just the purchase cost.

The Future

The future holds ongoing development of designs that include innovation as well as production friendly features. The ongoing drive to reduce our design to delivery timescales and reduce the cost of our products, whilst maintaining and increasing our quality will continue. This needs to be achieved by investing in our people and involving them in improving our processes. We must review and evaluate what we have achieved to date and be prepared to invest in our future.

We need to spread the knowledge of individuals within the organisation so that we have a greater number of people with general skills who are able to take up positions within the company whilst maintaining an acceptable level of specialists. These general and specialist people are of equal value to the organisation but have a different role. We will continue to provide as healthy and pleasant an environment in the workplace, whether it be in the offices or on the berth, as is practical.

We will develop the education part of our training to ensure that all understand the process and what the next person in the chain needs to complete their task. Our aim is that the overall process benefits and not just for one area to be better than any other.

In system terms we will implement the systems that the business needs, making them as simple as possible to use whilst ensuring that the overall cost is less than the value of the benefits gained.

The overall message is to involve everyone in the improvement process, educate and train them and implement the appropriate technology that the business needs.

VIRTUAL MANUFACTURING FOR SHIPBUILDING IN A GLOBALLY COLLABORATIVE ENVIRONMENT

Thomas W. Scotton, Deneb Robotics, Inc., New Britain, Connecticut, USA

Introduction

The list of ship building related companies that are currently using some form of Virtual Manufacturing technologies includes: Avondale, Fincantieri, General Dynamics - Electric Boat, General Dynamics - Bath Iron Works, George Sharpe, John J. McMullen Associates, Newport News Shipbuilding, M. Rosenblatt & Son, Inc. Companies such as these have customers throughout the world that continue to demand increased quality, improved performance, shorter purchasing lead times and lower costs for their highly sophisticated and extremely complex products. Our customers are also interested in ways to make our products easier to use, even more reliable and maximize the time of operation before servicing. The key to successful business is to provide the customer what he/she wants, in a timely manner, any where in the world, at the lowest possible price. We believe that Virtual Manufacturing Technology can help your company become much better equipped to respond to your customers needs.

Virtual Manufacturing is being aggressively implemented across key industries including Aerospace, Automotive, DoD, DOE, Nuclear, R&D, and other industries to become more competitive, shorten launch cycles, and improve communications across global enterprises. Companies that have adopted Virtual Manufacturing technologies for the launch of new products and streamlining of operations include General Dynamics, General Motors Corp., McDonnell Douglas, Lockheed-Martin, Caterpillar, Northrop-Grumman, United Technologies and many other industry leaders. Time to market is being reduced up to 50 percent and lifecycle cost savings of 30 percent are being realized through implementation of Virtual Manufacturing technologies. Seventy to 80 percent of the tooling re-engineering and rework dollars have been saved by building the fixtures right the first time.

Virtual Manufacturing

In this paper, our definition for Virtual Manufacturing (VM) is: **An integrated set of tools** that provide us with the ability to model products, processes, and factories in an efficient and effective manner. VM tools serve as an effective interface with our customers and partners when determining and verifying the product usability, serviceability and functionality all in the virtual environment before any components are built. These tools will be used to help us make effective decisions (as we conceptualize and develop new products) by showing the impact of design parameters on the ability to produce the product in a timely and cost effective manner. The product design modeling information will be used as a basis for the development of the appropriate manufacturing processes and new equipment if necessary. The process and equipment models will then be used to determine the factory production capability requirements to quickly determine an accurate production cycle time and cost. VM tools will help in the decision making process when it comes to issues such as: "make vs. buy" of the components, where to build them, how much will it cost, can it be built, what is the most cost effective production process, and how fast can it be provided to the customer? The same models will be used to develop maintenance procedures and serve as an integral part of the required product utilization and maintenance training tools. Avi, tiff, vrml and vrml2 formatted data files are easily created for inclusion in product user manuals, manufacturing process plans, and product maintenance manuals. This information is easily made accessible to the end user via CDROM or Web based formats. The same product data definition data will then be used to determine the factory production requirements and can be integrated into virtual factory models to determine the most effective equipment layout and to quickly identify the costs associated to produce the product within this facility. A single database of product/part information minimizes the potential for an engineer or process planner to develop design modifications or process plans based on outdated versions of information. The ultimate goal of successful VM users is to eliminate the need for engineering change orders from ever occurring because the product functionality, processing, maintaining and manufacturing concepts were successfully demonstrated and evaluated in the virtual environment before implementation. In a global business environment VM can dramatically enhance ones ability to communicate by transcending potential language barriers.

Virtual Prototypes

In the past, new products were conceptualized by design engineers and then scaled physical mock-ups were fabricated to determine the fit-up of components. It was not uncommon for unusable parts to be produced due to an oversight or miscommunication during the design process resulting in lost time and the generation of scrap. In good situations the parts may only have required additional machining, filing or shimming to allow for assembly. This iterative process could take weeks or years on very complicated products such as ships. Virtual Prototypes (VP) created during Virtual Collaborative Engineering (VCE) sessions clearly convey to the customer the functionality of the product. The same prototypes also show the manufacturing staff what needs to be produced.

Deneb Robotics, Inc.'s ENVISION® software provides the user with the following capabilities which makes it ideal as a tool for Virtual Prototyping:

- fast, interactive 3D graphics
- integral, complete 3D CAD system
- configurable color graphs that display cycle information
- CAD data translators including IGES, Pro/ENGINEER, Unigraphics, DXF, Intergraph, CATIA and VDA
- dynamic viewpoint manipulation, with multiple viewpoints that may be independently controlled and the ability to attach a camera to any moving or stationary part
- realistic lighting models that provide the ability to display shadows, colored lights, and part material surface highlighting
- advanced file management system featuring a Visual File Interface enabling files to be selected by either clicking on an image with the mouse or typing the file name
- a real-time system for collision, near miss, and minimum distance checks

For a company that deals in a global business environment we believe that this type of physics based virtual modeling technology truly enhances ones ability to communicate and to transcend potential language barriers, because it is much easier for the human brain to absorb visual data rather than verbal or paper documentation regardless of your native language.

General Dynamics Electric Boat Division are currently designing, evaluating, and optimizing new submarines utilizing this VP approach. Operating under a contract from the Advanced Research Project Agency's (ARPA) Marine Systems Division, Electric Boat demonstrated the feasibility of state-of-the-art simulation-based design (SBD). The objective was to implement an accurate, efficient, and dynamic environment for design, rapid prototyping, concurrent product and process development, mission planning, operation maintenance, and training. Traditional prototyping methods can take several months to physically build and test, and at a great expense. As part of this program a demonstration model was created to evaluate a proposed torpedo loading mechanical system as shown in Figure 1. The geometry files were imported from the Computer Aided Design software into the

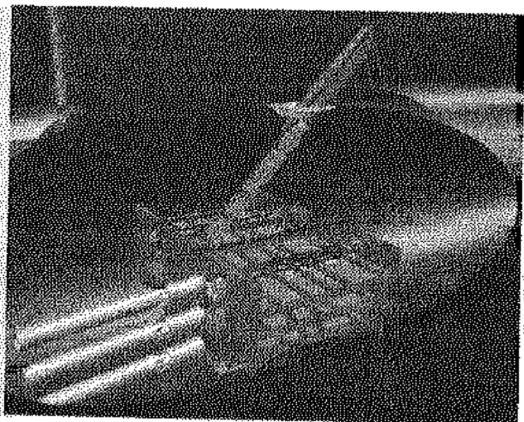


Figure 1

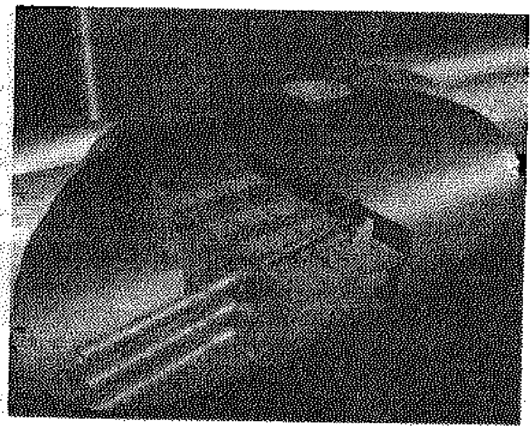


Figure 2

Envision VP modeling environment in the assembled condition. Within Envision, EB engineers were able to define the range of motion of the moving parts as well as the acceleration and velocities at which they were to move. The torpedo loading sequence was developed using Envision through a Graphical Simulation Language (GSL) to create a simulation of the process. Upon exercising the simulation, the collision detection capabilities of Envision captured a potential design flaw (Figure 2) which would have prevented the ability to use this mechanism as designed. The VP model was used to identify the colliding parts quite easily by highlighting them in red. Collision such as these will occur either because the functional sequence order was defined incorrectly or because there was not enough clearance designed in the system for the parts to move the necessary amounts. In this case, we saw that it was a spatial limitation. The parts were sent back to the Envision CAD world in the colliding positions, so that engineers could easily see the problem, and redesign one of the bulkhead structures, and thus eliminate the problem. This could all be done in a matter of a few hours as opposed to how much time and energy it would have required if it had been implemented on the real ship. The same model could conceivably be used to generate the process sequence programs for the controlling computer, and to verify that real control system sequence functioned correctly by connecting the model to the controller using digital/analog interface board. Other projects conducted in this program demonstrating the power of the Virtual Prototype included:

- A demonstration of the use of Envision to show the impact of various hull designs on the ship performance in a virtual sea based on the output of internally developed computer programs (Figure 3).
- The loading of a cargo ship at dock (Figure 4).
- The loading of a hover craft into the mother ship as they both floated on various sea state levels
- The Introduction of human modeling elements to validate the engine startup procedure. By mounting camera on the human model's eye it is possible to validate site lines within the ships various compartments.
- A simulation of the process of fire fighting on board a ship utilizing state of the art immersion goggles and human motion tracking technology.

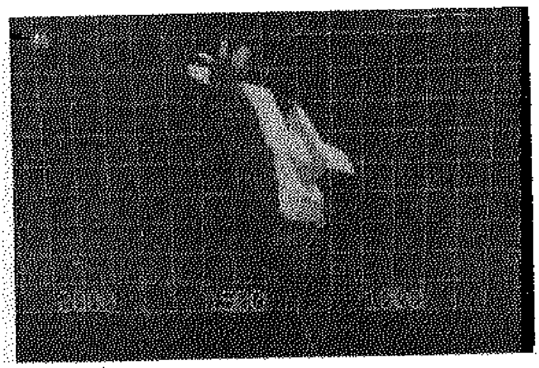


Figure 3

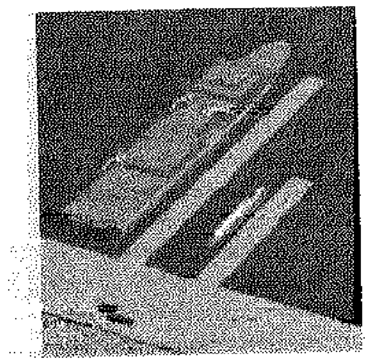


Figure 4

EB personnel generally feel that by simulating kinematic, dynamic and mechanical characteristics of a model submarine, its components and subsystems, engineers create a multidisciplinary environment in which to evaluate a wide range of parameters and optimize the design based on the results. Subsequent to the ARPA demonstration project, Electric Boat has adapted the SBD approach to design and now utilizes 26 seats of ENVISION VP software from Deneb Robotics, Inc. Envision fly-through design review sessions are conducted almost daily

Assembly Sequence Development

The end result of a typical CAD designing effort is the product and part geometry in the final assembled condition. It up to the manufacturing process planners to figure out how the pieces will go together and how they will be made. The Virtual Prototype can be of great value to this process as well. The Deneb/Assembly option for Envision VP provides the user with the ability to very quickly pull the various parts and subassemblies out of the final assembly state (using the collision and clearance detection capability to validate that the parts can indeed be removed through the space provided) via a dragging and dropping process with the mouse. During this process, a Gant bar chart is documenting the sequence. The Gant chart can be edited by cutting and pasting to change the order of disassembly, if necessary. The Gant chart can be exported into an ASCII text file for inclusion in the manufacturing process plans. Once the parts have been disassembled successfully, with a single mouse click, the sequence can be reversed, thereby generating the assembly sequence. The swept volume capability provided within Envision provides the user with the ability to create 3d geometric files that can be used to represent reserved space in the final design so that it will be possible for part removal in a maintenance operation. Once the assembly sequence is verified to be correct, tiff images, avi recordings, vrml and vrml2 files can be generated for documentation of the assembly process as well as the disassembly and re-assembly for maintenance processes. The use of Deneb/ASSEMBLY provides many benefits including improved communication, improved quality, reduced costs, and reduced time to market.

Virtual Process Models

Ergonomic Task Development

Deneb/ERGO option enables the user to rapidly prototype human motion within a workplace, then perform ergonomic analysis on the designed job(s). As mentioned above, this might be done when developing how to use a product, or maintain a product as well as how to build the product and its parts. The following functions and tools provide the ability to accomplish these tasks:

- a dedicated human motion programming interface that includes inverse kinematics and graphical programming for the human models
- 5, 50 and 95 percentile male and female models
- an energy expenditure prediction model
- guidelines for two-handed lifting
- a posture analysis system
- a work-measurement standard to estimate time standards for jobs

The features from ENVISION and Deneb/ERGO combine to give the user a versatile, flexible, and powerful tool for human factors engineering. In addition to studying the human element in the environment, Deneb/ERGO allows human models to interact with other moving entities like robots and conveyors. Figures 5 and 6 show some examples of some ergonomic models.

Figure 5

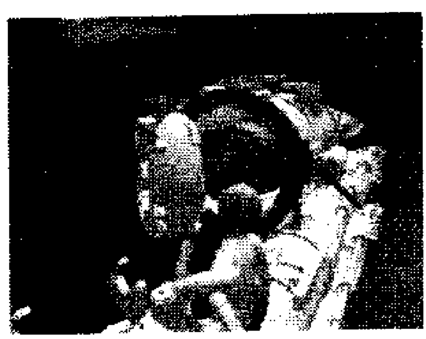


Figure 6

The human model functional activity is developed by the generation of poses that are recorded in the correct sequence for the operation. Graphical interfaces are created to help with this task by providing the user with an intuitively easy to understand format. For example when in the teaching mode, the first action for the human model might be to walk over to a table and grab an object. This is done by teaching the position at the table, and the software automatically generates the walking motion to that spot. Simple selections for reaching and grabbing the part are done via mouse clicks and then stored. Other poses such as sitting, stooping, kneeling and squatting are all generated with a couple of mouse clicks. Once the sequence of the operation has been completed within the model, the analysis of the process can then be performed. The output of this analysis will be information like the energy expended during the process, RULA (Rapid Upper Limb Assessment) assessment of the motion performed, NIOSH (National Institute of Occupational Safety and Health) lifting guidelines based on the identified mass of the objects being manipulated, and the time required to perform the task via the use of MTM-UAS (Methods Time Measurement - Universal Analyzing System).

Robotic Workcell Development

Fincantieri Shipyards S.p.A, Monfalcone, Italy, specializes in cruise and merchant ship construction. Welding stations at the Monfalcone shipyard include large multi-axis gantry systems provided by Motoman and IGM. The gantry systems support either two or four robots during the welding of large ship sections. The traditional welding method typically required 50 to 100 man-hours of on-line programming per one hour of robot cycle time. Therefore, a weldment requiring one hour of welding needed up to two man-weeks to program. As programming was done on-line, the robot was out of production the entire time. Shipbuilding is plagued with small weld batches as no two ships are identical nor are any two ship sections. Each section requires a unique program. In addition, the robot requires eight hours, or more, to weld one ship section. Skilled robot programmers, using the teach pendant pendent mode of programming, often takes as much as one month of programming time (two shifts per day) to complete each of the more than 100 sections in a typical ship

A virtual environment from Deneb Robotics, Inc. known as the Interactive Graphical Programming (IGRIP) provides the user with the ability to test out multiple styles and brands of robots, as well as how the robots might be integrated with a gantry style auxiliary axis positioning system. Figure 7 and 8 shows a close up of two different robotic welding workcell models that are used for the fabrication of structural sections. In this type of operation that consists of multiple

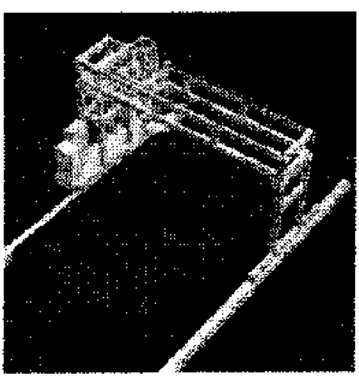


Figure 7

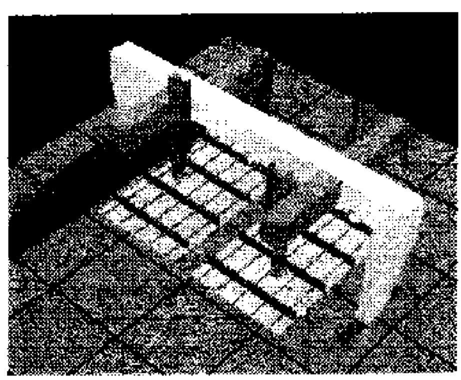


Figure 8

identical welding patterns being performed in a matrix like array, it is possible to install the robots in a master/slave (one robot leads, and the others follow the same process sequence) arrangement to increase the speed of the process. The workcells shown here were modeled with IGRIP with an optional software module called the Arc Welding Macro Programming system (AMP). The AMP system was developed jointly by a team of Deneb software engineers and Fincantieri process engineers, technicians, and operators to define the software requirements. It was designed to provide an easy to use arc-welding robot programming system for welders, not computer or robotic experts. It is within the AMP system that welding processing parameters are stored for typical types of welds. These are automatically set during the robot programming process by selecting the type of weld prior to generation of the robot motion sequence.

By developing IGRIP workcell models during the conceptualization phase of the purchase of the robotic arc-welding stations, Fincantieri was able to verify that the actual system would be able to gain access to all the necessary areas of the ship components. The actual programming of the robots was initialized before the real system was installed in the Monfalcone shipyard. Fincantieri now achieves 100 percent off-line programming. No touchups are needed and all robot process data is downloaded, including weld parameters, enabling the robot to maintain constant production. This is the critical benefit of off-line programming.

These same workcell-modeling tools are being used very successfully in aerospace, automotive and other commercial industries as well. Other specialized option packages are available for painting, finishing (deburring), spotwelding and telerobotics.

CNC Controlled Machine Modeling

Virtual machine models can be created to show the entire machine functionality during the removal of material in a machining operation. The generation of the NC code is done through the existing CAD/CAM system. This type of model is very easy to use to verify the functionality of a NC program prior to downloading it to the actual machine. The models can provide information on the processing times, collision detection and the efficiency of the program for processing materials. Deneb Robotics, Inc.'s Virtual NC modeling software is capable of identifying machining conditions, which have, been proven to cause chatter, reduce scrap through the detection of undercuts and gouges, and detect conditions which contribute to tool breakage. Traditionally, machining cycles waste up to 30 percent to 40 percent of the cycle time. This wasted time can be attributed to the use of slower feed motion when a rapid motion is acceptable, performing unnecessary motion, excessive motion, and/or requesting repetitive tool changes.

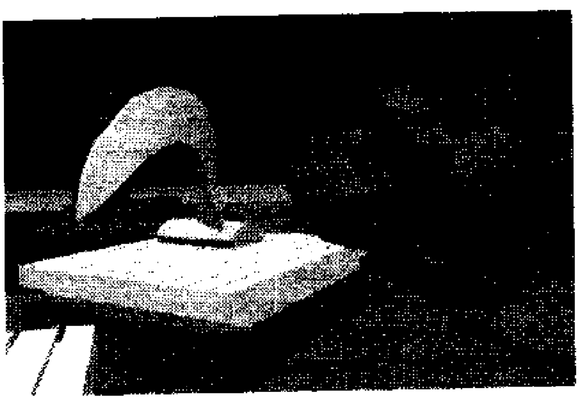


Figure 9

Lockheed Martin Energy Systems, Inc. (Energy Systems), Oakridge Y-12 Plant, TN, successfully modeled an 8-axis Giddings & Lewis eight inch boring mill with CNC control as part of a federally funded project for manufacturing process development for the Navy's Seawolf submarine program. The machine model (Figure 9) was created and validated by a team of Lockheed Martin and G&L engineers as part of the system design process. The model was used to develop the post-processor and probing routines for the machine before it was completely fabricated. While the machine was still being fabricated a team of nine NC programmers were developing and validating the NC codes that would be used to machine the parts required in this program. Using Deneb/Virtual NC, they significantly reduced setup time and began part production as soon as the equipment installation at the manufacturing site was complete. Tooling and fixture setups were also verified and readied for production during this time. Because simulation allowed these steps to be done in advance, actual part production could begin the day the machine was turned over to the production shop.

During the simulation of a particularly complex flat-end mill operation, for example, the software indicated a collision. It was not at the tool/workpiece interface but at a point about 40 inches from the tool and outside the operator's field of vision. The manufacturing engineer solved this problem by extending the tool an additional two inches, and sent the data back through the post-processor to create a new program. When the new program was simulated, the software detected no collisions. Prior to machining the part, the machine operator was called to watch the simulation so he would be prepared for the closeness of the clearances. Virtual NC automatically identifies the conditions in the cycle when a feed move has been programmed, but no cutting actually occurs. This information can be used to restructure the NC program to reduce this wasted cycle time.

Energy Systems' experience showed that use of Deneb/Virtual NC for machine tool and machine process simulation dramatically reduces costs by preventing crashes that can disable the machine and delay the completion of the project. Simulation also reduced hidden costs by optimizing machining processes, improving the effectiveness of operators and engineers, and improving process capability. One of the most significant benefits of Deneb/Virtual NC on this project is that 500, 8-axis NC programs have been used on the machine without a single crash. This program dramatically should how the virtual modeling provided a means to eliminate the risk of mechanical damage to the machine tool, workpiece, fixturing, and tooling. They were able to develop and debug the post processors with greater ease and ahead of time. The models were also used to train both NC programmers and the machine operators before the machine was installed, and as an offline training tool while the machine was producing parts.

Virtual Factory Modeling

In the past traditional factory modeling tools have been used throughout industry to determine how many machines were needed to provide some target production level. Once the answers were provided, the modeling effort stopped. Early factory modeling tools were cumbersome to use and the accuracy was often suspect because the results were numerical tables which needed to be interpreted by skilled professionals. Due to the visual nature of Virtual Factory (VF) modeling tools, it is now

Figure 10

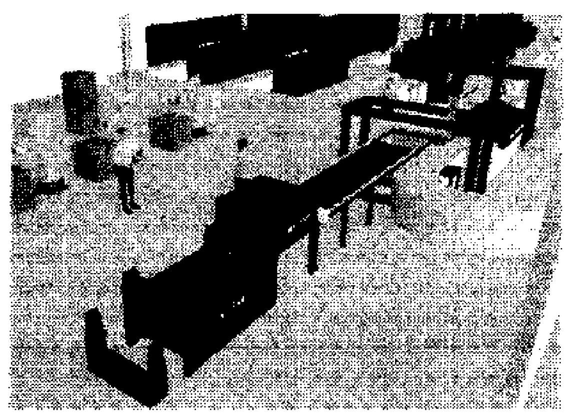


Figure 11

much easier to convince a customer that the model truly represents the facility functionality. Figures 10 and 11 show views of a Virtual Factory model generated by United Technologies Research Center personnel of a proposed composites fabrication facility for the High Speed Civil Transport Program. It is also much easier to conduct "what if" scenario analysis because the changes are very clearly seen and understood. VF models clearly provide the users with an:

- Improved understanding of factory functionality
- Identification and quantification of throughput bottle-necks, WIP, equipment/labor utilization, impact of down time and maintenance
- More accurate estimates of production costs
- Continuous improvement "What If..." analysis
- Evaluation of alternative scheduling approaches

VF models provide the user with integrated graphical presentation capabilities such as real time strip charts, pie and bar charts as well as object color changes to indicate resource status. Model summary reports quickly identify the material flow bottlenecks.

Many companies throughout the world are adopting the principles of Just in Time (JIT) and "Kaizen" (Continuous Improvement). Entire manufacturing facilities are being evaluated and re-organized to minimize the occupied real estate area, minimize part distance traveled, and minimize the product production cycle. VF models have been successfully used to enhance this process. An "as-is" VF model of an existing facility is used to track and show how the material flowed between resources. Spread sheet data files are used to input information for the part routing through the facility, as well as processing times, setup times, teardown times, labor, tooling and fixturing requirements. The distances parts travel and the time in the facility for the parts are automatically recorded and saved in a spread sheet format. By sorting and analyzing this spread sheet information it becomes clear which pieces of equipment need to be moved to minimize the distances the parts will travel. This is done in an interactive, iterative manner to determine a more efficient facility layout. Total part travel distances can be reduced by over 50% based on the implementation of equipment rearrangement determined by this "Kaizen on the Computer" activity. Virtual Factory modeling tools dramatically enhance the ability to understand a facilities functionality and serve as a means for continuous improvement in an organized and efficient manner. They should be used as "living" documentation of the facility functionality.

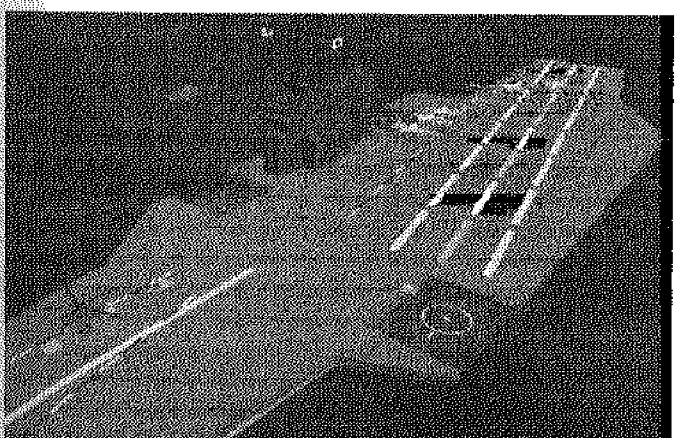


Figure 12

For the ship building industry the scenarios described above still apply for the land-based manufacturing facilities that create the components of the ship that are assemble at the ship yard. VF models could also be used to evaluate the functionality of the crew utilization, onboard shop functionality, chow line flow, and cargo loading and unloading. These models could also be used to evaluate the flow of ships through the yard to determine the quantities of dry docks and wet docks. Because these tools are physics based in nature, it is possible and necessary to create the models dimensionally accurate. Distances that ships, parts and people need to travel obviously determine the amount of time it takes for material to flow between locations. Figure 12 shows a sample model of the flight deck on an aircraft carrier, and Figure 13 shows a sample of a model of a shipyard.

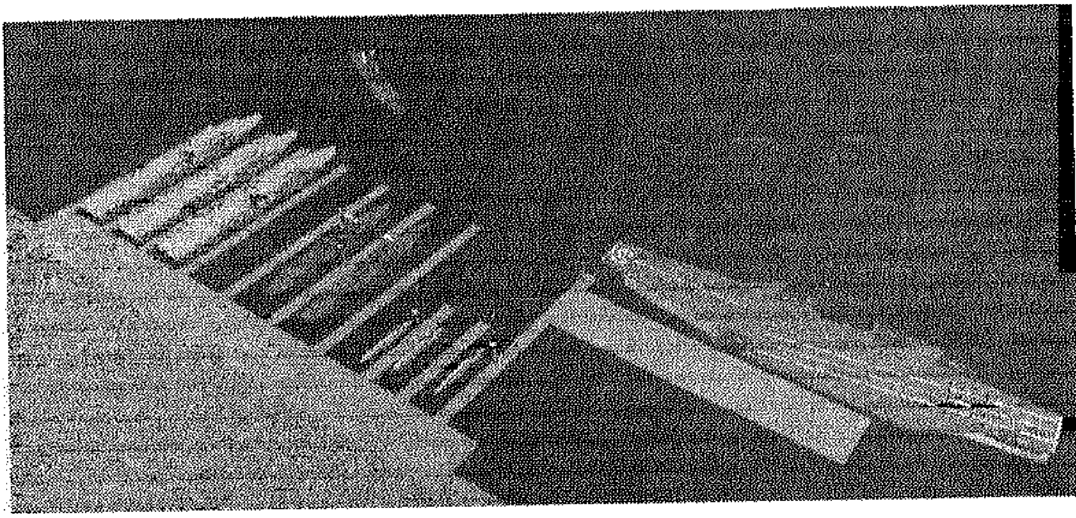


Figure 13

Virtual Manufacturing Benefits

Besides improved communications across multi-disciplinary product teams, specific benefits demonstrated and projected for key industries include:

- 50 percent reduction in time to market based on the increased speed of decision making and the parallel development of product and process designs
- $1-5 billion life cycle cost savings for new Navy/Air Force Aircraft development and production
- Achieve learning curve of third ship set on first ship set for new shipbuilding program
- Integration of global customers, subcontractors, suppliers and users early in the design and development cycle
- 25 percent reduction in the cost of new products due to the utilization of the Design for Assembly and Design for Manufacturing capabilities
- 70-80 percent reduction in rework costs for tooling and fixtures since they were proven out in the Virtual Manufacturing environment before initial fabrication

Conclusions

Virtual Manufacturing is being aggressively implemented across key industries including Aerospace, Automotive, DoD, DOE, Nuclear, R&D, and other industries to become more competitive, shorten launch cycles, and improve communications across global enterprises. Companies that have adopted Virtual Manufacturing technologies for the launch of new products and streamlining of operations include General Dynamics, General Motors Corp., McDonnell Douglas, Lockheed-Martin, Caterpillar, Northrop-Grumman, United Technologies and many other industry leaders. Time to market is being reduced up to 50 percent and lifecycle cost savings of 30 percent are being realized through implementation of Virtual Manufacturing technologies. Seventy to 80 percent of the tooling re-engineering and rework dollars have been saved by building the fixtures right the first time.

In the not to distant future we believe that the VM tools described above will provide us with an easy to use system for enhanced global communication. It will become common place for design engineering teams and their customers to communicate together while being simultaneously immersed in the same virtual environment without leaving their office. Vendors and equipment suppliers will provide virtual models of their equipment for NC creation, verification and training tools, as part of their deliverable package. We will be able to provide factory tours to people, showing them their components being manufactured without them leaving their office. Virtual Factory Models will become "living" documentation of a facilities functionality by the direct connection to the real facility resources for on-line data collection, tracking of equipment, material and labor status.

We believe that VM Technology enhances our ability to communicate product functionality, new concepts for component fabrication and verification of the factory functionality required to produce these products. The visual nature of VM provides us with the ability to transcend potential language barriers in a global business environment.

References

Conference Proceedings

1. Brown, R.G., Deneb Robotics, Inc., "An Overview of Virtual Manufacturing Technology", Advisory Group for Aerospace Research and Development (AGARD) Proceedings, May 6-10,1996
2. Favretto, A., "Arc Welding Robots Off-line Programming – An Experience of Automation in the Fincantieri Monfalcone Shipyard", Deneb User Group 1995 Proceedings, 1997
3. Scotton, T.W., United Technologies Research Center, "Virtual Manufacturing", Deneb User Group 1995 Proceedings, October 1995

Internal Deneb Robotics, Inc. White Papers

1. Nayar, N., Ph.D., "Deneb/ERGO® *For Ergonomic Assessment*", Deneb Robotics, Inc.
2. Sobetski, D., "Deneb/ASSEMBLY® *For Assembly Simulation*", Deneb Robotics, Inc.
3. Eshbaugh, J., "Virtual NC®" Deneb Robotics, Inc.

Deneb Robotics, Inc. – Customer Testimonial Marketing Documents

1. Arc Welding: Fincantieri Achieves 100 Percent Off-line Programming
2. Machine Tool Verification, Process Verification – Lockheed Martin Verifies 500, 8-axis NC Programs Without a Single Crash

3. Simulation Based Design: Next Generation Submarine being Developed with Virtual Technologies
4. Simulation Based Design: Systems Engineering and Assessment Ltd.(SEA) Simulates Naval Operations with ENVISION

Technical Papers

1. Beaudreaux, J., General Dynamics — Electric Boat, "ARPA SBD Contract."
2. McDonald, M, Sandia National Laboratories, "Virtual Collaborative Engineering," 1996
3. Rossi, J., McDonnell Douglas, "DMAPS"

Periodicals

1. *Armed Forces Journal*, "New and Newer Submarines," July 1995
2. *R&D Magazine*, "New Horizons for Aerospace — Simulate Your System Before you Build it," January 1996
3. *Fastrack*, Virtually Simulated Machining," February 1996
4. *Fortune*, "The Big Comeback in Robots," March 1996
5. *CIME*, "Full Speed Ahead," Spring 1996

PRACTICAL APPLICATION OF COMPUTER SIMULATION MODELING FOR ANALYSIS OF SHIPYARD PRODUCTION PROCESSES

Jeffrey Schaedig, National Steel and Shipbuilding Company, San Diego, CA
John Horvath, National Steel and Shipbuilding Company, San Diego, CA

Introduction

In an effort to build cost competitive ships for the world market, many shipyards are investigating building methods and equipment that will streamline production and reduce vessel construction times. With simple production process systems, this investigation is a fairly straightforward process. The existing "As-Is" production process is thoroughly analyzed through the use of flowcharts and performance statistics, then compared to a "To-Be" process incorporating new techniques and equipment. If the "To-Be" process achieves the objectives of management with a justified return on investment, the new process is implemented.

The construction of a ship consists of thousands of interactions between products, processes, and resources. All three entities have an effect on each other and a small change to one can cause a large change in the overall operation of the system. Many of the techniques utilized to analyze smaller systems become very labor intensive when manually analyzing ship production processes. Assumptions are used to break the process down into manageable pieces to be studied by a process improvement team. In some cases, the necessity of making these assumptions can cause problems by oversimplifying the process. The same details that make the system difficult to analyze are the details that contribute to the overall inefficiency of the process. Advances in personal computer processing speed and the development of user-friendly simulation software, however, now provide shipyards with an effective tool in performing investigations of difficult to analyze processes.

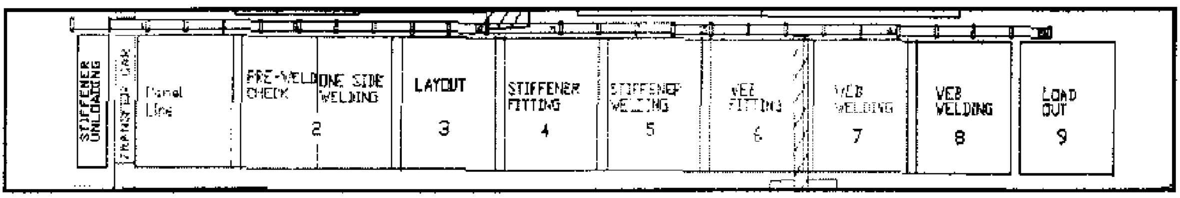
National Steel and Shipbuilding Company (NASSCO) in San Diego, California has been using computer simulation to make these difficult to analyze processes more manageable. The manufacturing simulation software, ProModel, was used as part of a process improvement initiative to reduce the cycle time of the yard's Panel Line to 4 hrs/panel. The variation of the work content in panels built on the line and the interaction between workstations made this a formidable task. Through the combined efforts of the team members, valuable insight into the everyday operation of the line was gained. This paper highlights the steps taken by the Process Analysis Team to analyze the Panel Line, NASSCO's use of computer simulation software in this initiative, as well as, the results of the analysis.

Description of NASSCO's Panel Line

The layout of NASSCO's panel line is shown in Figure 1. The Panel Line consists of nine process specific stations. Plates enter the line at Station 1 where they are fit and tack welded together. The fitting and welding of deck sockets also begins in this station. The ships currently being built at

NASSCO are Roll-On/Roll-Off ships for the Military Sealift Command. In order to facilitate the lashing of cargo, thousands of cloverleaf-shaped holes are cut into the deck of the ship.

Figure 1. Layout of NASSCO's Panel Line



Bowls are welded to the underside of the deck to prevent water from flowing through the holes. These bowls are referred to as "deck sockets." Deck socket welding continues as the panel moves by conveyor into Station 2. For the majority of the panels built on the line, the deck socket welding is completed in this station.

In Station 2 the plates are welded together with a semi-automatic one-sided welder. If any weld repairs are needed they are done in a pit between Station 2 and 3. The welded panel is then moved into Station 3 where layout of the longitudinals and transverse deck beams is done. The perimeter of the panel is burned to create a square panel. Longitudinals are manually fit in Station 4 and welded in Station 5. A semi-automatic welder is used in Station 5 to do the welding of the stiffeners. The welder is capable of welding both sides of four stiffeners simultaneously. After the stiffeners are welded, the panel moves into Station 6 for the fitting of transverse deck beams (webs). The welding of the webs and any additional small parts is done manually in Stations 7 and 8. Finally, the completed panel is moved into Station 9 where the inspection is done. If any additional welding needs to be completed on the panel it is also performed here before the panel is lifted off the line by a gantry crane and moved to the next process location.

Panels are moved into the next station as soon as possible and removed from the line when complete to make room for new panels. Removal of the panels is done in four locations including the end of the line. Blanket plates (panels without longs and webs) are removed after the one-sided-welder in Station 2. Panels that only require longitudinals are removed at Station 5. In some cases, the weldout of the webs is completed in the first weldout station (Station 7). When this occurs, the panel is immediately removed from this station.

Initiative to Implement the 4hr Line

NASSCO's objective in reducing the cycle time of the Panel Line was to increase throughput and efficiency. By implementing a line that consistently indexes every 4 hours, it would also be easier to schedule starts and lift-offs. On average, a crane could arrive at the beginning, middle and end of every shift to remove a panel from the line. A process improvement team was formed consisting of a cross-section of the Panel Line's process stakeholders. The team's task was to recommend the changes to the process that would be necessary to reduce the line's cycle time to 4 hrs/panel. Cycle time for the line was defined as the elapsed time between finished panels. The team worked on reducing the cycle time primarily through the elimination of non-value-added tasks and replacement of aging and inefficient equipment.

Process Analysis of the 4hr Line

The first operation of the Process Improvement Team was to flowchart the existing Panel Line process. Included in the flowchart was the average time and required manning for each task. Because of the complexity and scale of the Panel Line operation the 80/20 rule was used to determine which tasks should be included in the flowchart. If the task was performed on 80% of the panels built on the line, it was included in the flow. To further simplify the process a "standard" reference panel was created to represent the average work content of the panels run down the line. The work content of each panel was characterized by the number of plates, longitudinals (stiffeners), manufactured T's (webs), and deck sockets used to build the panel. The components were averaged for an entire ship's worth of panels to determine the number of parts in the reference panel. These averages were used by the Process Improvement Team for the reference panel and consisted of:

- 5 Plates (4 Seams) – Average 4.4
- 12 Longitudinal Stiffeners – Average 11.6
- 5 Transverse Manufactured T's (Webs) – Average 4.8
- 120 Deck Sockets – Average 120

After flowcharting the existing process, the Process Improvement Team reviewed the tasks to determine which added to the inefficiency of the panel construction either by not contributing to the physical change of the product or because of the use of ineffective equipment. These tasks were eliminated to improve the efficiency of the process. The order of many tasks was changed to accomplish them in parallel rather than series, and some work was moved off the line to further reduce each station's individual cycle time, thereby reducing the overall panel cycle time to 4 hours. The effects of these changes were quantified in terms of new times and manning and included in a flowchart of the "To-Be" process.

The times for each task on the "To-Be" flowchart were summed along the critical path to determine the individual station times. The station times for the "To-Be" Panel Line are shown in Table 1. One of the changes the Process Improvement Team made in the "To-Be" process was the timing of the panel movement on the line. Instead of moving a panel into the next station when it was completed, the entire line moved every four hours. This was done to simplify the manual analysis of the line. Since the whole line moved at the same time, each station had the same 4 hour average station span time (Avg Span). Span time was defined as the duration each panel occupies a station before being moved to the next one. The amount of time the panel was being worked on during its span time in the station was defined as the process time (Avg Process). The process times developed by the Process Improvement Team were obviously less than the 4 hour span time otherwise the average panel would have moved into the next station uncompleted. The maximum process time (Max Process) for the stations shown in Table 1 is equal to the average station process time since all of the panels are the same "standard" reference panel. The line span time (Average Line Span Time Per Panel) was defined as the time the panel is on the line from start of assembly to finish. The one-sided welding station has the highest process time (236 min), and thus is the pacing station for the 4 hour line.

Table 1. Station Times for the "To-Be" Panel Line (Times in Minutes)

	Process Analysis-Ref Panel		
Station	Avg Span	Avg Process	Max Process
1-Plate Fitting	240	180	180
1 and 2-Deck Sockets	n/a	n/a	n/a
2-One-Sided Welding	240	236	236
3-Layout/Per Burn	240	220	220
4-Stiffener Fitting	240	203	203
5-Stiffener Welding	240	218	218
6-Web Fitting	240	187	187
7-Weld Out 1	240	180	180
8-Weld Out 2	240	180	180
9-Inspection and Ship	240	121	121

Avg Panel Cycle Time	4.00 hrs (240 min)
Avg Line Span Time Per Panel	36.0 hrs

Computer Simulation Modeling of the 4hr Line

Although the process flow had been completed for the "standard" reference panel, there was a realization that the reference panel only represented a portion of all of the possible panel types built on the line. The effect of building different panel types that have varying work contents was unknown. It was decided that a computer simulation of the panel line was necessary to quantify the effect of building panels with variable work content on the "To-Be" Panel Line.

Development of the Panel Line Model

In order to quickly set up a model that would evaluate the "To-Be" process, the project team decided to enhance a Panel Line computer model originally developed for NASSCO by a local consultant, Kiran Consulting Group, for the purpose of investigating the use of robotics on the line. The project team needed to modify the computer model so it would be better suited for its new purpose of analyzing the "To-Be" Panel Line process. In Kiran's model, a single reference panel, which was different than the Process Improvement Team's reference panel, had a station time associated with it for each fitting and welding process that took place on the line. Some of these times such as the deck socket and web welding operations were dependent upon the number of people assigned to the task. The line, therefore, could be balanced by increasing or decreasing the manning for tasks that were creating bottlenecks in the line or had low resource utilization. Automation could also be substituted in place of manpower to determine the overall effect on the system. Manning and automation, however, were not the key issues in the "To-Be" Panel Line process. The model needed to be adapted to accept panels of different work contents rather than a single reference panel.

The first step in adapting the Panel Line model was to determine the level of detail necessary to accomplish the objectives of the task. When conducting a simulation project it is important to remember that the time and funding necessary to build the model is directly proportional to the detail of the model. As the detail of the model increases, so does the cost and effort necessary to build, debug, validate, and use it. Since the model's primary use was to determine the effects of building variable work content panels on the panel line rather than optimizing the line through process changes, design changes, and manning, it was decided that each individual task and required manning level did not have to be explicitly modeled. The fact that the model was to simulate the system variation due to work content and schedule, however, meant that it could not be modeled by merely inputting only the overall assembly times for each panel. The model had to be built in a fashion that captured both individual panel complexity and the effects of overall panel build sequence. These objectives were met by modeling the fitting and welding times on a per plate, long, and web basis to capture the variation in work content, and designing the model to accept a specific panel sequence.

Modeling Process Times

In order to develop the simulation model's assembly times for both reference and variable work content panels, the Process Improvement Team's reference panel and station times (shown in Table 1) were used to determine the process time per seam, deck socket, long, and web. For example, the Average Process time for the reference panel in Station 1 (180 minutes) was divided by the number of reference panel seams (4) to get the fitting time per seam at Station 1 (45 min/seam). This is the only time needed in order to capture the processing time for Station 1 when the identical reference panels are run through the model. When panels with variable work content are modeled, however, both the number of seams *and* the length of each seam will alter the total fitting time in the station. In order to represent this variation from panel to panel, the fitting time per seam was divided again by the number of reference panel webs (5) which run perpendicular to the seams to capture the effect of differing seam lengths. This resulted in a processing time per-unit for Station 1 of 9 min/seam/web. This same process was used to develop per-unit times for all nine stations on the line and captured the work content of both the reference and variable work content panels.

Additional time for tasks such as the setup of equipment, transportation of the panel, and clean up was not explicitly modeled, but included as part of the process time on a per plate, long, and web basis. Downtime was not included in the modeled processes for two reasons. First, it would take some time to analyze how it currently affects the process, understand how it would affect the "To-Be" process, and develop it into a form that could be utilized in the model. Second, it was not necessary to include it, initially, to meet the basic objectives of the project. If the results of the model indicated that a 4 hour cycle time could not be achieved in a system without downtime, then the project team could safely conclude that it would be impossible to achieve the goals in a panel line process with downtime. There was no point in spending the time to include downtime until it was known if the "To-Be" process could meet its objectives without it.

Modeling Process Flow

Because the computer simulation model could control the movement of each panel into the next station as soon as possible, it was not necessary to constrain the line by forcing each panel to have the same station span time. As mentioned earlier, the entire line moved every four hours in the "To-Be" process done by the Process Improvement Team to simplify their analysis. This restrains the throughput when variable work content panels are built. Smaller panels with a work content less than 4 hours in duration will not move off the line as quickly as they would if they were allowed to move to the next station when completed. Moving the panel as soon as possible, not only increases the throughput of low work content panels, but allows work to begin on the following panel earlier, creating a more efficient use of the line's limited space. This is the main reason the current practice on the actual NASSCO panel line is to move each panel into the next station as soon as possible.

Modeling Process Sequence

The final change necessary to adapt the existing Panel Line model for this project was to program it to read in a schedule of reference or variable work content panels by means of a data file. This schedule, created by the Master Planning Department, included all of the panels built on the Panel Line in the sequence in which they were built. It also detailed the number of plates, deck sockets, longs, and webs on each panel. The information is read in and mapped to variables in ProModel where it is used to calculate the process times. The first plate of the first panel arrives in Station 1 at the start of the simulation and from that point, the process times drive the rate at which panels are completed. During the simulation, ProModel keeps track of performance statistics on location and resource usage, quantities of parts, and process times. This information can be reviewed at the end of the simulation in both tabular and graphical form.

Results from Model for the Reference Panel

Once the model was adapted to represent the Process Analysis Team's "To-Be" process, it was run to validate that it produced the same performance. In order to do this, a schedule of 290 identical reference panels was created and entered into the input data file. This is the same number of variable work content panels that would be tested. The results of the reference panel model are shown in the "Simulation -Ref Panel" column in Table 2.

Table 2. Comparison of the Process Analysis Team's Reference Panel Station Times and the Model Reference Panel Times (Times in Minutes)

	Process Analysis-Ref Panel			Simulation -Ref Panel		
Station	Avg Span	Avg Process	Max Process	Avg Span	Avg Process	Max Process
1-Plate Fitting	240	180	180	239	180	180
1 and 2-Deck Sockets	n/a	n/a	n/a	29	28	29
2-One-Sided Welding	240	236	236	239	236	236
3-Layout/Per Burn	240	220	220	220	219	220
4-Stiffener Fitting	240	203	203	203	203	203
5-Stiffener Welding	240	218	218	218	217	218
6-Web Fitting	240	187	187	187	187	187
7-Weld Out 1	240	180	180	144	143	288 (143)*
8-Weld Out 2	240	180	180	216	216	360 (216)*
9-Inspection and Ship	240	121	121	121	121	121

Avg Panel Cycle Time	4.00 hrs (240 min)	3.99 hrs (239.4 min)
Avg Line Span Time Per Panel	36.0 hrs	30.2 hrs

* Max Due to Startup. Steady State Max Shown in Brackets

The Average Span times of the model's reference panel are less than or equal to the span times of the Process Improvement Team's reference panel. This is an effect of moving each panel as soon as possible into the next station. The Average Process times for both are equal with the exception of Station 7 and 8 Weld Out. This is the natural balance of the work in these two stations if panels are moved into Station 8 when it becomes available in order to make room for additional panels on the line. The sum of the work done in both stations of the simulation model, however, is still equal to the 360 minutes of work done in Stations 7 and 8 of the "To-Be" Panel Line process. Because 290 identical panels were used in the reference panel simulation model, the Maximum Process times are equal to the Average Process times for all of the stations.

The six hour reduction in Average Line Span time per panel is a minor issue in the comparison of the times. Because the panels are able to move to the next station when they are finished, the panels spend less time on the line. The biggest advantage to this is that the first panel comes off the line 6 hours earlier in the model. This reduction in span time is important in the scheduling of panels for the line since there is a 6 hour reduction in time from start to finish, but it is not an issue in the performance of the line. The most important factor in judging the performance of the line is the Average Cycle time per panel. This is the rate at which finished panels will come off the line, and is equal for both the Process Improvement and Simulation results. Because of the close agreement between the times developed by the Process Improvement Team and simulation model, the project team felt that the model was validated and was a reasonable representation of the process. The next step was to load a schedule of panels consisting of variable work content into the model.

Results of the 4hr Line with Variable Work Content Panels

Two hundred ninety panels with variable work content were input into the model rather than the identical reference panels. A portion of the data file is shown in Table 3.

Table 3. Variable Panel Input File

Panel Type	Plates	Seams	Sockets	Stiffeners	Dart 4 Passes	Webs
V188-5001	2	1	0	3	2	1
T167-A5001	3	2	54	7	2	4
T168-A2	2	1	36	7	2	4
T223-A5001	4	3	54	7	2	6
T123-A5001	4	3	76	8	2	5
T125-A5001	4	3	7	8	2	5

The results of the Variable Work Content Panel runs are shown in Table 4. The Average Cycle time per panel is 0.63 hours greater than that for the reference panel. This indicates that using the same processes developed for the reference panel will not result in a 4 hour cycle time for panels with variable work content. The 4.63 hours per panel is an optimistic view since downtime has not been included in the model. Any disruption in the normal operation of the line will drive the average cycle time even higher.

The Average Span time and Maximum Process times are higher for the panels with variable work content. The Average Process times are not equal to either the Average Span or Maximum Process times (since the panels had variable work contents) and most are lower than those for the reference panel. One would expect that a lower average process time would contribute to a lower panel cycle time. The larger Average Span times, therefore, must be the cause of the higher cycle time.

Table 4. Comparison of the Times for Panels with Variable Work Content (Times in Minutes)

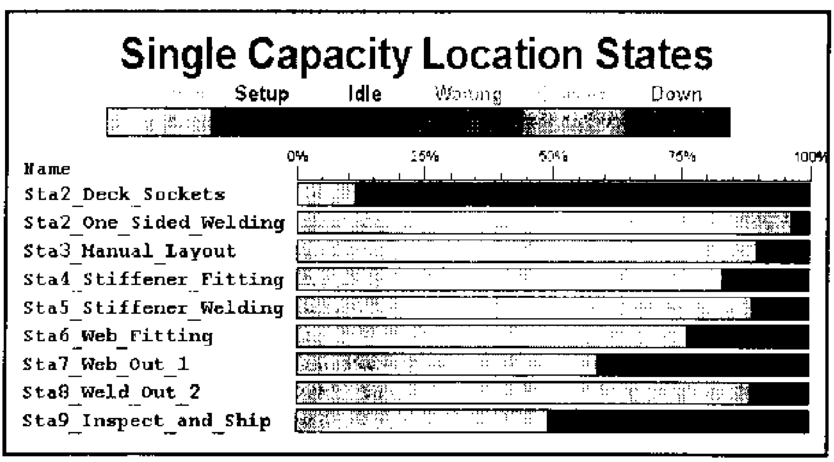
	Process Analysis-Ref Panel			Simulation -Ref Panel			Simulation-Variable		
Station	Avg Span	Avg Process	Max Process	Avg Span	Avg Process	Max Process	Avg Span	Avg Process	Max Process
1-Plate Fitting	240	180	180	239	180	180	280	144	378
1 and 2-Deck Sockets	n/a	n/a	n/a	29	28	29	273	30	160
2-One-Sided Welding	240	236	236	239	236	236	277	189	496
3-Layout/Per Burn	240	220	220	220	219	220	275	188	417
4-Stiffener Fitting	240	203	203	203	203	203	267	195	396
5-Stiffener Welding	240	218	218	218	217	218	253	236	436
6-Web Fitting	240	187	187	187	187	187	191	171	356
7-Weld Out 1	240	180	180	144	143	288 (143)*	110	110	360
8-Weld Out 2	240	180	180	216	216	360 (216)*	294	290	684
9-Inspection and Ship	240	121	121	121	121	121	121	121	121
Avg Panel Cycle Time	4.00 hrs (240 min)			3.99 hrs (239.4 min)			4.63 hrs (277.8min)		
Avg Line Span Time Per Panel	36.0 hrs			30.2 hrs			35.3 hrs		

* Max Due to Startup. Steady State Max Shown in Brackets

Causes of Higher Span Times In Variable Work Content Model

The higher span times are caused by blockages in the line. The Location States for the Reference Panel model are shown in Figure 2. The Location States for the Variable Work Content model are shown in Figure 3. These are the percentages of time that each location was in use (Operation), being setup for processing (Setup), not being used (Idle), waiting for resources to process a panel (Waiting), unable to move a completed panel into the next location (Blocked), or unable to process a panel due to planned or unplanned downtime (Down). The Figures show that the stations were either in operation, idle, or blocked during the simulation. Setup and downtime is not seen in the graphs since they were not explicitly modeled, and enough resources were used to prevent any wait time.

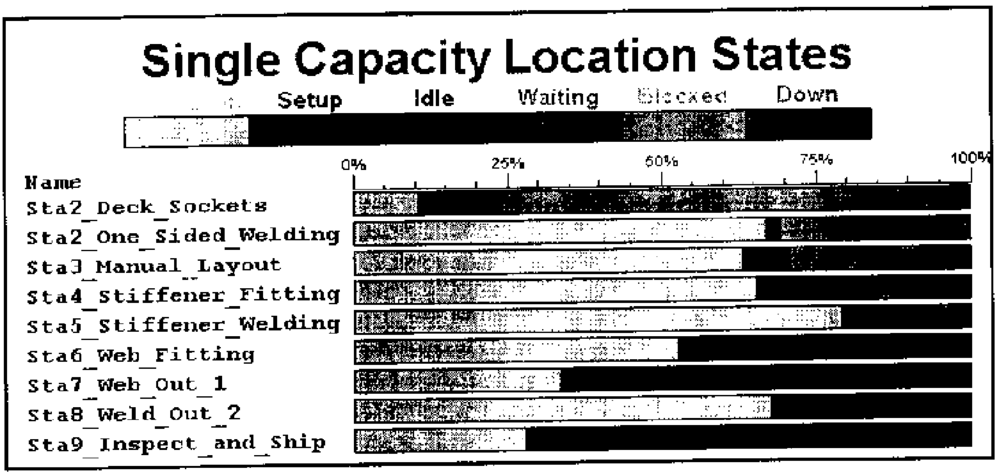
Figure 2. Location States for the Reference Panel



With the exception of Station 2, 7, and 9 all of the locations in the Reference Panel model are in operation for a reasonable percentage of the time. There is very little idle time. Station 2, 7, and 9 have the lowest process times and therefore, are in operation less than the other locations in the balanced system.

Figure 3 shows a much lower utilization of the locations in the Variable Work Content model. The remainder of the time is spent with each station either idle or blocked. The blockage is created whenever a downstream panel prevents a completed panel upstream from moving into the next station. Figure 3 shows three major jumps in the amount of blockage in the line. Starting at the end of the line, the first occurs between Stations 6 and 7 where the blockage time increases from near zero to a little more than 5%. The second occurs between Stations 4 and 5, and the third between Sta2_Deck_Sockets and Sta2_One_Sided_Welding.

Figure 3. Location States for the Variable Work Content Panel



These three jumps correspond to the three major bottlenecks in the line:

- Station 2 One-Sided Welding
- Station 5 Stiffener Welding
- Station 7 and 8 Weldout

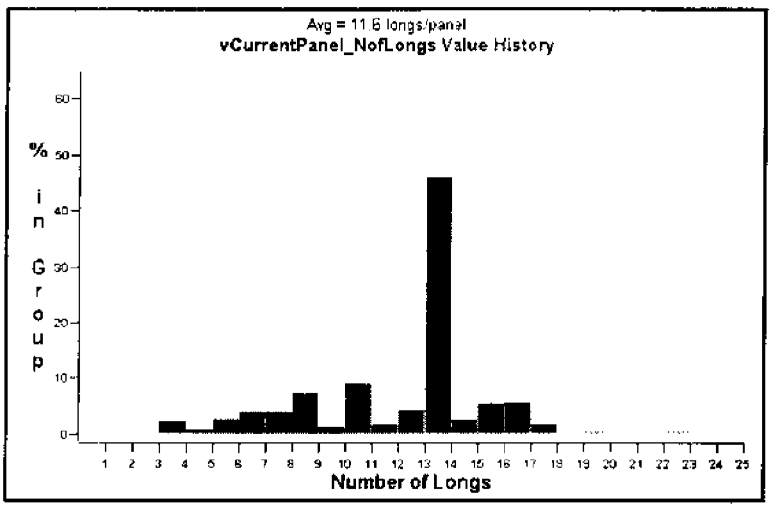
Station 8 is also considered to be a bottleneck due to the shared work and dynamics between itself and Station 7. When Station 8 becomes available, the panel in 7 is moved into 8 regardless of work content left on the panel. The next panel that moves into Station 7 is processed until it is completed (and then taken off the line) or Station 8 is again available. A possible scenario could exist whereby panels in Station 7 and 8 could be in process while the panel in Station 6 is complete. Station 7 in this case is not the bottleneck since it is ready to move when 8 becomes available. Until Station 8 is available, the panel continues to be processed. The blockage in Station 6, therefore, is due to the longer process time for the panel in 8.

Causes of Blockages

The blockage in Sta2_Deck_Sockets is primarily due to panels which have either no deck sockets or a number that can be completed in Station 1. If there is no work to be completed on the panel when it enters Sta2_Deck_Sockets it will want to continue on into Sta2_One_Sided_Welding. If a panel is in Sta2_One_Sided_Welding, this move cannot take place resulting in blockage time. Since Sta2_Deck_Sockets acts as a buffer station, this blockage is not necessarily a problem for the line. Understanding that the blockage exists, however, does present the opportunity for increasing the throughput of the line. By reducing the Average Span and Process times of the One-Sided Welder, the throughput of the line will increase providing that it is not reduced below the span times of the Stations after the welder.

The blockage caused by the stiffener welder in Station 5 is due to the variation in the number of longs that are on the panels welded in this station. Although averages were used to characterize the variable work content of the panels, the results of the simulation model show that the use of averages is not the best way to characterize this work. Figure 4 shows a histogram of the number of longs per panel built on the Panel Line.

Figure 4. Number of Longs per Panel



The average number of longs per panel is 11.6, however, almost 50% of the panels have 13 longs and only 5% of the panels have 12 longs. The additional long requires one extra pass with the stiffener welder. This increases the Span and Process time for the station as seen in Table 4, which causes the blockages shown in Figure 3. The Process Improvement Team understood that the extra longs would increase the station span time but felt that the 35% of the panels which have less than 13 longs, and in some cases as few as 3 longs, would draw the average process time closer to that of a 12 long panel. This was difficult to test without using the computer simulation model. The stiffener welder welds 12 longs in 218 minutes in the "To-Be" process. Since 4 longs are welded in a pass during the 218 minutes the average pass takes approximately 73 minutes. It would take 4 passes to complete a 13 long panel; therefore, the process time would be 292 minutes for this station when the extra pass is made. Table 4, however, shows that the average process time for the stiffener welding station is 236 min. The model indicates that the assumption made by the Process Improvement Team was correct. The average is reduced by the 35% of the panels with less than 13 longs, however, it is not enough to bring the average to that of a 12 long panel.

There is a second factor constraining throughput in the middle of the line. The Average Process time for the fitting work in Station 4 is less than the Average Process time for the welding in Station 5. The imbalance in work content *between* stations causes blockage and idle time, contributing to the higher cycle time of the line. The imbalance is further magnified by the scheduled assembly sequence of the panels. When very low work content panels follow very high work content panels, large blockages in the line are created in the upstream stations. If high work content panels follow low work content panels, idle time develops in the downstream stations. The varied scheduling of panels in such a sequence produces an "accordion effect" where gaps between panels open and close on the line. These gaps increase the cycle time of the line. Therefore, it is not only the variation in work content in each individual station that is important to understand, but the variation in work content and sequencing *between all* of the stations which influences the dynamics of the line. Averages were used by the Process Improvement Team to overcome the difficulties in manually analyzing this variation, but the simulation shows that understanding the variation is a necessary part of understanding the line.

The same situation occurs in the Station 7 and 8 weldout stations. These stations also interact with Station 5 to create blockages and higher span times in almost all of the stations on the line.

Effect of Variation on Scheduling

The variation of work content created problems in the scheduling of panel removals from the line which was realized once computer simulation was used to analyze the "To-Be" process. One of the main objectives in employing a 4 hour line was the ability to be able to schedule the removal of completed panels from the line at 4 hour intervals. Figure 5 shows the time between lift-offs for the variable work content panels.

Figure 5. Time Between Panel Lift-Offs

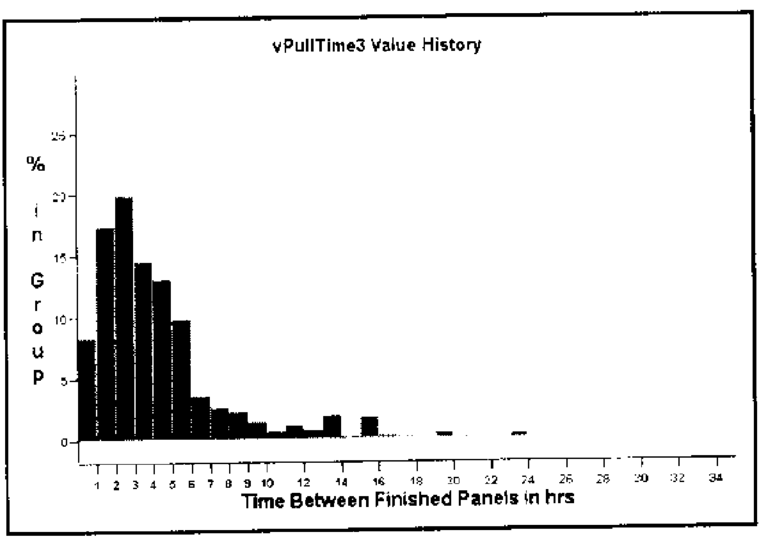


Figure 5 shows that panels are not being completed at regular 4 hour intervals and that the spread in lift off times ranges anywhere from a few minutes to 15 hours. The variation of work content and the removal of panels from four stations on the line make this very hard to analyze manually. The computer model, however, is quite capable of handling this task.

Possible Changes to Panel Line to Achieve 4hr Cycle Time

Once it was determined that the variation in work content resulted in an Average Cycle time per panel of more that 4 hours, four possible options for further reducing the line's cycle time were investigated:

- Change Panel Design/Take Work off Panel Line
- Reduce Process Times
- Add Automation (Robotics)
- Resequence Order of Panels

Change Panel Design/Take Work off Panel Line

Because of the timing of this project, it was decided that the design of the panels currently scheduled to be built on the line could not be changed. Panels for the fourth of a successful seven ship contract were being constructed on the line, so the cost/benefit of utilizing the resources to make the changes to the last three ships was not very good. In addition, there was an understandably high resistance to making changes to the product model when every change meant the potential of jeopardizing an already successful project.

Taking panels that had a higher work content than the reference panel off of the line also was not a feasible option at this time. The panels would have to be built in build locations already heavily utilized to construct other products. The Panel Line is one of the most efficient build locations within the yard for building flat products. A penalty in efficiency occurs by moving the panels from the Panel Line to one of the flat build locations. This penalty is justified as long as the gains in efficiency from moving work off the line offset the efficiency losses due to assembling the panels elsewhere. There is a point, however, when this tradeoff no longer becomes advantageous.

Reduce Process Times

Reducing the process times was a second option investigated by the Process Improvement Team. In order to create the "To-Be" process, however, a substantial amount of time had already been cut out of the existing processes. These cuts in time also came at a substantial cost. In order to assess the additional reduction in process times necessary to achieve the objective of a 4 hour cycle time for panels with variable work content the computer simulation model was again employed.

The process time per plate, deck socket, long, and web for each station was reduced in increments of 5% in order to reduce the Average Process time per panel. While this was occurring, resource and location utilizations were being monitored to determine the balance in per-unit process time reductions that would allow for the best usage of resources and equipment. This is yet another advantage of computer simulation. As changes are made, the effects of those changes on the system are immediately evident. Table 5 shows the additional percent decrease in per-unit process times at every station that would be necessary to achieve the objective of a 4 hour line (with variable work content panels) and utilize the resources in the best way possible.

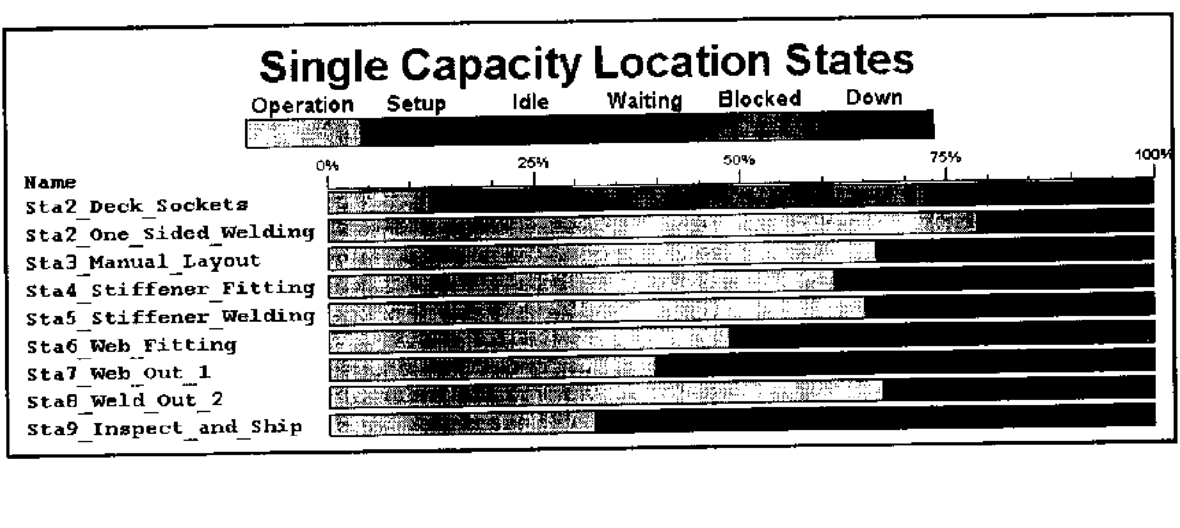
Table 5. Additional Per-Unit Process Time Reductions Required by Station

	Percent Decrease in Process Unit Time
1 - Plate Fitting	0%
1 and 2 - Deck Sockets	0%
2 - One-Sided Welding	0%
3 - Layout	10%
3 - Perimeter Burn	10%
4 - Stiffener Fitting (Press)	20%
4 - Stiffener Fitting (Manual)	20%
5 - Stiffener Welding	30%
6 - Web Fitting	10%
7 - Weld Out 1	10%
8 - Weld Out 2	10%
9 - Inspection and Ship	0%

The per-unit process time would have to be reduced in Stations 3 through 8, and a 30% reduction in per-unit process time is necessary in Station 5. Stiffener welding is a machine-based process. The reduction in time would have to be achieved by speeding up a machine that is already running at its optimum operating level based on the welding parameters of the material. This would be impossible with the existing welder and the material used to fabricate the panels.

Even if the further reductions could be made in the process, the results of the simulation show that the blockages would not be totally eliminated. Figure 6 shows the Location States for the 4 hour variable panel model with reduced times.

Figure 6. Location States for the 4.0 Hour Line with Reduced Process Times



The blockage is less than that shown in Figure 3, but has not been eliminated. The variability in the panels still holds up the line. In order to reduce the process times in each station just to achieve the location states shown in Figure 6, additional manpower and equipment would be necessary. These resources would be unutilized while the line is blocked. A penalty in resource utilization would be paid in order to achieve the 4 hour line.

Add Automation (Robotics)

A third option of using robotics to eliminate the bottlenecks in the line is being investigated as part of another simulation project. The preliminary results of the project indicate that the addition of robotics to assist in the fitting of stiffeners and weldout of webs would provide the same throughput rate as the existing system, but at a substantial savings in labor costs. The benefit in adding the automation, therefore, is not a decrease in panel cycle time over manual panel line processes, but a reduction in manual labor. Proper balancing of the panel work contents between stations would still be necessary in order to achieve the best possible panel cycle time.

Resequence Order of Panels

Resequencing the order of the panels is probably the easiest option to implement, but the most difficult of the four options to simulate. Changing the order of the panels to smooth out the flow of the work might reduce the average per panel cycle time to 4 hours without expensive equipment or process changes, however, many more variables come into play when optimizing the system in the model. Currently the panels are scheduled for the line by the need date of the next assembly. If the panel is late, the start of the next assembly is late. This may cause a ripple effect leading to the late erection of a block. There is some float in this date, but the amount is dependent upon current workload, complexity of the products, resource levels of the yard, and available storage space. An estimate of the amount of float in start and complete dates for the panels is necessary in order to determine the window in which the panel can be moved to accommodate work flow. Because of the dynamics of the variables involved, computer simulation is a necessary tool in the analysis of this window.

The position of the panel on the line over time also plays a role in resequencing the panels. As mentioned earlier, panels are pulled off the line as they are completed in four places. The lift-offs between these four areas need to be coordinated in order to make the best use of line space and schedule. Trying to juggle all of the variables such as work content and process time necessary to balance the line at the take-off points will be difficult even with the use of computer software.

For these reasons, the project team is investigating the use of optimization software to drive the sequencing of the panels. The optimization package included with ProModel is SimRunner. SimRunner takes input constraints from the user and optimizes the model to those constraints through the use of factorial design of experiments and other optimization algorithms. Once the constraints are specified, SimRunner works on its own, reducing the time necessary for the user to perform the experiments. Because of the complex logic involved in determining the possible orders of the panels, some changes to the model will be needed to interface with SimRunner beyond the normal interactions. The changes necessary to perform these interactions and the amount of time required to make the changes are currently being investigated.

Conclusions

As the level that shipyards investigate their processes for possible improvements becomes more detailed, the tools typically used to analyze them become more cumbersome to employ. Assumptions are used to simplify the processes in order to make them more manageable to flowchart and understand. Sometimes the simplification of the processes can inadvertently lead to a loss of detail that is necessary to completely understand the system. Capturing the variation in process times and work content is also very important to creating a clear picture of the manufacturing process. Increases in personal computer processing speed and the development of user-friendly computer simulation software now provide shipyards with an ability to not only include a higher degree of detail in their analysis, but a way to capture the variation in process and work content.

NASSCO's use of computer simulation proved to be a valuable tool in the analysis of the company's Panel Line. First and foremost, it demonstrated the complexity of the dynamic interactions between stations, products, processes, and schedules. In order to simplify the paneling operations for manual process analysis techniques, the Panel Line was viewed as a rather straightforward process. Panels were moved into a station, processed, and then moved to the next station when it was time for the line to move. The stations with the highest span time determined the cycle time of the line, so the focus was to optimize the processes in these stations to meet the performance objectives. When working with averages and constant numbers this happens to be the case. When variation in the process times, products, and pull locations exists, however, the cycle time cannot be obtained using the same methods. Sub-optimizing the individual processes and line stations often can lead to performance degradation of the overall system if it is not known how the individual components interact to affect the system's performance. Simulation software is very capable of keeping track of all of the variables and interactions involved in the overall operation of the line. Variance in the products and process times can be introduced to better understand their role in planning and forecasting the performance of the system, while changes made to the model can be immediately quantified. It is advantages such as these that make computer simulation an important tool in the analysis of complex shipbuilding processes, and it is the understanding of these processes, which will help yards produce cost competitive ships into the next century.

APPLICATION OF ADVANCED SIMULATION MODELING TO SHIPBUILDING - A DEMONSTRATION PROJECT

Louis E. Alfeld, ScD., Decision Dynamics, Inc., Bethesda, MD
J. Chris White, Decision Dynamics, Inc., Bethesda, MD
Colleen S. Pilliod, Decision Dynamics, Inc., Bethesda, MD

Introduction

This paper describes the results of a U.S. Navy (USN) Naval Sea Systems Command (NAVSEA) sponsored project to support a LPD-17 pilot production planning study. Decision Dynamics, Inc.'s (DDI) simulation tool, ShipBuild™, was used to evaluate preliminary production designs by quantifying tradeoffs:

- between cost and schedule,
- among alternative design choices, and
- among alternative production methods.

Results showed that ShipBuild can automatically transform a list of task backlogs and a list of yard resources into a schedule and manning forecast. Furthermore, the program will do it over and over again, in just minutes, helping planners discover the optimal task layout and the most efficient allocation of shipyard resources.

Problem Definition

Planning is the most critical and vexing problem in the shipbuilding process. To be successful, a strategic plan must integrate and manage the multitude of functions that are key to the construction process. Planners must also learn how to minimize the impact of changes and delays and how to quantify their contribution to the total cost of a ship. What, for example, is the best construction sequence for a ship? How can engineers design a ship for affordable construction? How can a shipyard best utilize its resources during the construction process? How can the negative impacts of design changes and delays be minimized?

Resolution Method

Shipbuilding is a very complicated activity, perhaps the most complicated of all construction undertakings. New management tools are needed to help unravel complicated relationships and bring new understanding to the control of complex dynamic processes. ShipBuild fully demonstrates the capability of dynamic simulation modeling by capturing both the essential physical shipbuilding and the management decision-making activities that support the physical production process. The purpose of the model is to provide shipbuilders with a much-needed method for simulating the outcome of multiple "what-if?" scenarios, thereby allowing shipyard managers to determine the best work sequence, the most effective use of manpower and equipment and the least-cost solution to each new ship construction job.

Research Objective

The objective of this project was to demonstrate how a dynamic simulation model could assist LPD-17 management in selecting among competing "what-if?" options to help create the most affordable results. To fulfill this requirement DDI applied its ShipBuild simulation model to LPD-17 production design issues. ShipBuild is capable of capturing the planned sequence of activities required to fabricate, assemble and test a complex product such as the LPD-17. The model is capable of providing a baseline for exploring the cost and schedule impacts of alternative designs, process technologies and management decisions. DDI worked with participants from Avondale Shipyards and the USN design team to develop data for the model and to evaluate the results.

Model Description

Key Features

ShipBuild simulates the dynamics of shipbuilding activities and management policies. The model is capable of simulating both aggregate and detailed elements of the fabrication and assembly process. By employing the same dynamic feedback structure but using different parameter values, the user is able to aggregate the work into larger work packages to simulate bigger assemblies and even an entire ship. ShipBuild also contains a shipyard submodel that provides the user with the ability to modify the parameter values associated with the facilities and labor of any shipyard. ShipBuild gives the model user an unprecedented capability to develop and test alternative "what-if?" scenarios for the purpose of improving both the productivity of ship designs and the efficiency of shipyards. ShipBuild can realistically simulate project performance over time because of several capabilities that are above and beyond the capabilities of traditional project management tools. Two examples are:

- Schedule pressure. Schedule pressure can arise whenever projects or tasks lag behind their planned completion dates, unexpected delays cause missed milestones, or when changes add new tasks to the work scope without compensating increases in the schedule. Even though a project or task is currently on schedule, if its anticipated future rate of accomplishment falls below the planned rate, then its projected completion date may miss the target and schedule pressure can rise. Schedule pressure may also build for a project or task that has not yet begun but which, because it starts late, will likely finish later than planned. Schedule pressure pushes managers to act to get projects and programs back on schedule. Internal feedback mechanisms which mimick management actions can alleviate schedule pressure for a task by increasing the rate of work accomplishment by adding more labor to the task and/or by initiating overtime work. Both actions, however, can result in productivity losses.

- Out-of-sequence impacts. Schedule pressure can also result in out-of-sequence (OOS) work, particularly when a bottleneck occurs or when work is delayed. Working OOS means that a task is working ahead of its progress dependency relative to other tasks. OOS can also lower productivity and generate rework. Often there is a delay between the time when rework is created and the time when the people working on the task recognize the rework and incorporate it into their work plans. Rework also represents the unanticipated additional work generated for a task as a result of poor or imperfect work.

Schedule pressure and OOS can occur whenever work progress deviates from a baseline plan. The unique power of ShipBuild allows managers and planners to analyze the consequences of alternative "what-if?" scenarios. Such scenarios can pose possible delays or work changes, can impose resources constraints, and can alter production plans. The model can also identify the best mix of management strategies and actions to overcome problems.

Data Inputs

ShipBuild simulates project performance based on the following user inputs:

- Product-oriented work breakdown. The Work Breakdown Structure (WBS) represents the hierarchy of work tasks, from large tasks down to smaller and smaller tasks, that make up the larger project. ShipBuild also supports multiple projects simultaneously.

- Dependencies among work tasks. Each individual work task within a project may depend on the status or completion of one or more other individual work tasks. Some tasks may work in parallel, while others must work sequentially so that a later task does not start until the previous task is complete. Still others may start when only a portion of work from another task is complete.

- Resources required for each work task. ShipBuild classifies resources in one of three categories: labor, materials and work stations. Within each category, there can be numerous types. Resources are collected within any number of facilities.

Management Rules

The management functions allow the user to specify how management will respond to various conditions and situations that may occur during the execution of a project. For instance, what actions does management take when a project is behind schedule? When is a project considered "behind schedule"? Examples of management rules include:

- Management perception and action times. These variables indicate how quickly management recognizes and responds to certain conditions and situations on a project. The user can specify the average time it takes to respond to schedule pressure (days), the average time it takes to add rework to backlog (days), the level of schedule pressure that must occur for a task to begin OOS work, and when OOS work stops because of decreasing schedule pressure.

- Indicated schedule pressure. The user can specify how schedule pressure will be measured for an individual task based on the ratio of work remaining to the days remaining.

- Labor multiplier due to schedule pressure. The user can specify how management will increase or decrease the level of labor working on an individual task in response to schedule pressure. Labor may be shifted from less critical tasks to more critical tasks to speed production.

- Productivity loss due to over/under-manning. The user can specify how labor productivity will increase or decrease based on the ratio of actual to desired labor.

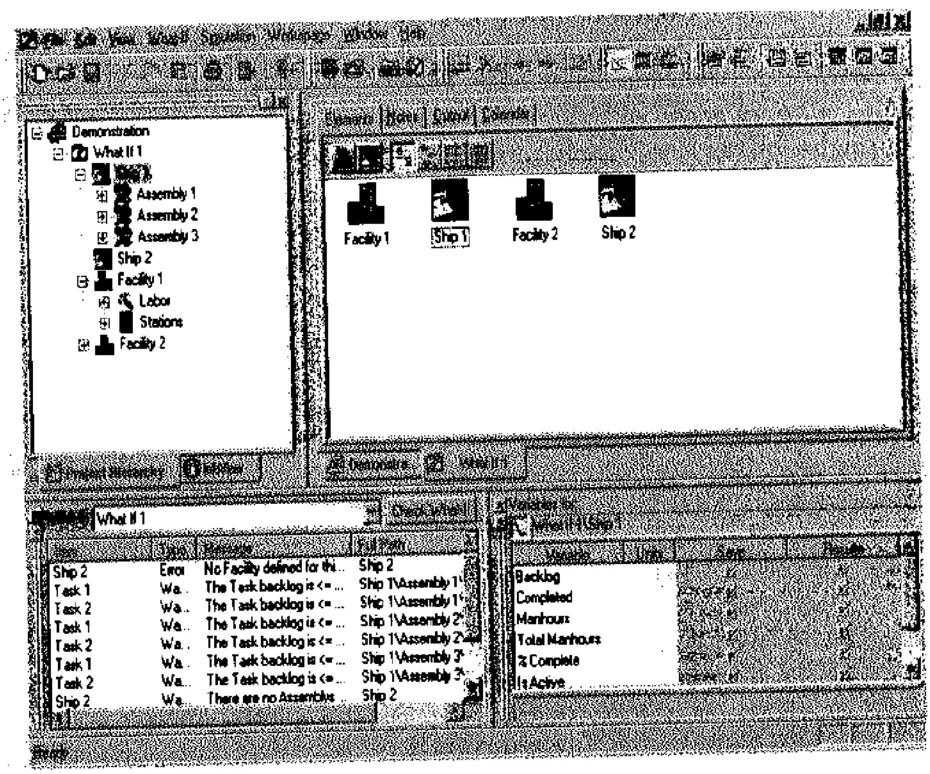
- Overtime worked due to schedule pressure. The user can specify how management will add overtime hours in response schedule pressure.

- Productivity loss due to fatigue. The user can specify how labor productivity is impacted by the level of fatigue currently experienced by the labor when working overtime.

ShipBuild User Interface

The user interface reflects management's need for simple-to-use access to the model's power and versatility. The interface architecture contains four windows (Figure 1). The upper left window is the Project Explorer, the upper right window is the main Workspace, the lower left window is the Check Model, and the lower right window is the Variable Picker.

Figure 1. ShipBuild User Interface



Workspace

The Workspace is the main work area in ShipBuild; it always remains open while a file is open. In the Workspace, the user defines the WBS and, through various icons and tabs, enters all project data as well as the management rules. All data is contained in icons for facilities and ships which are themselves contained in a "What-If?" folder.

Project Explorer

The Project Explorer displays a hierarchical "tree" view of the contents of the Workspace for easier navigation. For instance, the topmost element is a Study. The next level contains various "What-If?" Scenarios defined by the user. Under the "What-If?" Scenarios, the icons show Facilities (with Labor, Material, and Work Stations defined by the user) and Products (with breakdowns of Assemblies and Tasks defined by the user) within the "What-If?" Scenarios.

In the Project Explorer, any element at any level of the tree can be double-clicked and its window will appear in the Workspace. Also, at any time in the Workspace, the user can click on the "Sync to Explorer" icon in the toolbar to highlight the element in the Project Explorer tree that is the current active window.

Check Model

Check Model is used to verify that a "What-If?"? Scenario has all necessary inputs to run a simulation. The user selects the "What-If?" Scenario from a list and then clicks on Check "What-If?" Scenario to have ShipBuild list any data errors and warnings pertaining to the selected "What-If?" Scenario. For both errors and warnings, the user can double-click the error or warning from the list in the Check Model and ShipBuild will open the proper window in the Workspace to simplify data inputs or corrections.

Variable Picker

When the user highlights an element in the Project Explorer, the list of its variables will appear in the Variable Picker. If users wish to plot the simulation results for a variable, they can open either a Time Series Viewer or Gantt Viewer and click-and-drag the desired variable from the Variable Picker onto the viewer window. The Save box must be checked for a variable to have its results saved after simulation. If there is a check in the Results box for a variable, results have already been saved and are available for plotting on a viewer. If there is no check in the Results box for a variable and there is a check in the Save box, a simulation must be run to obtain results for plotting.

LPD-17 Data Collection

The data used to populate ShipBuild for the LPD-17 pilot demonstration project included information on both the resources currently available in the shipyard along with the basic WBS for a preliminary hull design.

Shipyard Resources Data

ShipBuild classifies resources in one of three categories: labor, materials, and workstations. For the LPD-17 modeling effort, data were collected concerning labor and workstations (plattens) but not for materials. For this demonstration of the model, material resources were not considered a constraint in the production process.

The following criteria established the data used to model the labor and workstation resources at the Avondale facility for a Baseline study. When modeling different work schedules, the appropriate changes were made to the availability of labor in different shifts.

1) Shifts
   - Shipfitters and Welders – 2 shifts
   - All other labor trades – 1 shift
   - 8 hour days
   - 5 days a week

2) Labor Trades Available per Shift
   - Welders
   - Shipfitters
   - Laborers
   - Sheetmetal Workers
   - Electricians
   - Machinists
   - Pipefitters
   - Quality Assurance Inspectors
   - Other

3) Workstations – Plattens. Table 1 lists the facilities, number of workstations and associated productivity rates used to develop the Baseline study.

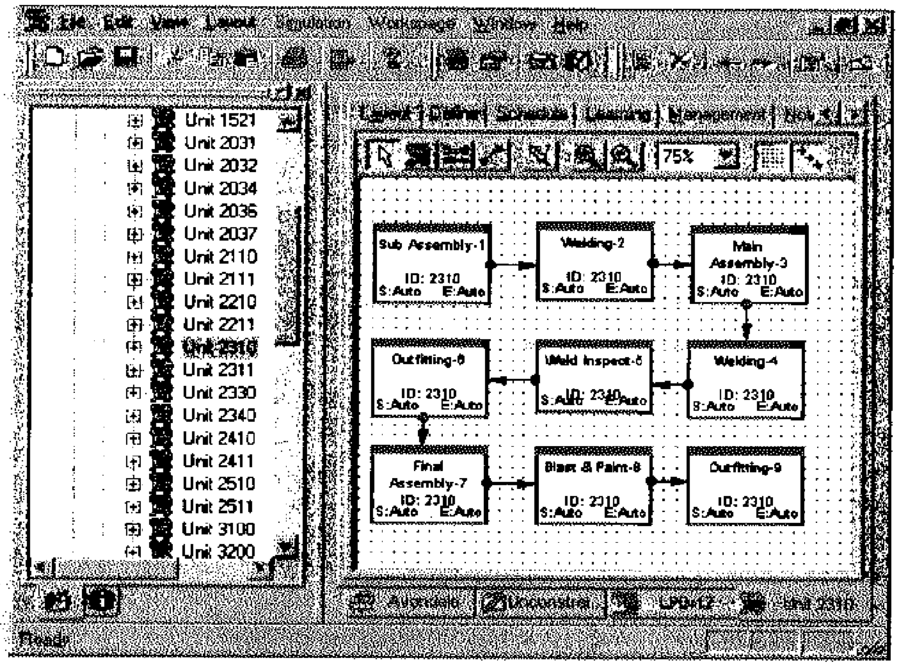
Table 1. Facility Definition

Platten Facility	No. of Grids per Platten	Productivity Rate
Factory 20A	8	1.05
Factory 20B	8	1.05
Factory 20C	10	1.05
Factory 20D	10	1.05
6	3	.95
10	10	1.00
13	20	1.00
14	11	1.00
16	12	1.00
17P	13	1.00
18	6	.85
19	13	1.00
22	6	.85
26	58	1.00
307	16	1.00
AB	100	1.00
BW	8	1.00
UY	100	1.00

Assembly Process

The WBS for the LPD-17 pilot demonstration includes descriptions of each major type of unit that makes up the hull. ShipBuild used second level data that listed the sub assemblies, assemblies and outfitting that make up the unit (Figure 2). The dependent relationships for assemblies and tasks were defined according to the work schedule defined by Avondale's preliminary design plan.

Figure 2. WBS Unit Assembly Example



The lowest level in the WBS hierarchy defines the tasks needed to construct the item. Task data includes:

- manhour estimate for amount of work,
- type of work being performed,
- type of labor and amount (maximum, desired, and minimum) necessary for the task,
- workstation needed, and
- dependent relationships among tasks.

The work type, labor type, workstation and dependent relationship input data were determined from information provided by Avondale. The desired number of laborers (per trade) assigned to each task was based on the number of feet of welding or outfitting and the duration of the task. The craft trade labor was assigned according to the normal number of each trade assigned per unit. In addition, it was assumed that at least one laborer per trade was required to begin each task. The maximum amount of laborers that could be assigned to a unit was based on the physical size of the unit.

Verification and Validation

DDI conducted a wide range of verification tests to ensure the satisfactory performance of features. Validation of the ShipBuild model required passing three tests:

- Model structure – is the model's feedback structure representative of real-world, cause-and-effect relationships?
- Model behavior – is the model's behavior representative of observed real-world system behavior and is the model's response to new inputs robust and reasonable?
- Model data – is the best data used to populate the model?

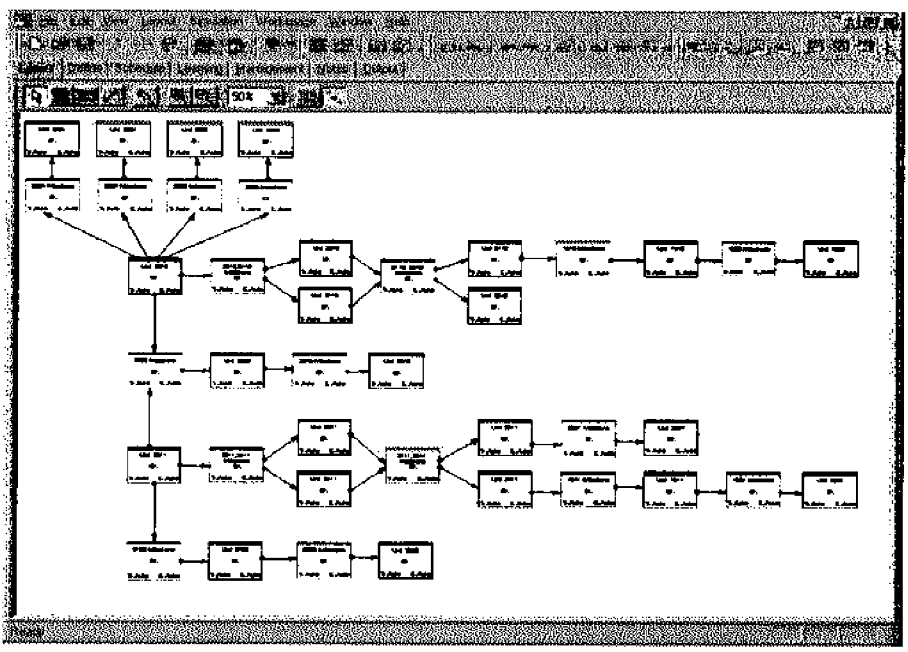
Unit Testing

The initial test cases illustrated that ShipBuild can accurately simulate each major unit type in the preliminary design plan. Each unit type was validated by comparing model simulation results against actual Avondale planned schedule durations. For example, one of the unit's simulation results indicated it would take 27.188 days to complete as compared to an actual schedule time of 27 days. Once each unit simulation was validated, units were duplicated and dependencies modeled according to the overall erection sequence. The scenario for the total ship included more than 200 high level units which contained over 5,000 tasks.

Model Testing

The model testing involved the production of one section of the overall ship. The scenario focused on 23 of the units that make up the lower sections of the ship hull (Figure 3).

Figure 3. Test Scenario Assembly Layout

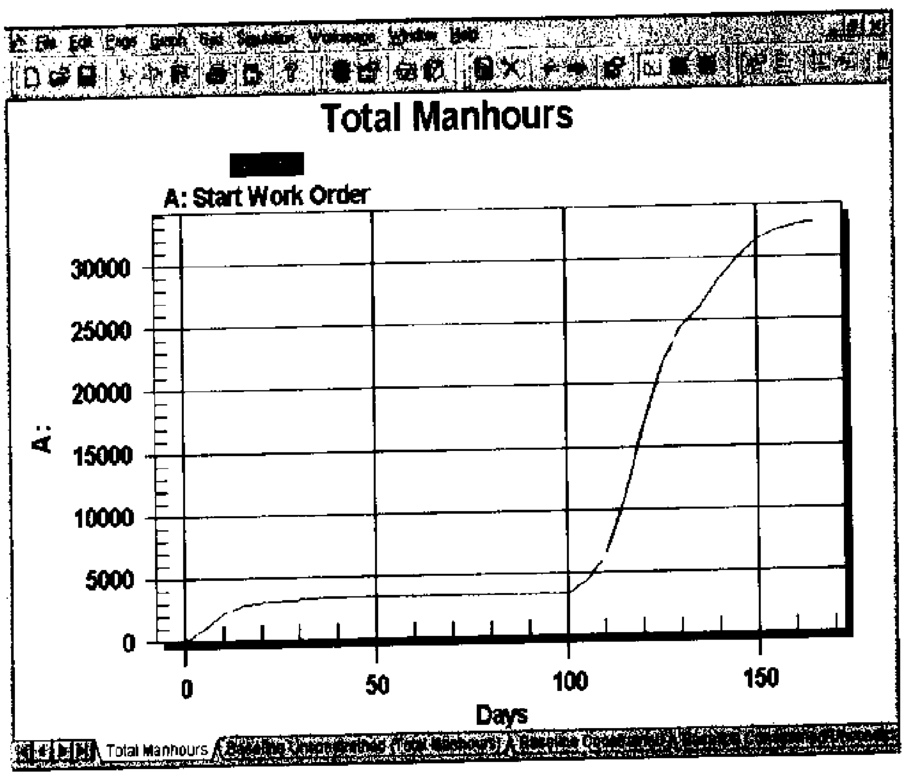


Baseline Validation

The first test re-created Avondale's planned schedule (Figure 4). The assembly units were connected to each other by milestone tasks that contained scheduled start work dates as described in Avondale's preliminary design plan. The scenario simulation results were then compared to Avondale's planned schedule dates. The simulation results corresponded with the actual calendar dates, thereby validating the accuracy of the model. The input data reproduced an accurate representation of the actual construction plan.

Note that the output displayed in Figure 4 indicates an early leveling off of manhours followed by a rapid increase in manhours. This curve does not reflect the total number of planned manhours expended in the shipyard during this time period but reflects only the planned manhours expended for just those blocks used in the Baseline Validation scenario. As the slope of the curve indicates, the majority of the blocks in this scenario are planned to be constructed after the first 100 days of the project.

Figure 4. Test Scenario Output



Model Demonstration

Unconstrained Baseline Scenario – Test Case Description

To test alternative "what-if?" scenarios, DDI used the scenario involving the lower portion of the ship hull construction. The Unconstrained Baseline study developed for comparison eliminated the Avondale start work dates. Instead, the Unconstrained Baseline study assumed that (1) each task could begin when the resources were available; and (2) there was no constraint on Avondale's resource availability. It is important to note for the Unconstrained Baseline scenario, that only welders and shipfitters worked two full shifts while the remaining labor trades only worked the first shift.

Figure 5 displays four results. The first reproduces the Avondale planned work schedule (red). The second (blue) shows the unconstrained baseline, assuming that all needed resources are available. The third (green) utilizes only one shift and the fourth (purple) uses two full shifts.

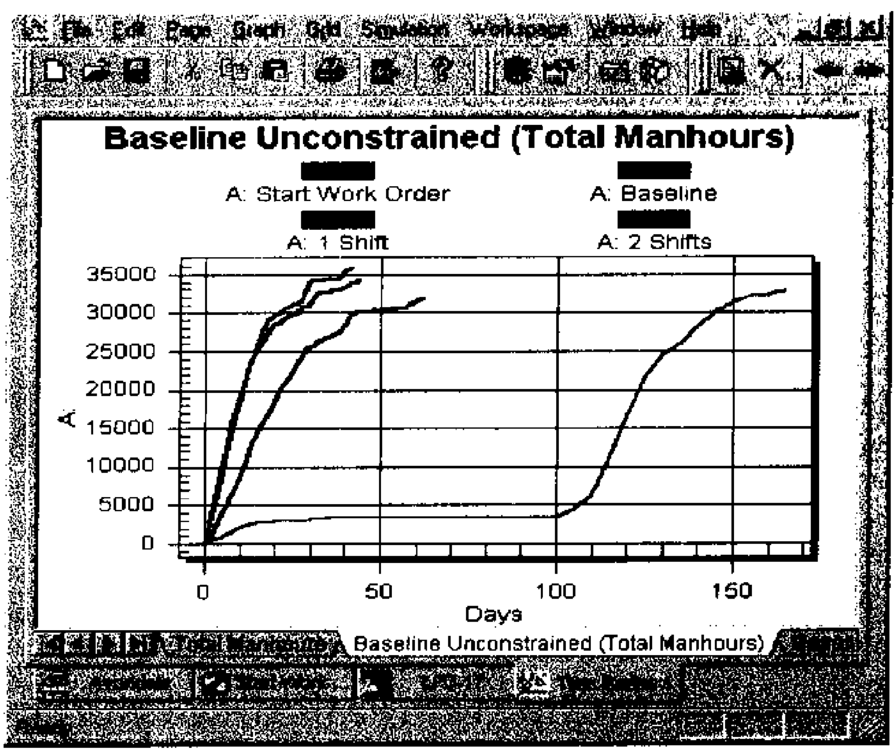
Unconstrained Baseline Scenario – Results

When comparing the simulation outputs plotted in Figure 5, it is obvious that the Unconstrained Baseline scenario finishes much earlier (approximately 45 days to finish) than the Start Work Order scenario (approximately 170 days to finish). This is the expected behavior because the Unconstrained Baseline scenario does not contain the delays imposed by calendar start dates. The tasks in the Unconstrained Baseline scenario begin as soon as resources are available.

By limiting the Unconstrained Baseline scenario to only one shift, the number of days to complete the project increases (approximately 60 days to finish). However, the number of total manhours has realized a small decrease. One of the assumptions in this demonstration study defined a decline in productivity for the second shift of workers. By performing all the tasks in the first work shift, where the productivity is highest, a decrease in total manhours is realized.

The last scenario applied two full shifts to the Unconstrained Baseline scenario. The comparison of these two scenarios indicates a slight increase in total manhours with an even smaller savings in the number of project days to completion.

Figure 5. "What-if?" Baseline Comparison



The model behavior in this "what-if?" comparison reflected what one would expect in a real-world situation. To further validate the model, several additional scenarios were developed and demonstrated to the satisfaction of all participants.

Management Functions Scenario – Test Case Description

A "What-If?" Scenario comparison was used to demonstrate ShipBuild's ability to simulate management's response to schedule pressure. Whenever a change or a delay causes the ShipBuild simulation to deviate from the planned baseline, tasks that are delayed begin to generate schedule pressure. As schedule pressure rises, it can trigger a variety of management actions. (These actions are dependent upon user-controlled settings.) For example, schedule pressure may translate into adding more workers or adding overtime hours. Alternatively, schedule pressure can be ignored in order to forecast what would happen without management intervention.

In this scenario, the availability of a critical workstation used to construct several units was constrained to simulate the impact of parallel work using the same workstation. To compare the impacts of management intervention in the production process, one simulation was run without applying any management functions and one was run assuming management responded to the schedule pressure by adding overtime hours. In the comparison displayed on Figure 6, the total manhours of the Unconstrained Baseline (blue) scenario was compared to both the scenarios with (green) and without (red) management functions applied. Figure 7 displays the manhour distribution for this same comparison scenario.

Figure 6. "What-if?" Comparison with Management Functions (Total Manhours)

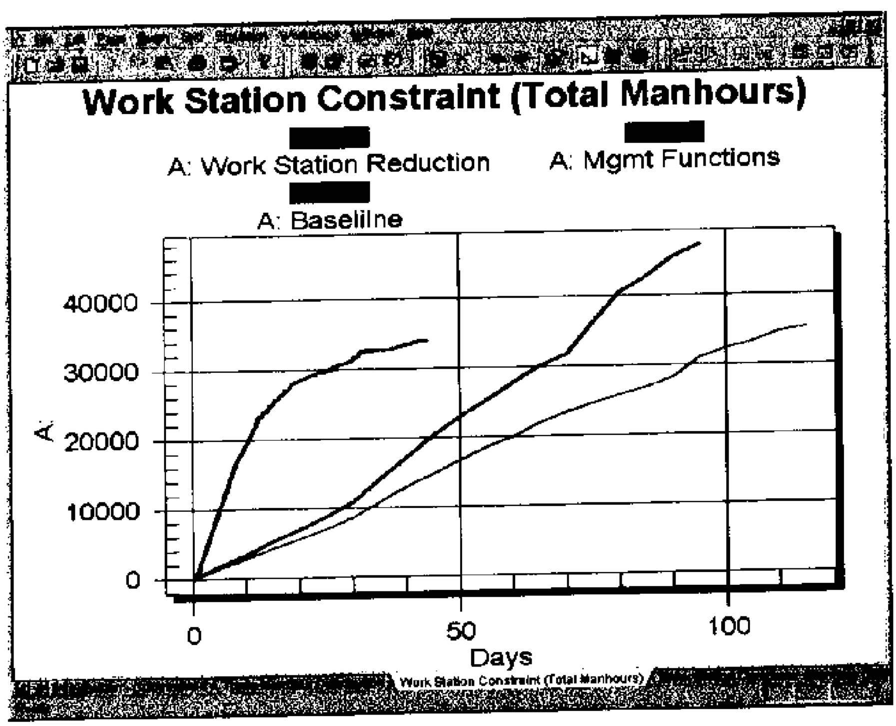
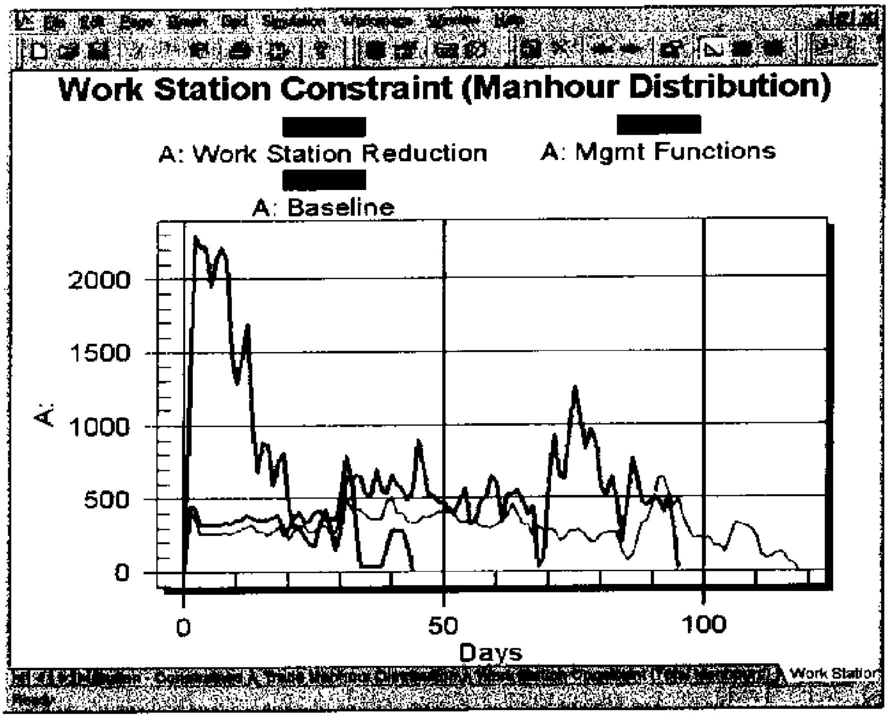


Figure 7. "What-if?" Comparison with Management Functions (Manhour Distribution)



Management Functions Scenario – Results

As expected, the results in Figure 7 show the Unconstrained Baseline scenario (blue) has the highest number of manhours at the beginning of the project and finishes the earliest (approximately 45 days to finish). The Unconstrained Baseline in this instance has sufficient resources in the form of workstations and labor to begin each task as early as possible.

The Workstation Reduction scenario (red) does not reflect the same manhour distribution as the Unconstrained Baseline because of the constraint imposed by the availability of workstations. The management functions options were not applied to this scenario. Therefore, the simulation made no changes to the amount of labor applied for schedule pressure reasons during the course of the simulation. As the results indicate, the manhours are more evenly distributed at a lower level causing a major extension in the number of days to complete the project (approximately 115 days to finish).

The final comparison allowed management to apply overtime hours to the tasks to try to get the project back to the Baseline schedule. As graphs show, in the Management Functions scenario (green), the model increased the number of manhours over the course of the project resulting in a reduction in the number of project days expended (approximately 95 days to finish) in the Workstation Reduction scenario. However, the physical limitations caused by the workstation constraint prevent the project from regaining the Baseline scenario schedule.

Conclusions

It should be clear from the description of these two simple scenarios, that ShipBuild can be used to explore not only real changes and events but also "what-if?" assumptions and actions. By defining a series of "What-If?" Scenarios, a model user in the previous example could continue comparing the relative impact of many different variables on system behavior. For example, alternative ship designs, task sequences, shipyard resources, problem areas and additional management responses could all be quickly tested in a search for the best solution.

ShipBuild provides program managers with the ability to successfully develop a strategic plan by integrating and managing the multitude of functions that are key to the construction process. The results achieved and the output available from simulating with ShipBuild include:

- schedules for all tasks, work packages, blocks;
- overall ship schedule;
- labor manning (by shift and by trade);
- labor hours for all tasks, work packages, blocks; and
- total labor hours for the ship.

Results showed that ShipBuild can automatically transform a list of task backlogs and a list of yard resources into a schedule and manning forecast. Furthermore, the program will do it over and over again, in just minutes, helping planners discover the optimal task layout and the most efficient allocation of shipyard resources.

Use of ShipBuild will assist design engineers and shipyard planners in three important ways:

- Greater flexibility to plan early, often and more effectively, by permitting users to evolve plans that best address anticipated ship and yard conditions and by permitting users to quickly and efficiently replan whenever necessary.
- Greater control over work sequence, task activities and resource allocation to ensure that the most important work gets done first and that manhour cost and schedule tradeoffs are clearly assessed.
- Greater assurance that the plans are correct, that manhour cost and schedule can be safely predicted and that risks are reduced to a minimum.

MODELING OF PRODUCTION SCHEDULING AND DEVELOPMENT OF SHIPYARD SIMULATOR

Kazuhiro Aoyama and Toshiharu Nomoto, The University of Tokyo, Japan
{aoyama,nomoto}@naoe.t.u-tokyo.ac.jp

Kunihiro Hamada, Hiroshima University, Japan
hamada@naoe.hiroshima-u.ac.jp

Shyoji Takechi, Sumitomo Heavy Industries, Japan
Syj_Takechi@shi.co.jp

Abstract

An insufficient production schedule only results in a loss. Hence, the planning the best production schedule has been most significant technical issue for a long time. In shipbuilding industries, the fact is that great experience and intuition of a skilled scheduler are needed to plan the best schedule because of the complicated nature of scheduling problem. A scheduler has to deal with huge amounts of information in a scheduling and consider many kinds of restrictions on production to plan the best schedule. For the purpose of supporting a scheduler properly, either the in-house or the commercial scheduling system has been introduced aggressively. However, most of those systems have not been used effectively because it is hard to generate and input the necessary data (such as work time for all needed tasks) for the scheduling system.

In order to support the production scheduling, which is terms of results in more effective production, this paper discusses the next generation system for the product scheduling. The authors have been studying the Computer Integrated Manufacturing system (CIM) and have implemented the prototype CIM named SODAS (System Of Design and Assembly in Shipbuilding). In this system's concept, the product information is organized from a design stage through a production stage by using the product model.

Due to using the product model effectively, the authors have implemented the production planning support system for the shipyard. The information on production such as operations and resources is organized, and then the hierarchical operation model and hierarchical factory model are defined in this system. In addition, the authors have implemented the shipyard simulator that can simulate the production activity in the shipyard. This simulator has been integrated with the production planning support system. In this simulator, the Petri Nets are introduced to consider the restrictions on production. Since this simulator is implemented on the production scheduling support system, a scheduler can immediately evaluate his/her schedule when he/she finishes scheduling. A scheduler can change the variety of production conditions and evaluate the efficiency of the planned schedule. Furthermore, this paper discusses the effectiveness of a use of a factory simulator.

Introduction

In manufacturing industries, many manufacturing systems have been developed and used to deal with huge information efficiently. In design stages, various design systems such as Computer Aided Design systems (CAD) are developed and used to generate product information. In addition, Computer Aided Engineering systems (CAE) have been used aggressively to evaluate information on products. Recently, information technology gives designers a good design work environment, they can design the best product easier. Since an effective use of a computer is important for production planning, various systems that can suggest useful information on production, have been developed[1,2]. Although those systems are useful for a production planning, it can be understood that they are limited to the generation of information on production schedules. In fact, a professional skill, great experience, and intuition of a scheduler are necessary to evaluate possibility and effectively of the planned schedules. Therefore, if a scheduler should deal with huge information on product planning[3], it will actually be hard to evaluate the best schedule. The authors conjecture that a new generation system is necessary which can support a scheduler more powerfully. This paper discusses the development the factory simulator. This simulator can support a scheduler to plan the best production schedule with the evaluation of the possibility and effectively of the schedule. Since, this gives him/her much information with consideration various restrictions of production resources. This paper's objectives are shown as follows:

- To clarify information that is necessary to simulate production in a shipyard by arrangement activities of production.
- To define functions of production in a factory and to develop a factory simulator by Petri Nets.
- To organize information that is obtained by the simulation and to evaluate the effectiveness of the factory simulator in the production planning.

Factory and Production

Characteristics in Shipbuilding Production

A shipyard is a class of job shop in which a production route is decided to an individual product. From viewpoints of "Products Flow" and "Job Flow", the characteristics in shipbuilding production can be classified (Fig.1).

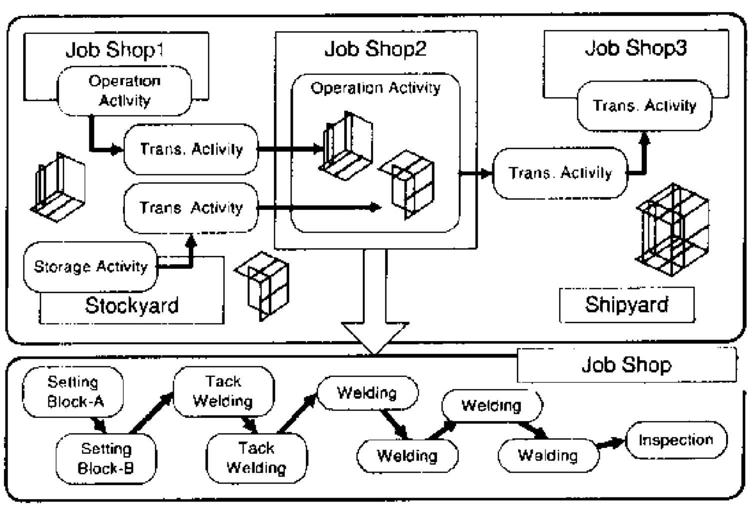


Figure 1 Products Flow and Job Flow

1. Products Flow: In a shipyard, there are various production stages, which are arranged according to the production purpose, and a ship is built by combining various production stages. In those production stages, the activities change the form of material and products to final products. This paper calls this activity as "Operation Activity".

Sub-assembled products have to be transported to a next production stage smoothly in order to assemble with others. Furthermore, they sometimes are stored in a stockyard. In this paper, those activities (transportation and storage) are called "Transportation Activity" and "Storage Activity" respectively. The "Products Flow" can be recognized between the production stages when a shipbuilding production is considered as a combination of activities of "Operation Activity", "Transportation Activity", and "Storage Activity".

2. Job Flow: Each "Activity" has some actual jobs. For instance, "Operation Activity" has some actual welding jobs. Since each job has job order, the connection (flow) of jobs in a production can be considered "Job Flow".

Restriction and Constraint on Production

Prior to an actual production, restriction and constraint on production resources have to be considered as much as possible in a production planning. They are "Restriction by the Ability" and "Constraint in the Time".

1. Restriction by the Ability: At dispatching the job to a production resource, it is necessary to judge whether job is possible or not by a selected production resource. A physical restriction on a production resource relates to this judgment. This paper defines this restriction as "Restriction by the Ability" and considers this restriction as an absolute requirement to accomplish the flow of products and job.

2. Constraint in the Time: Each schedule of production resources is not independent. Therefore, a partial change of one schedule affects other schedules. Assuming this relation of schedules is a kind of restriction, this restriction is defined as "Constraint in the Time" in this paper. This influences the efficiency of "Products Flow" and "Job Flow".

Characteristics in Production Planning

A detailed content and schedule of a job are planed by a production planning. In general, a production planning is classified into "Process Planning" and "Scheduling" (Fig.2).

1. Process Planning: In a process planning, the production scenario is decided with consideration the ability of productive resources. This planning is classified into "Process Design" and "Job Design". "Process Design" is a planning for sequence of whole production activities and making "Products Flow" clear. "Job Flow" is a planning for a production scenario by extracting jobs and making "Job Flow" clear. In a process planning, jobs and their relations are planned with consideration "Restriction by the Ability" of production resources. Moreover, the sequence of jobs is planned. A characteristic of a process planning is that it plans for an individual product.

2. Scheduling: In a scheduling, production schedule is generated from the production scenario with consideration the efficiency of working time, workplace, and working facility. The characteristic in a scheduling is to make schedules of each factory. Since each job is not independent, it is necessary to consider the context of assembly jobs (for example, sequence of assembly). From the viewpoint of production resources, it can be understood that the schedule of a production resource is planned by a

scheduling.

Problem in Production Planning

A Gantt chart generally shows job schedules generated by a production planning, and the impossibility and uselessness of schedules are checked on a Gantt chart. Moreover, a Load chart shows a distribution of load of jobs for particular productive resources whose schedule are important to be control. A scheduler corrects an efficient plan with considering of an overall balance of the performance by finding the problem of the plan.

However, when the planning becomes large-scale, more time and power are needed to take account of "Constraint in the Time". As a result, hence an increase in loads of a scheduler for production planning is not avoided, it will be hard to make the best plan. It can be expected that the best production schedule will be planned, if "Constraint in the Time" of production resources could be considered easily.

Requirement for Factory Model

A job shop has an advantage to produce multiple types of products. However, if a bad production route is planned, many production resources waste time and productive efficiency become worse. Therefore, it is necessary to consider how to keep "Products Flow" and "Job Flow" more efficiently, and it is important to know "State of delay of the Products and Jobs" easily. In order to develop the system that can support production planning, it is necessary to organize information on "Products Flow" and "Job Flow" in the production process. A couple of items are required for factory model. One is that the products can flow in the factory, and another is that the worker can move job stages and work.

In order to model a factory, some abstract resources for activity have been defined, which produce the above-mentioned activities (activities in the Products Flow) (Fig.3). These resources are called as "Process Resource". In this paper, three classes of "Process Resource" are introduced. First one is Operation Resource for Operation Activity such as a workplace. Next one is Transporting Resource such as transporting equipment (including trailer, crane, and etc.). Last one is Storage Resource such as a stockyard. In a process resource, there are actual production resources like

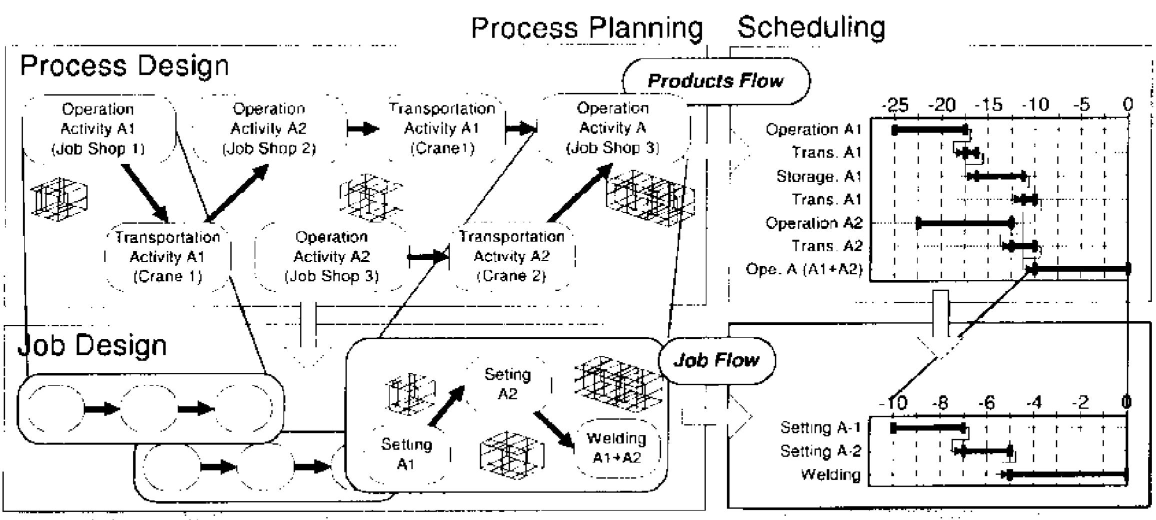


Figure 2 Process Planning and Scheduling

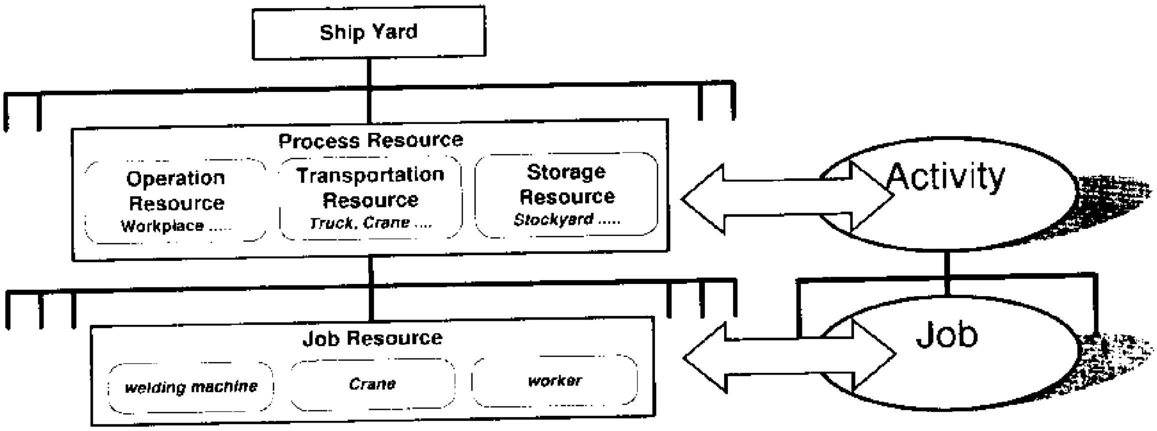


Figure 3 The Hierarchy of Factory Model in Shipyard

"Operation Activity" has some actual jobs. For instance, they are welding machine, crane, worker and etc. In this paper, these are called "Job Resource". By describing productivity in "Job Resource", production time can be estimated easily.

Representation of Factory by Petri Nets

Elements of Petri Nets

In a job shop, event of a production facility is concurrency and synchronization. A Petri net is proposed as a tool which can model such event[4,5,6] and is a directed graph that consists of Transition, Place, Arc, and Token (Fig.4). The event is illustrated by Transition, and the condition and the state of event is illustrated by Place. Arc connects Transition with Place, and then Token flows in Petri net.

Activity and Resource

1. Unit of Activity: The production activities in a shipyard can be classified as follows:
   - Operation Activity: The operation activity is the main activity in which the form of the product changes in production stage (Processing Stage, Assembling Stage, Erection Stage, and etc.).
   - Transportation Activity: The transportation activity is the activity in which the position of the product changes. Products are transported between various production stages.
   - Storage Activity: The storage activity is the activity which controls the progress in the time of the product. This is used intentionally to make "Products Flow" better when there is job waiting in a next production stage. Although this is useful for an effective production, this activity should be as few as possible.

2. Process Resource: This paper defines some abstract resources for activity, which produce the above-mentioned activities, as the factory model. These resources are called as "Process Resource".
   - Operation Resource: Workplace in which Operation Activities are produced.
   - Transportation Resource: Transportation Facility (including Truck, Crane, and etc.)
   - Storage Resource: Stockyard (for temporary stocking parts and sub-assembled modules)

3. Job Resource: In the process resource, there are actual production resources. For instance, they are welding machine, crane, worker and etc.. In this paper, these are called "Job Resource". The "Restriction by the Ability" is described in this resource. Moreover, due to describing productivity in "Job Resource", production time can be estimated.

Basic Representation by Petri Nets

1. Event by Transition: There are two events in production activities. One is "Event in the activity start" and another is "Event in the activity end". Each event is modeled and represented by Transition (Fig.5(a)).

2. State by Place: "The state which waits for the activity", "The state for activity", and "The state that activity ended" are generated by the event. The state of the activity, which exists between events, is modeled by Place (Fig.5(a)). Moreover, the condition of an event is modeled by Place also. Fig.5(b) illustrates the connection of Transition with Place.

3. The progress in the time: In the simulation of the production, it is necessary to consider the progress in the time of the activity. Hence, "Time Petri net" is introduced. A Place modeled as "The state for activity" holds the token until specified time passes.

Production Target and Job Resource

1. Representation of Production Target: Since a product is modeled as a token, "Products Flow" is represented by a token's movement. Information on the product model of parts, sub-assembled modules, and etc. are linked with a token. A token moves from one Place to another. Since the Place is modeled as the state of an activity, a scheduler can know the state of an activity by a position of a token. In addition, the history of activities can be acquired by collecting the histories of the token movements.

2. Representation of Job Resource: The job resources can be assumed the condition associated with the event by modeling the job resource as a token. Information on a worker and a production facility is

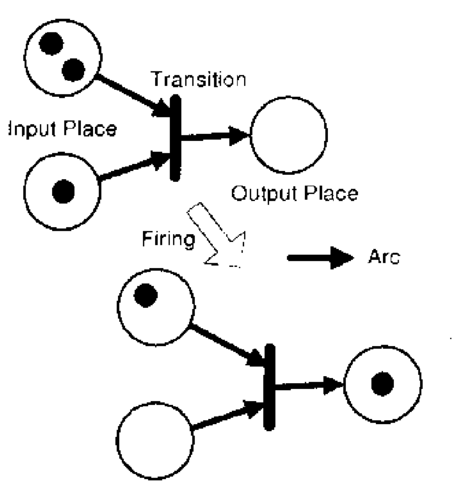


Figure 4 Petri Nets (Place, Transition, Arc, Token)

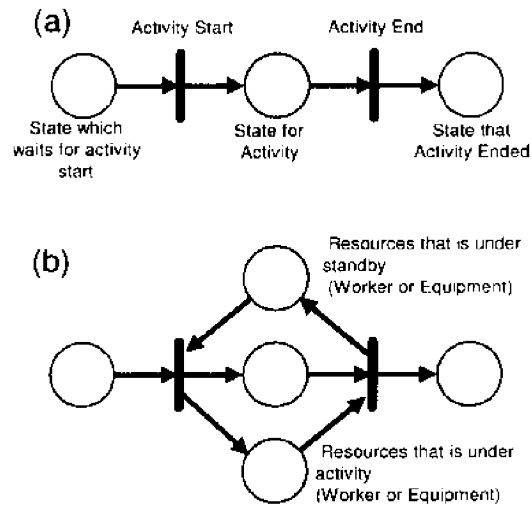


Figure 5 Event, State, and Condition of Job Shop

individually described in a token of the job resource. Moreover, the state of a job resource can be represented by preparing place, which is modeled the state such as "Under activity" and "Under standby". In addition, the working histories of workers and production resources can be acquired by collecting histories of the token's existence in the place.

Representation of Production by Petri Nets

A production process is represented as follows by using basic representation of a Petri net (Fig.6).

1. Representation of Operation Activity: This activity can be produced when the products are carried in the workplace. "Products Flow" between the workplaces is expressed by Transaction firings of "Transporting in" and "Transporting out" of the workplace. It is assumed that there is no "Products Flow" in a workplace to simplify the problem in this research.

- Transition: Events for Transporting in, Activity start, Activity end, and Transporting out.
- Place: States of Activity (Under standby for Operation Activity, Under Operation Activity, Under standby for Transportation Activity) and Job (Under Job, Under standby for Job).
- Token: Products, Workers, and Production facilities

2. Representation of Transportation Activity: This activity is for calling a product in and out workplaces or stockyards. This activity begins by setting a product on a transportation facility.

- Transition: Events for Setting on, Transportation start, Under transportation, Transportation end, and Releasing.
- Place: States of Activity (Under standby for transportation, Under transportation, Under standby for release) and Job (Under job, Under standby for job).
- Token: Products, Workers, and Transportation facilities (crane and transportation truck)

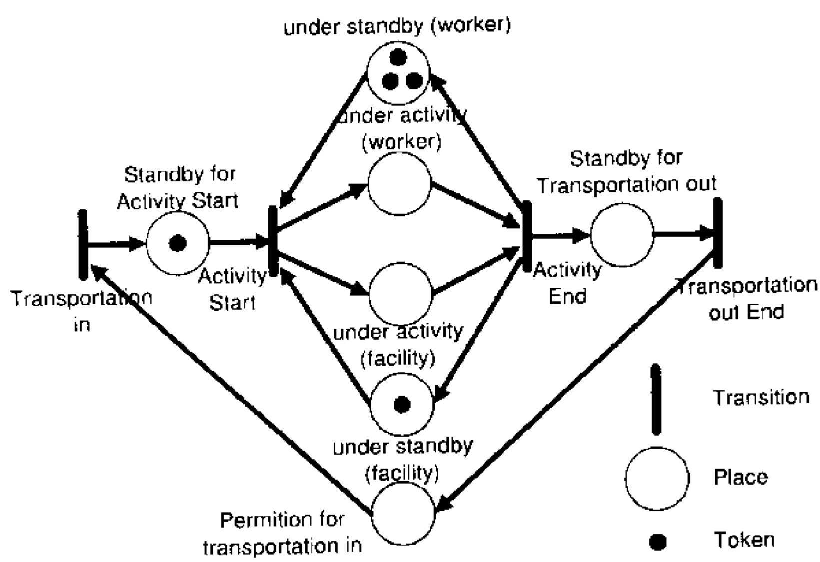


Figure 6 Petri Net Simulation Model

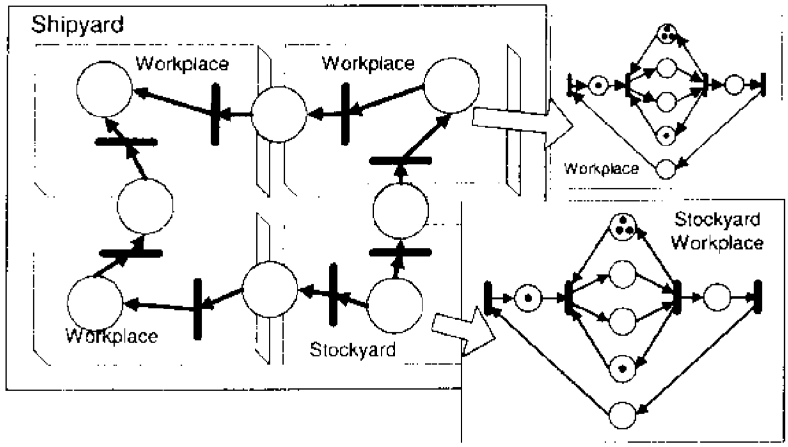


Figure 7 Hierarchy of Petri net Simulation Model

3. Representation of Storage Activity: This activity is for keeping a product in storage space such as a stockyard intentionally to produce an efficient production activity. This activity is modeled by the Petri net same as Operation and Transportation activity with consideration the limited storage space.

- Transition: Events for Transporting in, Storage start, Storage end
- Place: States of Activity (Keeping, Under standby for transporting out)
- Token : Products

4. Estimation the Production Time: The functions for controlling production time is added to the places of Operation and Transportation Activity,. These Places are represented the state of "Under Activity". Production time is estimated as follows.

- Operation activity: The production time is estimated by the combination of the productivity of the production resources (workers and facilities) and products attributes (position and length of connect line, weight of products, and etc.). These attributes can be obtained from information on a product linked with a token.
- Transportation Activity: The transportation time is estimated due to the combination of the transportation ability of the facilities and products attributes (weight, size and etc. of products). The distance between the workplace is considered also.

Hierarchical Representation of Petri Nets

A Petri net can be grouped in one Place by Petri net's hierarchical structure. The grouped Places are connected by Transition. In the Petri net expressed hierarchically, an upper layer represents "Products Flow" in the factory and the lower layer represents "Job Flow" in the workplace (Fig.7).

Preparation for Simulation

Prototype system for CIM in shipbuilding

The authors have been developing the prototype system for CIM in shipbuilding[7-9]. The system is named SODAS (System Of Design and Assembling for Shipbuilding) and Fig.8 shows its overview.

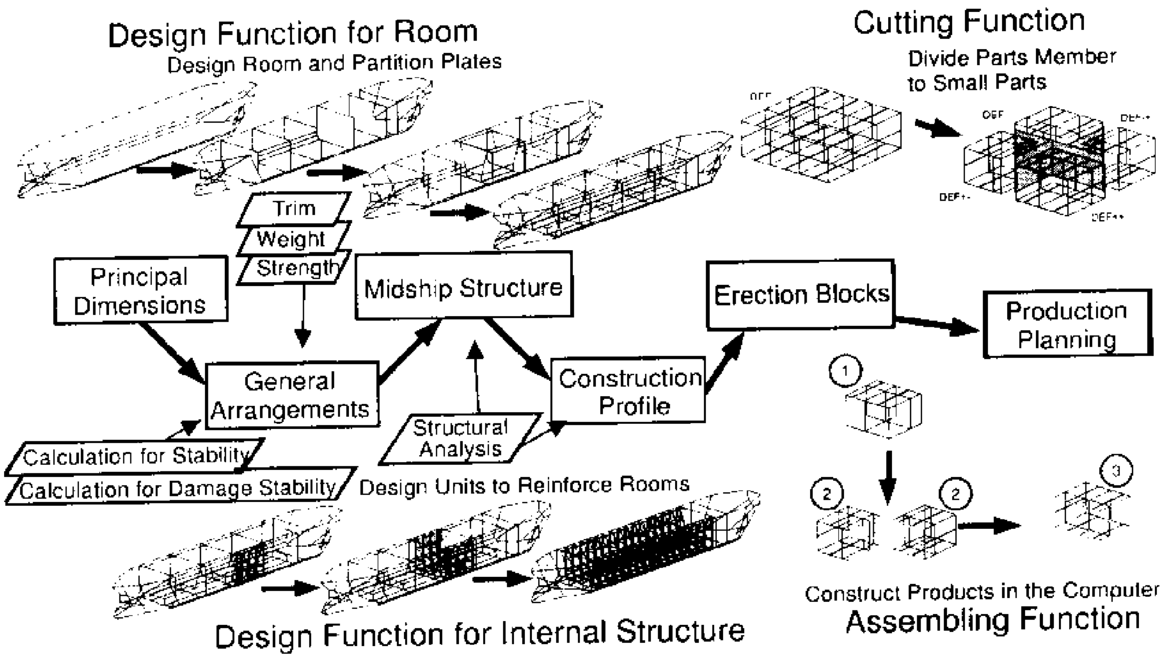


Figure 8 Overview of SODAS (System Of Design and Assembly for Shipbuilding)

In the SODAS, three functions, "Design Function", "Cutting Function", and "Virtual Assembling Function" are proposed to generate and store information on a product as the Product Model.

1. Design Function: "Design Function for Space(Room)" is function for designing the compartments and various tanks of a ship. In this function, the closed space is called as room. "Design Function for Internal Structure" is function for designing internal structures such as the longitudinal members, floor members, and transverse ring members to reinforce the ship hull structures. The concept of a parametric design is introduced for this function and it is considered to design the complicated structure easily.

2. Cutting Function: This function is for designing the actual parts shape and decomposing the designed ship hull structure into erection modules or sub-modules. A parts member is pick up and divided into a lot of small parts. A new connect line is generated automatically at the cutting line where is a divided parts member. Typically, this function can be used for dividing a hull structure.

3. Assembling Function: This function is useful for planning of the assembly process. In order to generate the production information, the assembly order are input by picking out each parts member from a module that is designed by the Cutting Function.

Extractions of Products and Selections of the Assembly Order

In SODAS, the assembly module has been defined as an unfinished product by using the Cutting Function and the Assembling Function. Moreover, the Operation Model, which manages the information on contents of job, has been defined as Operation Object in SODAS. Therefore a scheduler (system user) can store and use the information on an operation easily.

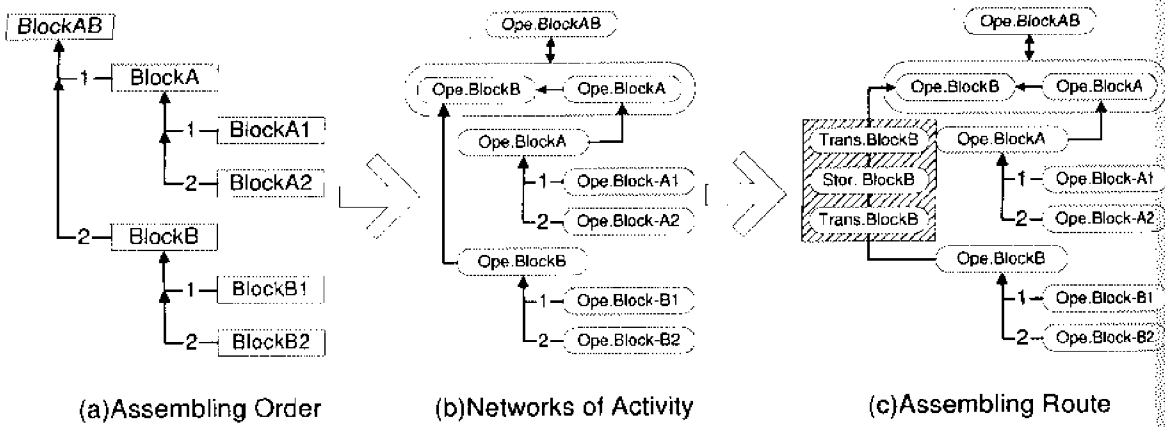


Figure 9 Generation of Operation Model

1. Extraction of Assembly Module: By using the block dividing functions in SODAS, the assembly modules can be extracted from the ship structure that is designed as a final product. The production time can be estimated easily by the combination production attributes (weight of the product, length of connect line, and etc.) and productivity of production resources.

2. Selection of the Assembling Order: After taking account of assembly relation between product in the pre-production stage and the product in the post-production stage, the assembly order can be taken out (Fig.9(a)). The assembly order is input, the graph of the operation model then is generated automatically (Fig.9(b)). Each node illustrates an operation model, and an assembly module is linked with each operation model.

Determination of Assembly Route (Production Route)

1. Selecting Production Resource: The possibility of assembling is judged by a combination the production attributes and ability described in the workplace. After selecting the workplace, a selected workplace is linked with an operation model.

2. Selecting Transportation Route: The operation model of the pre-assembling for a module has already been generated. The assembling sequence between the workplaces is related by describing a workplace in an operation model. If a transportation of product is necessary, an operation model for transportation activity is generated and it is inserted between operation models for operation activity (Fig.9(c)). The transporting equipment is linked with the operation model for transportation activity.

3. Selecting Stockyard: In order to store the product in the stockyard intentionally, the operation model for storage activity is generated between operation models such as an operation model for transporting (Fig.9(c)). The selected stockyard is described in this generated operation model.

By the above-mentioned processing, all operation models are illustrated as a digraph. Moreover, a selected production resource is linked with an operation model and the relation between an operation model and a production resource is generated. By this relation, a production resource and an operation model can refer each other. Moreover, products know production routes by networks of operation models.

Inverse Simulation Function

The assembly order generated by the process planning must be the relative order for all products. However, it is difficult to determine the relative order with consideration to assemble simultaneously. A scheduler should determine a transporting order of products in a workplace with consideration to assembling priority and so on. The transporting order of one workplace affects a operation activity of the pre and post workplaces. Therefore, the assembling procedure that is not determined in the process planning should be determined with consideration of a whole factory schedule.

In this paper, the inverse simulation is proposed to support the determination of the assembling procedure, which is difficult to determinate by process planning. The inverse simulation starts from the ship (a final product) and proceeds decomposing a product into a plate steel material. This inverse simulation is produced by modification the factory simulator that is implemented in this research. The direction of the arc that connect Transition and Place is set in the opposite direction. The order of the assembling module can be obtained by arrangement the work order of the inverse simulation. This inverse simulation function can be used as dispatching of the assembly task. The assembling order is generated in each workplace by an inverse simulation. As a result, the simulation can be applied multiple ships.

Function for Simulation

In order to plan an better schedule by trials and errors, this simulator has some functions, which can show the results of simulation and change various conditions in simulation, are defined as follows.

Information Acquired by Simulation

1. Visualization "Products Flow": This simulator can show "Products Flow" and "Job Flow" visually. Products are displayed as moving nodes those are colored by the state of work. A scheduler can confirm positions and states of products by these displayed nodes.

2. Operation Ratio: Even though a product is allocated in the workplace or transporting equipment, there is a state that work is not produced. This delay of a job is generated by waiting for worker, equipment, and products. An actual rate of production time (eliminated the delay) of each product can be displayed in the graph by schedules, which are managed in a process resource (Fig.10).

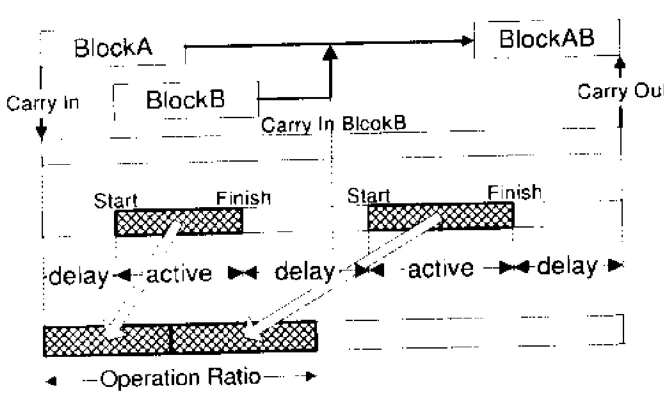


Figure 10 Delay in Job Flow

Function for Modification of Simulation Conditions

1. Modification of Process Planning: A scheduler can change the priority order to assemble a product at a workplace. There is a possibility that transporting in and out order to a workplace changes a schedule of the factory. And a scheduler can change a production route of a product by trying to allocate a product in a different production stage. It means that this function changes "Products Flow".

2. Modification of Factory Model: A scheduler can change number and ability of a worker, production equipment, workplace and stockyard. A layout of job shop is available for changing.

3. Modification of Product Model: A scheduler can change a schedule directly by handling the product which flows in the factory. This change can be easily done by integration with design support system based on the product model, hull structure is then available for changing. A scheduler can change the positions of the block division and the assembling procedure. And more he/she can change allocated production resource and production route.

Partial Execution of Simulation

This simulator can start to simulate partially at any production stages. Therefore, the influence caused by the breakdown of equipment, the turbulence of the weather, and etc. can be easily shown in a short time and the plan can be regenerated.

Simulation Example by Factory Simulator

Design of Factory

The factory model can be designed freely with the graphic user interface. As for the process resource of the designed factory, the Petri net of the process resource is automatically generated. An individual Petri net are connected by the connection between the process resources. Productivity is described in the designed workplace, transportation equipment, and stockyard. The restriction on production resource is represented also.

Plan of Process Planning (Preparation for Simulation)

This simulator is carried out after the process planning. The procedure of the process planning is shown as follows.

- The product model of the hull structure is generated with the design support system and undergoes the block division.
- The block division extracts the products (such as erection blocks, panels, parts, and etc.). The products are defined as assembly modules.
- A scheduler inputs the assembling procedure and selects workplace by the production support system. S/He inputs then the required number of workers and equipment for producing each job.
- The Operation Models are generated. The information on the product and production resource is linked with them. These model are defined as Operation Object in this system.
- The operation objects are illustrated as a node and joined each other by arc.

Procedure and Example of Simulation

The example of the simulation is shown below. The defined shipyard is shown in Fig.11.

Furthermore the designed ship is divided into several modules and the process planning are planned as shown in Fig.12.

1. Generation of schedule information: The schedule information (shown in Fig.13) is generated by the scheduling support system that has been developed in SODAS by the authors[10]. However in this schedule, the restriction on production resource is not considered at all, because only the procedure of work is considered. Therefore, the schedule allows one production resource produces several jobs at the same time.

2. Execution of simulation: The schedule shown in Fig.14 is generated after the factory simulation. In this schedule, restriction on resources is considered and the schedule can be shown in which the production resource does not produce several jobs at the same time. Animation is displayed to "Products Flow" in the factory, and moreover, the bad flow of "Products Flow" and the problem of planned schedule can be confirmed easily.

Modify the Plan and Simulation

A scheduler can judge "Which production resource is bottleneck?" from his/her simulation result. A scheduler can change the simulation conditions by using several functions, simulate again, and then get another result.

- To increase the number of the transportation facilities from two to five, which exists in the route of material stockyard to the assemble place, and allocate the transported product.
- To find time zone with a lot of workplaces that work is not executed referring to the Gantt chart of a workplace and the work efficiency chart of a workplace and allocate jobs of this place.

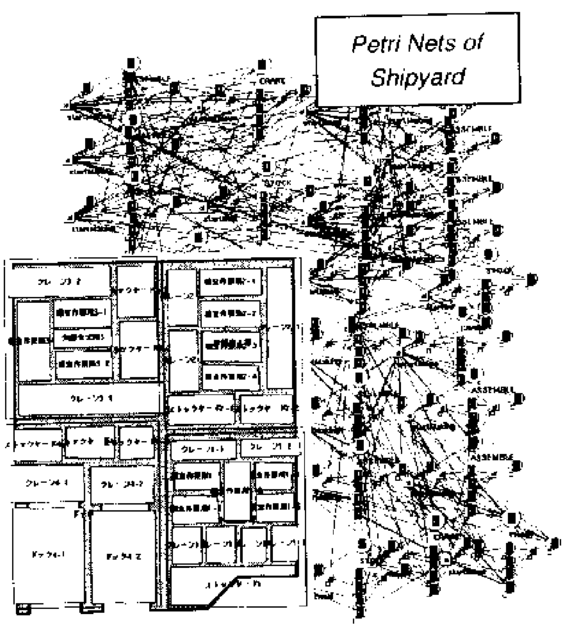


Figure 11 Petri Nets for Simulation of Shipyard

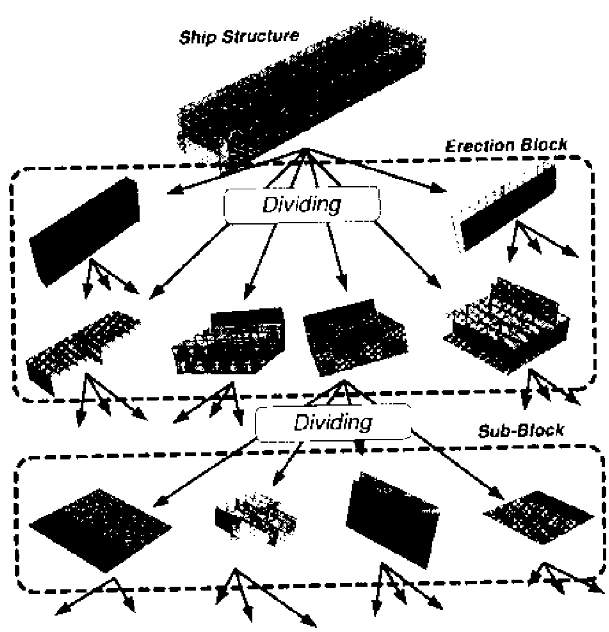


Figure 12 Hierarchy of Dividing Blocks for Simulation

It can be understood that work is dispersed uniformity from the chart of loads shown in Fig.14(B) by an above-mentioned changing plan. Moreover, it can be checked that the flows of transporting in and out the block to the workplace become smoothly and operating ratio of the work place become higher from the chart of Fig.14(C) by increasing the number of the transportation equipment.

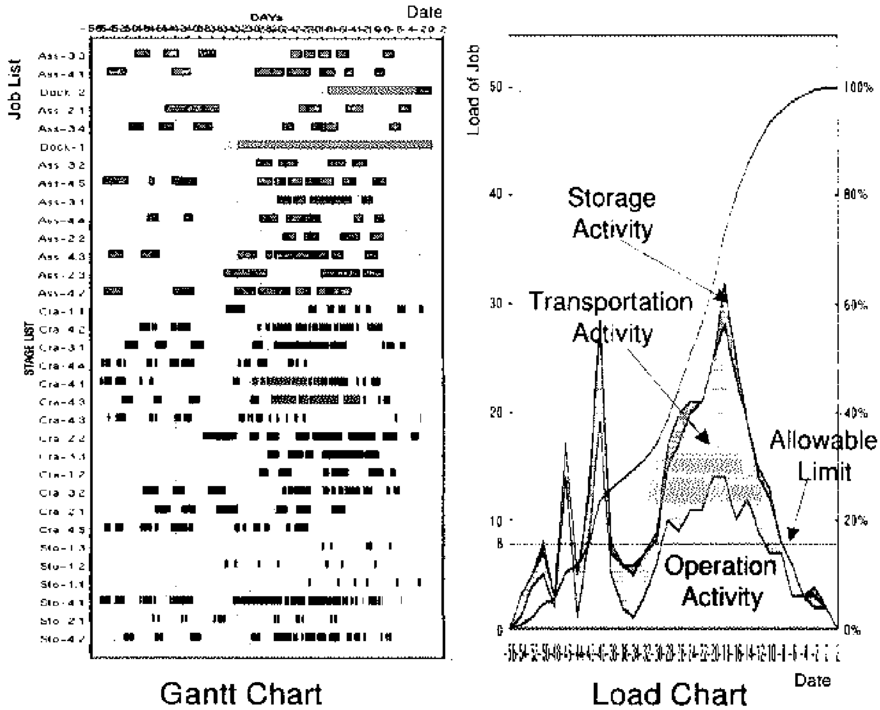


Figure 13 Schedule Data without Restriction

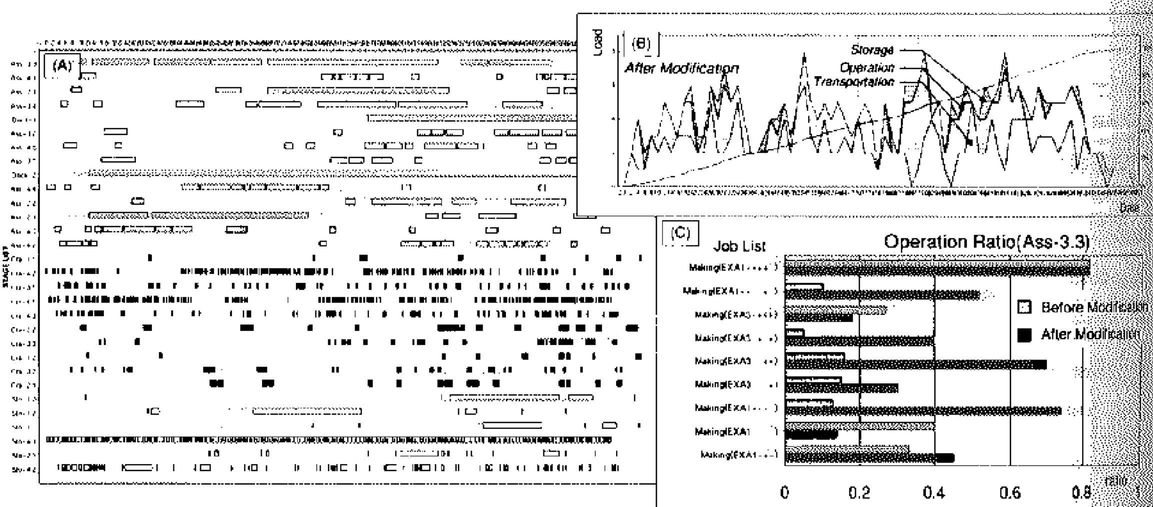


Figure 14 Results of Simulation

Conclusion

This paper discussed the organization of the activity and resource in shipyard and proposed a simulation system for shipyard by Petri Nets.

In order to implement the shipyard simulator, the basic production function of the shipyard and production resources have been organized. In this paper, three kinds of activity are introduced. They are "Operation Activity", "Transportation Activity" and "Storage Activity". The production resource, which corresponded to those activities, was defined and modeled by a petri net.

The effectiveness of petri net could be confirmed. The petri net is useful for modeling the restriction on a production resource. Furthermore, due to describing the ability of productive resources on the model of a productive resource, production time could be estimated easily, so that this simulator could generate schedule.

The production information has been organized to simulate the processes in the factory and arranged information that is obtained by simulation. In order to plan the best production schedule, it is necessary to modify easily a product, production route, and factory. This paper presented that the Integration of a product model and a factory model are very important to carry out these modifications.

References

1. Hitomi K (1990), Manufacturing Systems Engineering, Kyouritsu Syuppan
2. Sasaki F (1984), Scheduling and Process management system. The system technology in shipbuilding. Society of Naval Architects of Japan, pp.157-168
3. Minemura T (1993), A study on the expression of information on schedule for shipbuilding industry CIM. Journal of Society of Naval Architects of Japan Vol.174, pp.673-681
4. Aoyama M, Uchiyama N, and Hiraishi K (1995), Theory of Petri net, System control information library, Asakura Shyoten
5. C.Petri (1962), Kommunikation mit Automaton. Ph.D thesis, Universitat Bonn, Bonn, West Germany
6. Yoshikawa T, Son M, and Sekine K (1992), Production simulator system by Smalltalk-80, Proceedings of 2nd Smalltalk symposium, pp.223-230
7. Aoyama K, Hamada K, and Nomoto T (1997), Basic Studies on Design Supporting System for Offshore Structure, 9th International Conference on Computer Applications in Shipbuilding, pp.143-157
8. Aoyama K, Takechi S, and Nomoto T (1997), Basic Studies on Accuracy Management System for Shipbuilding, 9th International Conference on Computer Applications in Shipbuilding, pp.323-338
9. Aoyama K and Nomoto T (1997), Information Models and Functions for CIM in Shipbuilding, Journal of Marine Science and Technology, Vol.2 pp.148-162
10. Aoyama K (1994), A Study on Modeling of Production Planning based on Product Modeling System, Journal of The Society of Naval Architects of Japan, No.176, pp.563-574

MATERIAL MANAGEMENT IN SHIPBUILDING, SUCCESS CRITERIA

Marianne Mogensen, Logimatic A/S, Denmark

Introduction

This paper focuses on why Material Management in shipbuilding is conceptually different from material management in other types of industry, e.g. manufacturing cars, TV, mobile phones. The description is given through a listing of the main characteristics the material management processes have in shipbuilding.

By using these characteristics a set of requirements for the conceptual important functions needed in a Material Management System are described. Based on these functions reflections are made on the core functions available in the ERP Systems. Finally general considerations on the use of this type of systems in shipbuilding are given.

The information in this paper is based on more than 10 years experience of implementing a material control system in different shipyards worldwide. The experience comes from the work on design, development and implementing of dedicated solutions on systems to support the material management process in shipyards. No direct reference is made to any of the yards involved and the paper should be looked at as a 'Pragmatic description of an area of processes in a pragmatic business'.

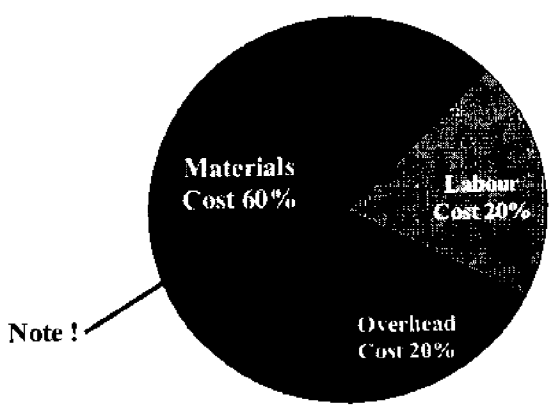
Why Focus On Material Control?

The price competition in the shipbuilding industry is traditionally strong and has gotten even stronger during the last few years. To keep the market position and gain new contracts every shipyard is working hard to become more productive, cutting down any possible overhead and not least reducing the labour cost.

In all the attempts to become more competitive there is often much to little focus on the importance of effective and accurate control of the materials required although this is probably one of the most important cost factors. Not only the potential direct savings in purchasing the "right" material at the right time and thereby reducing the financing cost is important. But also the importance of avoiding material shortage and interrupting/holding back the production is vital.

As a general thumb rule the split of the cost for commercial new building is as visualised in the following figure:

Figure 1 Split of costs for New Building



As it should be noted a 10% saving in Labour Cost gives a total saving of 2% whereas a 10% saving in material costs gives a total saving of 6% of the total newbuilding cost. For navy ships the material cost is even higher and close to 80%.

The high material cost improves the competitiveness for high labour rate countries while it on the contrary makes it almost equally important for the cheap labour countries to optimise the material management routines.

Characteristics of Shipbuilding Business in Relation to Material Management

What makes it complex and difficult to deal with material control in shipbuilding?

The primary reason is that an essential characteristic of material control in shipbuilding is high degree of uncertainty. This becomes critical in the light of the early commitment, which the shipyard has to make. Usually, the shipyard is requested to sign a contractual commitment at a time where the uncertainties in connection with the considered order are at the greatest e.g. procurement contract has to be made based on preliminary design detail.

In relation to this it is also a challenge to avoid purchase of material that ends up being surplus. The goal can be summed up: To ensure:

"The right material,
at the right time
at the right place,
in the right quantity"

When reflecting on the shipbuilding business in relation to material management the following specific characteristics can be highlighted:

- The project identification is the prime bearer of information.
- Shipbuilding is an interactive, concurrent time-wise process between design, procurement and production.
- Due to short project period the detailing level of the information is rather low in the beginning of the project.
- Shipbuilding companies are specialised in design, construction and production of individual plants where in principle all information except from general stock control is project/order related.
- All production is dedicated to a specific order/project
- The functions of the ship is described through a system breakdown structure
- Cost control performed up against a system breakdown structure
- No prototyping
- Requirements to traceability of material
- Deadlines for purchase and production controlled by the project plan (activities)
- The technical design is a basis for the purchase process
- The purchasing process is technically oriented complex and comprehensive
- Early purchasing of "key" items
- Project/order specific material > 80% of total material cost

- Special routines required for handling of steel material
- Outfitting process managed by palletising of materials

System Breakdown Structure and Cost Control

For every new building the functions in the ship are described through a system breakdown structure. The system breakdown is normally (and should be) functional related to support the design process, which is often done system wise. The different systems are split into smaller design tasks.

The specifications, estimates, material definition, purchase and material handling is all related to the different system and the pre-calculation, budgeting and follow up on material costs are based on systems structure.

Please note that the system structure will normally be totally different from the accounting structure used in the financial system.

Design

The design process is to a great extent performed as concurrent engineering with limited (none) prototyping. The challenge is to build the prototype and then to repeat it in a relatively limited series of sister project/ship.

The design is performed system oriented where the hull itself represents one system and the other systems covers the outfitting of the hull.

For a newbuilding project up to 80% of the material is project specific. This means that technical design is a basis for the purchase process and often the designers are in direct contact with the suppliers during the design process. For some key-items the designers performs the purchase work except for the actual commercial detail like payment term and delivery conditions, which are managed by the purchasing department.

Material Definition

For the strategic material needed for a project only a very small part can be considered as ordinary stock material and reordered based on regular consumption. Most of the material is defined specifically for each individual project although they might be standard materials. The material definition is performed as a part of the engineering process and concurrently when building up the part lists.

When the engineering process is performed as a concurrent process it is not possible to complete the material definition and material requirements before the purchase process can be initiated. By reason of this it is important to focus on the right material and quantity rather focusing on trying to bulk the material requirement which most likely will change at a later stage due to change in plan and/or design.

Material in a shipyard can be divided into two main categories: Outfitting Materials and Steel Material. The outfitting material are material like different types of equipment to be purchased. Steel material are divided into raw material to be purchase (Plates and profiles) and different steel parts, panels, pre-fabrications for building the hull.

Planning

The planning of the work is performed as a building plan defined through the creation of a work breakdown structure.

Much planning is based on experience and assumptions and requires therefore to be performed in steps starting with a rough schedule that is refined during the phase for the initial design and more elaborated during the detail design and production design phase.

In the final stage where the actual production is performed the plan is often changed caused by late deliveries or late changes in design.

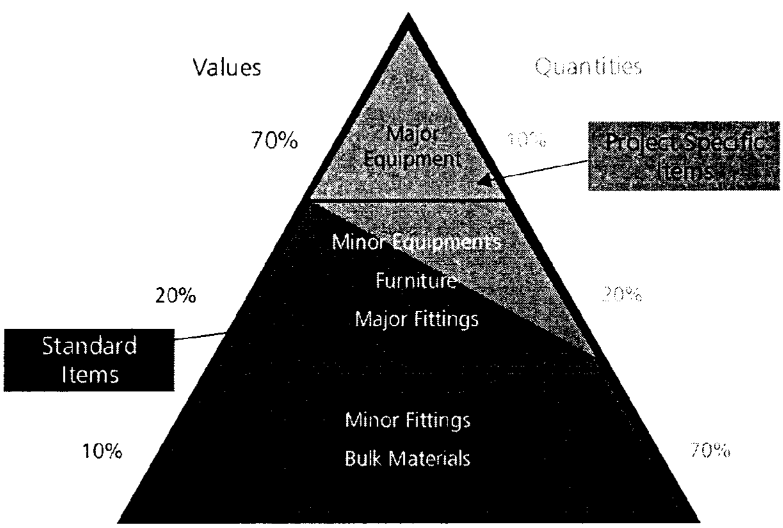
Procurement

The most difficult procurement processes for shipyards are those related to project specific components. The process is difficult because:

- Need for early purchasing of "key" items
- Information required is created iterative and final information is often only available after a purchase contract is placed.
- The purchasing process is very technically driven and requires teamwork between designers and buyers.
- Purchase of material is often made before the final need-date in production is decided.
- Steel material (plates and profiles) often need to be treated are project specific material

It is extremely important to focus on the key-items for a project. The key-items are the project specific material that is defined, purchased and used for a project and are often the materials with the longest delivery time and are the most critical items for the material handling. Buying in a wrong quantity will affect the material costs, which is illustrated in the following figure:

Figure 2 The Material Triangle



The figure shows a typically split between the quantities of material in relation to the values they represent. The figure underlines the importance of being able to uniquely identify and focus on the project specific materials as these represents up to 80% of the material costs.

Material Handling

The material flow inside a shipyard is performed according to the project plan. Two different material flows exist: One for handling the different steel parts for building the hull and one for handling the deliveries to production of the different outfitting material.

Delivery of material to hull production

The delivery of the plates and profiles is based on the planned start date for building a steel section. The materials are delivered to the shops where the planned cutting process is performed. The building structure and deadlines for each section control the following flows of material between the steel shop.

Delivery of material for outfitting

The deliveries in production consist of packages of material needed for a certain activity, 'palletising of material'. The material flow is not controlled by the use of routing and lead-time information. The foreman requests for the delivery of the material, which means that the materials are not pushed into production according to a predefined schedule, but are pulled according to a planned and confirmed need.

The planning of the production is very much de-centralised. The detailed planning of the production activities are performed according to the status of other activities, but also to a high degree of the availability of material e.g. a change of plan because of late deliveries of material.

Practical IT-solution

Based on the characteristics for the ship building industry the work process can be mapped into general requirements to a conceptually correct IT-solution.

In the following the key-functions needed in a practical IT-solution for a successful material control process are described.

Work Process Areas

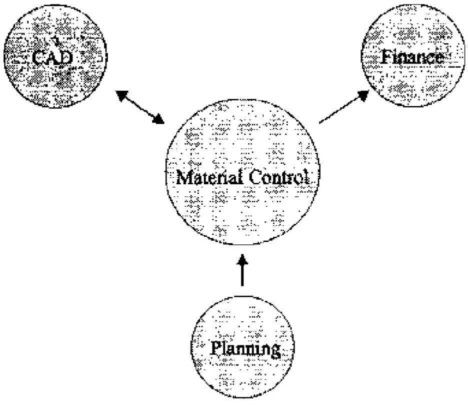
The material control process with special requirements in relation to shipbuilding are:

- System structure and cost management
- Design and material definition
- Part Listing
- Purchase
- Material handling

The main work processes focuses on the material control process. As a result of the identification of the characteristics of shipbuilding the production planning is performed as a work break down structure as an activity plan.

The important issue is that the timing of e.g. deliveries of material, completion of certain parts of systems can be defined without the plan is made in detail. It is therefore important to be able to relate the activities (milestones and jobs) to the material requirements and when a change of plan is introduced the timing of the delivery deadlines for materials are changed accordingly. The planning process is therefore not needed to be an integrated part of the material control system but as a vital tool like the CAD system and financial system, as illustrated in the following figure.

Figure 3 Material Control and related systems



System structure and cost management

In the project set-up process the definition of the System Structure of the new building is performed. The system structure serves two purposes, 1) the designers' tool for planning and dividing the work between the different design groups, 2) basis for Cost Management.

For serving these main purposes the system accounts must be an integrated part of the material control system. In relation to design/engineering areas such as material definition and part listing the relation to the different systems must be established, and in relation to the follow-up on the committed and final costs.

Design and material definition

The design process is performed as a concurrent process. Next to that the purchase of certain material and production of certain systems must be initiated before the design is completed. This sets up some requirements to the tools to use.

<u>Material Definition</u>

Traditionally, only one material catalogue is available. When only one catalogue is available it contains all the different material used in design, purchases, production etc. Material that can be characterised, as project specifics are often used only once but stays in the catalogue forever.

A use of only one material catalogue does not support the necessary flexibility in design process where a material can start up as being different items e.g. 2 different pumps, but after changes in design a different standard pump can be used instead.

Another important issue is that often the specification of material develops during the negotiation with the supplier and the material can therefore not be considered a standard material for other designers to use. Material like this needs to be separated from the standard materials.

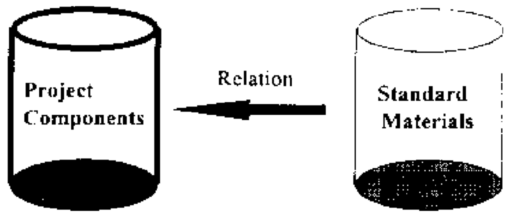
What designers need is a project specific catalogue for defining project components.

The project Component catalogue makes it possible to define project specific material. A project component is identified by: Project Id. + System Number + Free Number. This identification makes is possible to achieve full tracing of each component from the early definition through procurement, receiving, issuing of material. Further it is automatically organized systemwise.

During the design process of a system the materials needed are defined concurrently. The materials are defined as project specific materials, Project components e.g. the designer needs a certain amount of valves in a system. The valves are then defined as project Components and defined for a specific system. For some of these project components, e.g. main engine, an early purchase is needed.

For other project components the designer concludes that the valves can be purchase as a standard material. The designer then relates a standard material number to the project component.

Figure 4 Project Component and Standard Material Catalogues



As the figure illustrates the project components are placed in a separate catalogue. It is possible to relate a standard material to the project component. For timing of the requirements of these project components an activity can be related to each component describing the deadline for use for purchase or production.

Early Purchase needed

For some project components it is necessary to start up the purchase process before the design of the system is completed. Various reasons for this exits e.g. design is dependent on the suppliers ability to supply, long delivery time. At this stage these components are not yet finally placed in the final outfitting part list but the activity related to the components defines the timing of the material requirement.

The function to support this early start up of the purchase process is the Technical Purchase Order. In the technical purchase orders all technical details on the components to buy are defined. When technical purchase process is completed the order is transferred to the mercantile handler for completing the purchase order by adding various commercial information.

Part Listing

The part listing process in shipbuilding is split in two different areas: Part list for building of hull, and part list for outfitting. Both part list types are used for supporting the material control process in relation to purchase requirements and delivery of material to production.

Part list for Hull

The part lists created for the hull describe a hierarchical break down of the entire steel structure into elements such as blocks, section, sub-assemblies and panels. This parts list supports the building of the hull.

Traditionally material like steel plates and profiles has been considered stock material and purchased in bulk. For supporting the purchase process for buying steel and decrease the surplus amount, it is necessary to have a separate steel estimating function for defining and reservation of steel requirements pr. section/block. This estimation is then used when buying steel and when planning the delivery of plates and profiles to the different shops.

Part list for Outfitting

The part listing process consists of building up the material needed for the different design units. A design unit can be an assembly, a drawing, a pre-fabrication or similar. By relating the material from the catalogues to the parts list the material list for the design unit are defined.

It is possible to create a hierarchy of different design unit by relating these to each other. Example: A part list describes the material needed to fabricate a pipe spool. The part list the pipe spool is related to is the part list describing the material to install the pipeline the pipe spool is a part of.

The part lists and their relations, together with the activity, describes the material needed and the deadlines.

What is important is that by building the part list up in different design units the design and planning can support the production in a parallel process.

The part lists for outfitting have two main purposes:

- Defining the material requirements.
- Support of the outfitting process

For material and components for which early purchase is not needed the part lists defines the material requirements, which is presented in the material status.

To every part list line an activity is related and the timing of the material requirement is defined. The start date of the activity defines the date of need in production. By relating the activity to the different lines it is specified in which activity the material is to be used and a relation between the system breakdown structure and the work breakdown structure is established.

Procurement

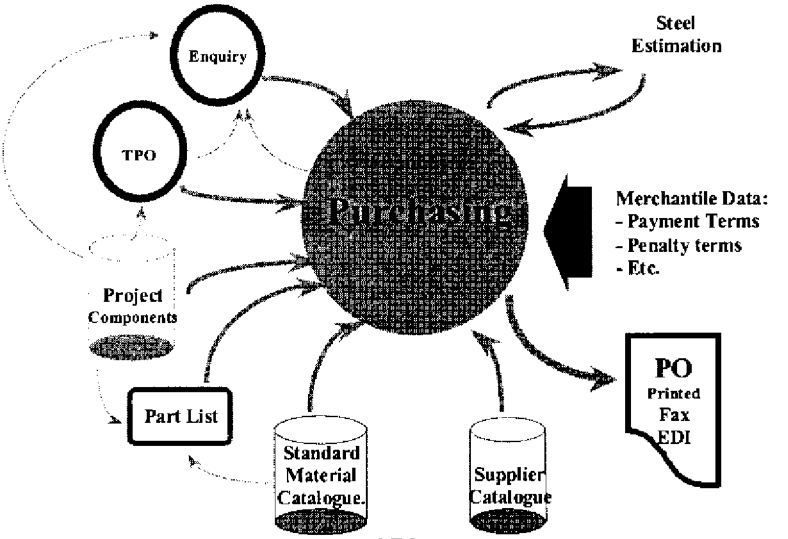
The procurement process is dependent on the category of material to purchase.

For project components and standard project material the purchase process is initiated by the designers creating the technical purchase orders. The buyer can then decide to perform an enquiry process.

For other category of material the purchase process is initiated by dealing with requisitions from designers and by checking the material status.

The overall philosophy is that this process is partly manual performed. The challenge is not how fancy the tool deals with reordering of standard catalogued items but to focus on the critical items. The purchase process is seen as a on-going process that collects the information from all different parts of the material control system, as illustrated in the following figure.

Figure 5 Purchase Process

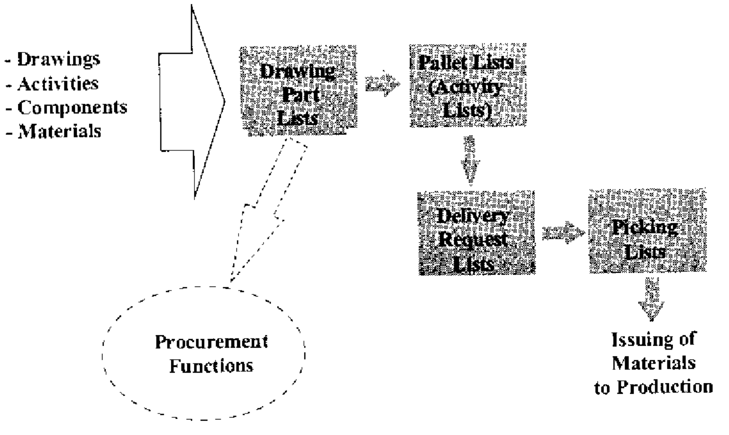


Material Handling

In shipbuilding the production plan is never that detailed that routing information can support a delivery of material to a specified workshop. Since the production planning is highly decentralised and performed by the foremen in production the materials are not delivered to production without any 'Request for material' made.

In the next figure the entire outfitting process is shown.

Figure 6 Outfitting Process



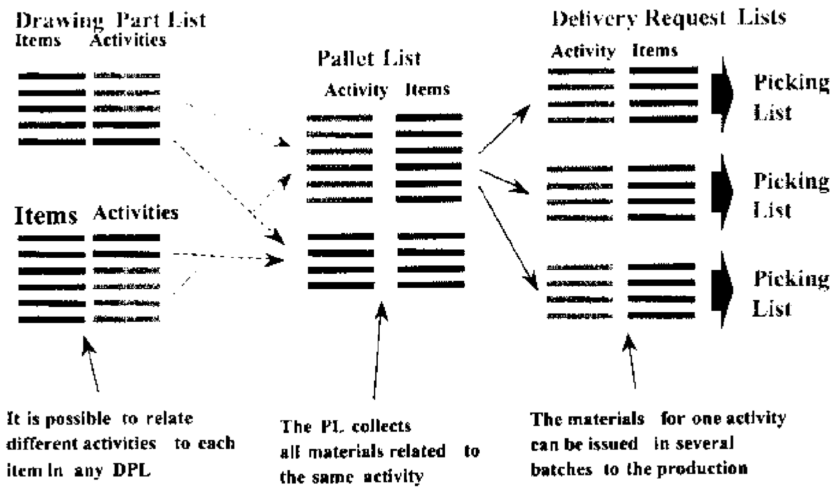
Before an activity/job is performed the material needed for this job can be identified by the use of the pallet list. The pallet list is the 'work package' of material for an activity.

The foreman in production request for the material to be delivered. Prior to the actual request for material the foreman must check the availability of the material in order to make changes to the outfitting plan. To avoid stops in production the goal is to install/outfit using the material available.

When the foreman in production approves the pallet list is becomes a 'Delivery Request List' and initiates the stock handling for creating the picking list and the final issuing of material to production.

When planning the outfitting work often the result is that a job involves material from different design unit (drawings, systems etc.). It is therefore needed to be able to relate different activities to lines in the same part lists. This function is show in the following figure.

Figure 7 Activities in different design units



ERP in ship building

Based on the facts that the shipbuilding industry demands different type of IT-solutions than the ordinary manufacturing industry many shipyards did earlier develop their own IT-systems. Today this is against the general IT-strategy as most companies try to replace 'home-made' solutions with readymade systems. This is one of the reasons why general-purpose ERP solutions have experienced explosive business development, especially within the ordinary manufacturing industry.

For shipyards it might also be easy and in some way attractive to follow the main stream and implement one of these EPR concepts. Some shipyards have even decided to go that route and have started up the practical implementation process.

The implementation of these systems has often failed and the result has been a burden to the whole organisation. Further the later operation of the system is troublesome even though considerable resources and costs have been spend on adapting the system to the company's working procedure.

The main shortcomings of ERP in relation to shipbuilding can be summarised as follows:

- Most ERP systems are based on MRPII theory which is developed for the general manufacturing industry (mass productions)
- ERP systems are fundamentally not designed for Project Driven Industry (One of a kind production) and their data structure will never become suitable for shipbuilding. Consequently the shipbuilding industry will never become their core business.
- The data structure of traditional ERP systems presupposes input of comparatively much information before valuable output can be extracted. This is in contradiction to the work process in a shipyard that is precisely characterised by an iterative process between design, procurement and production.
- The material control functions in ERP systems are normally based on a "predefined" and stable <u>B</u>ill <u>O</u>f <u>M</u>aterial (BOM). This provides the user with information for lead-time for purchasing well defined detailed material requirements and leadtime based on detailed and correct information on each assembly stage in a well defined line for production. All this information requires the required

final product to be totally defined up front both from a material view and a construction view when the order for the product is signed.

- There is a tendency that ERP systems put a heavy attention on work order generation, stores handling, stock control and production of standard items. These functions often create a lot of transactions that from a strategic point are less important for a shipbuilding enterprise.

Going for ERP or 'Best in Class'

Today the demands from the market are changing towards more modularised concepts where each core process is supported by a "Best in Class" solution.

In contrary to the integrated ERP solution the objective of the "Best in Class" concept is to provide the optimum IT-solution for each core business process. For the shipbuilding industry the main business processes are:

- CAD / CAM
- Material & Production Management
- General Project planning
- Financial & Human Resource

These systems should be based on open database technology. Each of them providing the necessary standard interfaces, which can be adapted to fulfil the need of each particular customer. If possible they should operate on the same relational database which makes it easier to extract the necessary management information into an Executive Information System.

Some of the main strengths of the "Best in Class" concept are:

- It provides optimum solution for each core business in the shipyard
- No compromises in functionality in order to achieve "one" integrated product
- Flexibility to upgrade each application as appropriate and required
- Less depended on one supplier
- It complies with the requirements to open and flexible systems

After all it is totally vital that the shipyard makes the optimum and correct decision on systems to support the core businesses which are the processes directly related to building of the ship. The optimum material & production control system is therefore probably the most important sub-system within the shipyard IT-environment and no compromises are allowed when choosing that software. Only the ultimate best functional solution is acceptable.

CAPTURING AND EXPLOITING KNOWLEDGE IN A COMPONENT-BASED SHIPBUILDING PRODUCT MODEL

Michael A. Polini, Newport News Shipbuilding, Newport News, Virginia
Patrice Blanchard, Intergraph Corporation, Paris, France

Abstract

During the past five years, significant research has been undertaken by the shipbuilding industry to apply object-oriented technology to product modeling information systems so as to capture and associate behavior with shipbuilding objects. The primary driver behind these efforts has been the need to become globally competitive in the face of a shrinking industrial base. Challenges facing the industry that must be addressed by the next-generation information systems include an aging workforce, the need to standardize not only the design but the information derived from product models, and the realization that automating the ship design process contributes to compressed acquisition schedules and lower overall costs. Advances in Information Technologies are rapidly changing the application development landscape and the current trend is towards user-friendly applications assembled from modular software components that adhere to documented protocols. In addition to the promise of a high level of re-usability, these components are appealing due to their ability to function in distributed, network environments—a necessity for scalable systems and virtual collaborative design environments. This paper will discuss the development of an integrated suite of shipbuilding applications developed using Microsoft's Component Object Model (COM) running under Windows NT on high-end PC's. It will address how knowledge is captured in the system, how the system can be customized to meet varied enterprise needs, how this knowledge must be managed in a collaborative, multi-discipline environment, and how this knowledge can be exploited to automate the ship design process

Introduction

At one extreme, computer scientists think of knowledge as something commonly associated with Artificial Intelligence (AI) technology and expert systems and their research concentrates on expanding the ways computers can use problem-solving knowledge to perform tasks that seem to require intelligence. From a different perspective, a mother would say that knowledge is not going out in the cold without a coat on. The Random House College Dictionary identifies *information*, *knowledge*, and *wisdom* as "terms for human acquirements through reading, study and practical experience". It goes further to define information as "that which applies to facts told, read, or communicated that may be unorganized and even unrelated." Knowledge "is an organized body of information, or the comprehension and understanding consequent on having acquired and organized a body of facts". Pragmatically, from the shipbuilder's viewpoint, knowledge represents the human elements that transform the abstraction of a ship concept into the multitude of parts that, when assembled, produce a vessel that meets its functional, financial, and schedule objectives. In a simplistic approach, shipbuilding knowledge can be represented by the "rules and standards" employed by an organization through the course of a design.

Newport News Shipbuilding (NNS) has been devoting a significant number of resources over the past several years to studying and optimizing the activities associated with the shipbuilding process across the entire enterprise. These business process re-engineering efforts have led to numerous and far-reaching changes to the shipyard—the most notable being the establishment of the "Shared Data Environment (SDE)" project. The objective of this effort is to link the individuals throughout the shipyard through an information network and provide integrated business and engineering applications that eliminate redundancy, centralize information, and minimize "paper-based" data exchanges. One of the key components of SDE is a next-generation shipbuilding product modeling system. Using state-of-the-art technology, this system is being designed to support the optimized shipbuilding processes being developed (i.e. the "best practices") and to provide a flexible mechanism for allowing the shipbuilder to define and embed knowledge—in the form of rules and standards—into the system. This paper will address technology that is the enabler for this knowledge capture and is deemed to be fundamental to achieving the productivity gains necessary to attain world class competitiveness.

The issues that will be discussed are symptomatic of the entire shipbuilding industry, as evidenced by the recent study sponsored by the National Shipbuilding Research Project (NSRP) in support of its Maritech Advanced Shipbuilding Enterprise (ASE) Program. The findings are documented in a Strategic Investment Plan (SIP) for the industry [reference 1] that identifies six major initiative areas that must be addressed to enhance the global competitiveness of the industry. The topic of this paper and the underlying research is germane to the "Advanced Design, Simulation, Estimating, and Analysis" and the "Component-based Software" sub-initiatives within the Systems Technologies" major initiative.

Background

Newport News Shipbuilding has been developing proprietary shipbuilding design systems for over a decade. Beginning in 1996, it built on the research efforts NNS had performed in support of the DARPA Simulation-based Design Program with respect to an object-oriented product modeling system and developed a design tool that was commonly referred to as the "smart product model". This UNIX-based system, known as workstation VIVID®, was put into production use for structural design at the shipyard in the fall of that year. It was a very advanced system that had a rich set of behaviors associated with a broad spectrum of shipbuilding objects and it prototyped many of the re-engineered processes within the engineering community as well as many of the leading-edge technologies in the computer science field. Significant progress was made in the areas of design standardization and automation through the use of rules.

In 1997, the team of Newport News Shipbuilding and Intergraph Corporation began working together to develop and deploy systems technology under the DARPA Maritech Program based on Microsoft's Windows NT operating system, their Component Object Model (COM, today called COM+), and their Distributed interNet Architecture (DNA) technology. This project was aptly named the Common Object Model of Products/Processes for an Advanced Shipbuilding System (COMPASS). It was believed that ground breaking shipbuilding systems—based on modern information systems technology—were required to achieve drastic productivity gains. In addition, empowerment at the worker and manager levels would be enabled when these new systems were financially attractive, intuitive, familiar, easy to use, and available at their desktops. Many vendors of commercial shipbuilding systems are faced with these same challenges and there are a variety of avenues that can be pursued to achieving these goals. This strategic partnership conscious choice to approach a next-generation shipbuilding design system by

- provide a core set of shipbuilding components to the industry
- supply a suite of end-user applications in the form of customized User Environments tailored to focused tasks in the shipbuilding process
- define processes that reflect the "best practices" associated with these tools

Figure 1 - COMPASS Benefits

developing a new architecture that utilized the experiences gained from their legacy products. The ramifications of such a decision led them away from having to worry about porting existing applications to new platforms, compromising the architecture to protect a large installed user-base, or considering putting a modern Graphical User Interface (GUI) veneer on top of legacy code. Instead, the team began building, from the ground up, a solid and sound system for shipbuilders that represented the first major technological innovation for shipbuilding information systems in the past decade. Figure 1 illustrates some of the objectives that the COMPASS program will fulfill.

Component Technology Overview

It is outside the scope of this paper to detail all the technical innovations and issues underlying the COMPASS program, however, an overview of the technology is warranted. This team is committed to component technology that is based on a 3-tier client-server architecture. These layers are commonly referred to as the "client" tier, the "business" tier, and the "data services" tier. This team follows a formal object-oriented development methodology that begins with Use Cases (the requirements as specified by the application domain experts) and progresses in an iterative manner through the software development activities of Analysis, Design, Code, and Test. For deployable products such as this, two additional activities are required, namely Training, and Implementation. These activities produce a variety of interim products (e.g. schedules, architecture and design documents, data models, test plans, implementation plans, and training materials) that culminate in the software components put in the hands of the shipbuilder.

Terminology in the software industry changes at a pace nearly as rapid as the underlying technology and those not immersed in the details—shipbuilders among them—tend to banter these terms about without understanding the subtleties associated with them. The ease with which some vendors changed their systems from "CAD tools" to "product modeling tools"—with no underlying architectural changes—as the term came into vogue is a case in point. A similar phenomenon can be observed with the terminology surrounding software development methodologies and frameworks, specifically procedural development, object-oriented development and component-based development. What are the fine distinctions between these seemingly innocuous terms? Taking a definition from an article in the October 1998 issue of the "Rose Architect" published by Rational Software, a component is "a non-trivial, nearly independent, and replaceable part of a system that fulfills a clear function in the context of a well-defined architecture". Components are about interfaces—documented, immutable specifications to enable consistent and dependable communication channels. Object-oriented programming focuses on the development of object-based components. Component development is concerned with deploying these components in such a manner that they can inter-operate with one another, can be assembled to form more complex, integrated systems, and that they can be widely distributed in client-server environments with no knowledge of who their clients are. As implied by the name, it is the software equivalent to the electronic industry's "black box" which, given some explicitly defined inputs, produces some defined outputs. What happens internally within the black box is of no concern and if one replaces this black box with another that has the same inputs and outputs, the result will be the same.

Component technologies can be traced back to a common origin—the Common Object Request Broker Architecture (CORBA) standard developed by a consortium of industry members known as the Object Management Group (OMG). Shortly after this standard was published, Microsoft created its own competing standard called the Component Object Model (COM+) and more recently, Sun Microsystems and IBM joined forces to develop their own competing standard called Enterprise JavaBeans (EJB). Each of these standards address the basic problems facing today's application developers—namely language-independent object definitions and transparent distribution of objects within a single process, between multiple processes, and remotely over a network (reference 2 provides an in-depth treatment of this technology from the COM perspective). It is the distribution capabilities of these technologies that has the biggest impact on the application architecture separating computer roles into "client", "business", and "data" categories (i.e. multi-tiered systems) and having enormous consequences for performance, scalability, reliability, security, and cost.

Intergraph was first in the engineering design tools industry to understand and appreciate the importance of Microsoft, Windows NT, COM+ and component-based interoperability standards, as well as the general effectiveness of the Windows development environment. Consequently, they moved quickly to create Windows-based graphics capabilities and implemented several user interface innovations specifically for graphics environments.

Intergraph further undertook definition of an extension of Object Linking and Embedding (OLE), called OLE for Design and Modeling (OLE for D&M) that permits engineering graphics tools to inter-operate with one another and with office automation (OA) tools such as word processors, spreadsheets, and presentation graphics applications. More recently, Intergraph has created a new, component-based Enterprise Engineering Environment (EEE) for application development and

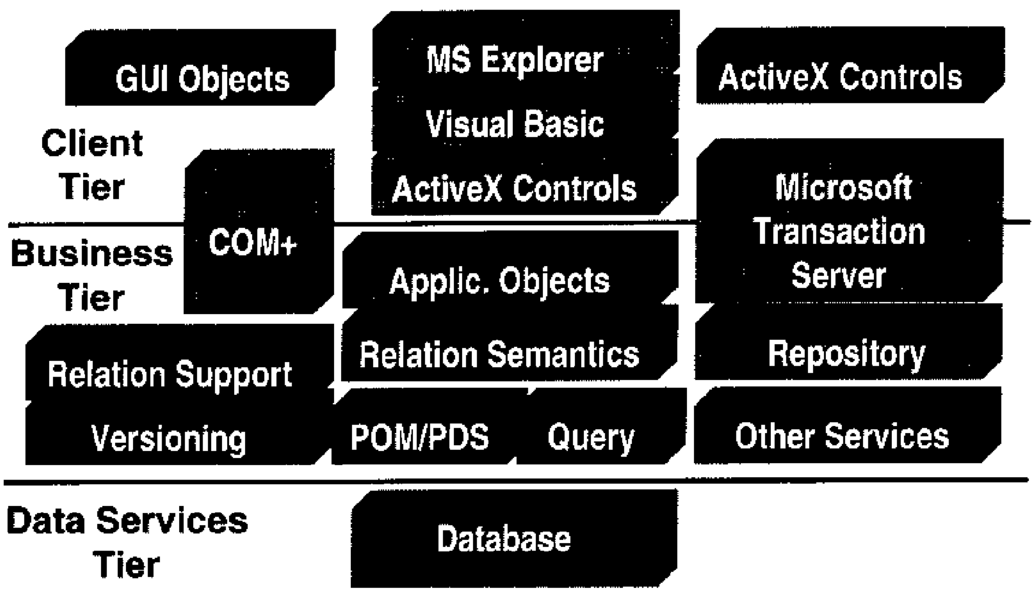


Figure 2 – EEE Framework

engineering data management. This EEE is designed to address the problems with today's CAD engine-based applications. It is the infrastructure, integrated tools, reusable software components, interoperability standards, and other capabilities needed to improve design efficiency, to facilitate early stage design optimization, and to integrate engineering systems with both the enterprise and operation and maintenance environments. Intergraph's EEE system will minimize overall information technology costs by leveraging Microsoft's Windows NT operating system, Windows user interface and application design standards, the COM+-based DNA architecture, incorporation of OLE-compliant OA applications, and COM+ compliant development tools and capabilities.

The EEE also includes user applications. Windows NT and the DNA infrastructure are foundational elements; EEE framework components layer upon the operating system and distributed computing infrastructure; application components layer on the framework; and numerous small applications use the services of the components and underlying infrastructure (Figure 2). Applications, known as User Environments (UE), developed on this architecture are small, task-focused, and easy-to-use. Intergraph, along with its shipbuilding development partners, envisions a broad slate of these task-focused and easy-to-use ship design and production applications.

In summary, the importance of the preceding section is to highlight the breadth of new technology that can be brought to bear on the problems facing the shipbuilding industry.

Elements of Knowledge Capture

The ultimate question from the shipbuilder is "what does all this mean to me in terms of my day to day business of building ships?" To answer this question, the Molded Form User Environment (UE) will be studied in detail. This UE has been specifically tailored to enable the creation and modification of objects that represent the zero-thickness curves and surfaces that give the ship its basic form and will be used by shipbuilders in the early stages of design. These objects live on the middle-tier of the architecture with numerous objects from other UE's. Collectively, these are commonly referred to as "business objects". While there are many objects within the system at this level the majority are necessary to support the application developer and only a few are meaningful and exposed to the shipbuilding end-user. In order to effectively create and interact with the business objects, software must be written at the client-tier level. This software takes the form of the commands and user-interface components necessary to effect the task-focused UE that reflects the domain of interest. Microsoft's Visual Basic (VB) language is commonly used to develop client-tier applications and components. In addition, VB can be used to define "rules" that can be invoked by business objects to modify and/or create other business objects. Each of these three areas—User Environments, Business Objects, and Rules—will be discussed in more detail in the following paragraphs.

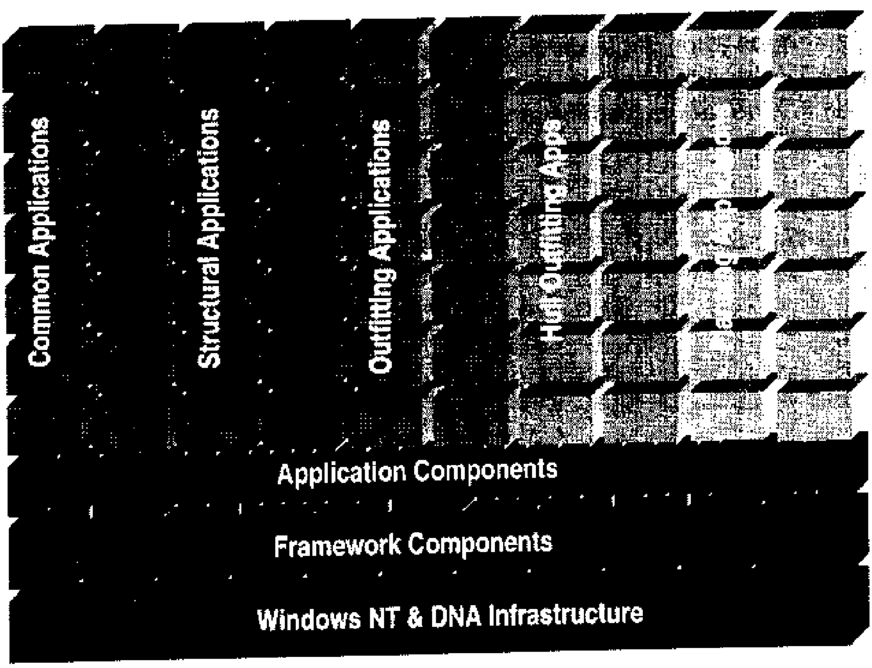


Figure 3 - User Environments

User Environments

The 3-tier architecture represents a layered system approach whereby some services build on other services. Those at the lower level are typically referred to as core services. There is another partitioning that can take place that subsets the application domain space rather than the technology domain space. This sub-setting corresponds to the User Environments described above. At the most elementary level, these UE's can be thought of as the most rudimentary form of knowledge since they organize the vast amount of data (i.e. information) about the ship into small, well-focused, partitioned subsets. Figure 3 illustrates this concept.

Business Objects

Business Objects are another major element necessary to enable knowledge capture within the system. There is a close coupling between the business objects in the middle-tier and the software written for the client-tier. Trying to deal with enforcing rules at the client-tier means that each client must understand the intricacies of the architecture and implement the rules on a case by case basis. Obviously, this approach is unlikely to succeed because it will be difficult if not impossible to educate all potential clients on the architecture and impossible to validate whether all client applications correctly implemented the rules. Even if it were possible to overcome these obstacles, the resulting system would be prohibitively expensive to maintain as changes to the rules would have to be accompanied by changes to each of the clients. Concerning the underlying knowledge, there is a significant advantage in embedding this information into the business objects themselves as opposed to handling it at the client-tier level. At the business object level, this knowledge need only be captured once and it is guaranteed to result in consistent behavior for each and every client.

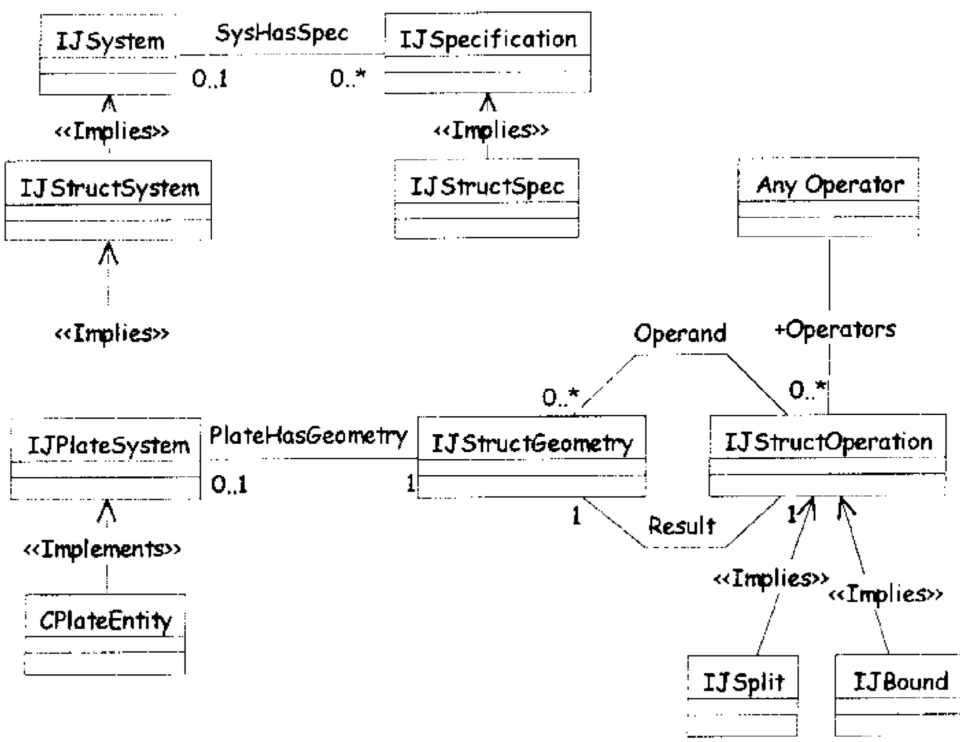


Figure 4 - UML Data Model

Data models of the components are developed as part of the software engineering process. These data models define the attributes (i.e. properties) and relationships between the various objects. The Unified Model Language (UML) is used to formally capture this information. Figure 4 represents a sample UML data model. Business objects can be categorized by their expected behavior within the system and can either be persistent—that is saved from session to session—or non-persistent living only for a brief period of time within a single session.

Passive entities are the persistent business objects that represent the physical things making up the ship. These entities store the static attributes appropriate to the object. A "*plate system*" will be used as an example of a passive entity in our scenario, however, *profile systems, seams, openings, connections*, and *structural specifications* are other business objects within the Molded Form UE. The business objects in this category are those typically presented to the application user.

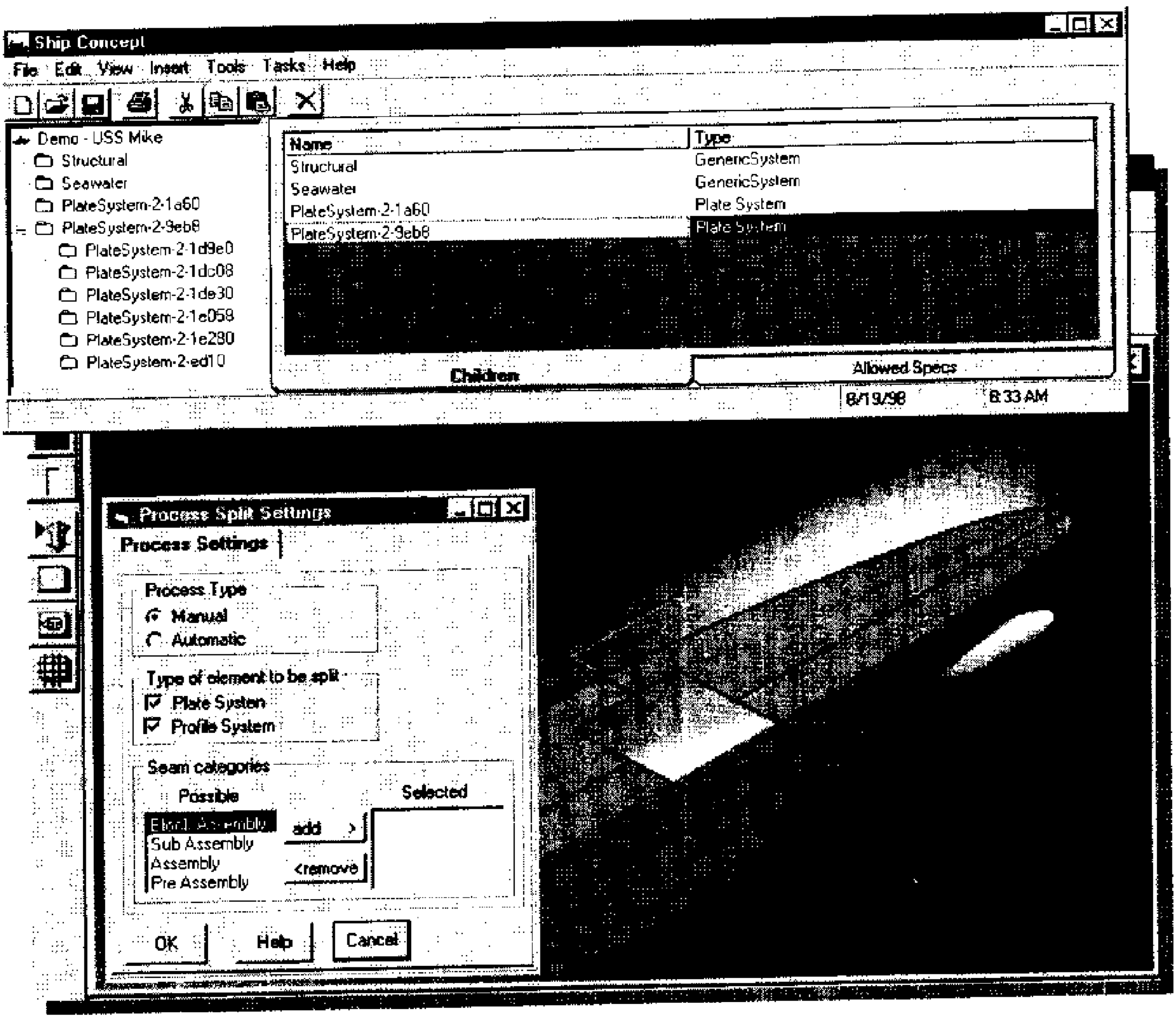


Figure 5 - Split Operation

Active entities are the persistent business objects that represent the operations that manipulate and transform the passive entities. These entities are responsible for identifying the inputs and outputs of the operation as reflected by relationships with other passive entities. Again, within the Molded

Form UE, *create*, *split*, *cut*, and *bound* are operations represented as active entities. These entities are usually presented indirectly to the application end-user in the form of commands associated with a tool bar or a menu choice and usually only directly addressed by application developers. Figure 5 is an example of the split operation.

Semantics are the non-persistent objects that perform the update to the passive entities based on the relationships defined by the active entities they are associated with. These semantics are triggered by the relationship support component when any of the inputs to the active entity are modified. These semantics make use of utilities that perform much of the actual modifications to the passive entity. These objects are never exposed to the end-user and are created and maintained by the application developer.

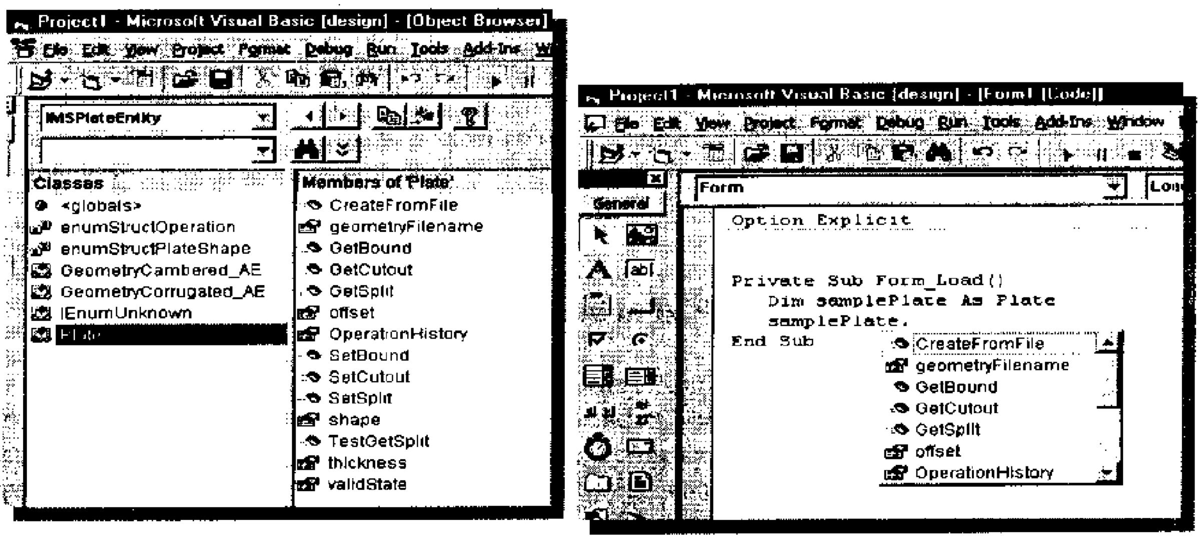


Figure 6 - Visual Basic Example

Through careful analysis of end-user requirements, appropriate business objects can be designed and implemented that result in components suitable for the task at hand. These components can be presented to the user through a narrowly focused, highly customized user interface that has been optimized for a particular work flow within some aspect of the ship design process. These business objects capture and record—either through attribution or via explicit relationships—information as the design process progresses. This information embodies knowledge transferred from the mind of the designer into the product model database in a manner that is suitable for exploitation by other aspects of system.

Rules

One means of supporting rule definition that is both flexible and extensible is through Visual Basic (VB) programs. Some number of these programs could be supplied by the system developers and serve as templates for users wishing to customize their system. Because the data models are captured within the repository, the business objects are capable of automatically exposing their properties and interfaces from within the development environment. Figure 6 is a screen capture of such a situation.

Given that there exists a means to define a rule, the challenge is then to provide a mechanism by which the rules can be invoked. The authors propose the concept of a "structural specification" business object that can serve as the "dispatcher" for the default rule behavior. This structural specification can be associated with the plate and profile system business objects and can be divided into categories that correspond with the common operations acting on the objects. Within these categories may be any number of situations where a specific rule would be needed. As an operation is executed, the specification would be queried for the known configuration and the rule invoked to supply the default behavior. Several categories and common situations are outlined below:

Naming Rules

- Plate systems
- Profile systems
- Plate parts
- Profile parts
- Seams
- Openings
- Connections
- Features

Connections Rules

- Continuity
- Profile end to profile end
  - aligned
  - knuckled landing curve
  - orthogonal landing curve
- Profile end to profile edge
  - aligned
  - orthogonal
- Profile edge to profile edge
- Profile edge to profile face
- Profile end to plate face
- Profile edge to plate face
- Profile edge to plate edge
- Plate edge to plate edge
- Plate edge to plate face
- Plate face to plate face
- Equipment to plate face
- Equipment to profile face
- Hanger to plate face
- Hanger to profile edge

Penetration Rules

- Tightness
- Profile penetrating plate
- Profile penetrating profile
- Seam penetrating plate
- Seam penetrating profile
- Opening penetrating plate
  - Normal Access (e.g. doors)
  - Limited Access (e.g. manhole)
  - Lightening
- Pipe penetrating plate
- Pipe penetrating profile
- Duct penetrating plate
- Duct penetrating profile
- Electrical penetrating plate
- Electrical penetrating profile
- Opening penetrating profile

Knowledge Exploitation

Application data models are publicly exposed within the Microsoft Repository and they may be created and/or modified with graphical editing tools such as Rational Rose or Microsoft Visual Modeler. These tools present data models as Unified Modeling Language (UML) diagrams. This eliminates the use of obtuse ASCII file-based "meta-data" definitions typical of many current generation applications. Application data models may be extended with new properties, which will display in property dialog boxes. Or, they may be extended with new objects and relationships that may be implemented as COM objects identified within the meta-data. These can be manipulated with either modified application components or new commands in client applications. Likewise, new relationship semantics can be added to implement specialized rule-based behaviors triggered by the framework. Thus, the Repository supports a completely unprecedented level of open applications.

In addition to the application components described above, a host of framework-supplied reusable software components are available or under development and include:

- ActiveX controls for generic application commands (e.g., view manipulation commands, printing and plotting, etc.)
- A Relation Support component for change notification and propagation
- Query and Versioning components
- A persistent data manager component to provide database isolation
- A variety of miscellaneous framework services.

As mentioned earlier, the "relation support" component is the key to enabling the exploitation of the knowledge embedded into the system in the form of all three categories of business objects.

In order to support rule-based design, a "relation support" component has been provided and is responsible for reading application meta-data (i.e. the information model or schema) out of the Repository and using it to intelligently propagate the impact of user actions and software-induced changes to the data. The meta-data contains knowledge of application logic associated with relationships between application entities. These so-called "relationship semantics" are intelligently triggered for execution by the Relation Support component, thereby providing the generic mechanism required to solve the problems of application editing, reuse of historical data, and change management described earlier.

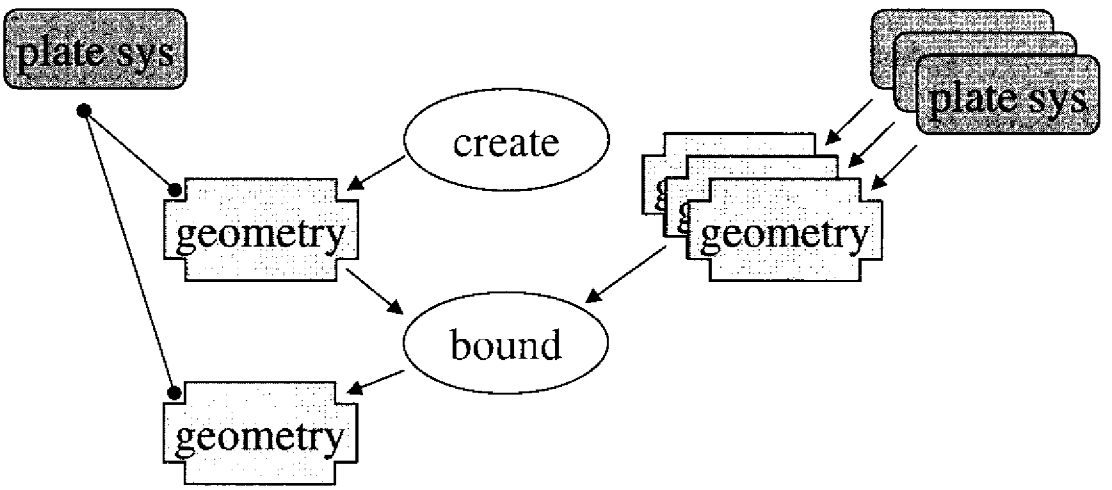


Figure 7 - Relationship Graph

Figure 7 is an example of what the internal state of the product model may look like after a series of user actions. A passive plate system object was created and at some point, a "create" operation resulted in some geometry being associated with the plate system. At some later point, a "bound" operation was invoked and the geometry from other plate systems was recorded as input. The semantics associated with this active entity caused the geometry of the original plate system to be trimmed by the geometry of the bounding plate systems. In a similar manner, downstream design activities can result in cut and split operations that recorded the relationships with the various business objects. What is important to point out is that it is possible to control the degree of propagation of change with such a mechanism. For instance, should an attribute on one of the bounding plate systems be modified, the relationship support component would not trigger the semantic associated with the bound active entity because the plate system was not an input. Only when a change to the geometry of the bounding plate system occurs will the update take place.

Business Case

To round out the discussions related to embedding knowledge into product modeling systems, a high-level overview of the business case behind this effort will be presented. As stated earlier, NNS has been studying and refining the activities performed by the engineers and planners over the course of the ship design lifecycle. These studies have identified three major issues related to product modeling that have a significant impact on productivity. The first is the need to begin product modeling in the earlier stages of the lifecycle, the second is the need to standardize the design products, and the third is to defer the actual creation of the parts for as long as possible. Using the illustration in Figure 8, each of these will be discussed in more detail below.

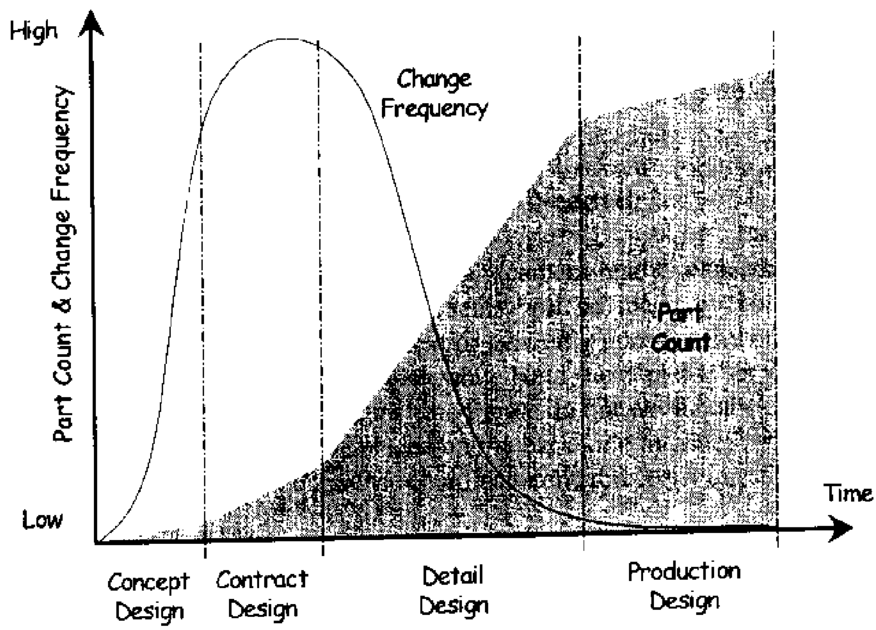


Figure 8 – Product Lifecycle

Product Models in Conceptual Design

Most traditional CAD modeling systems use geometry as the underlying representation of the product and then associate attributes with it to enhance its definition. While this approach can be used to build ships, it comes with one major drawback—one cannot begin to build the CAD model until the geometry is known. In the early stages of design, much of the information is not geometric in nature. One may know very early that there will be five decks on the vessel, but they may not know where they are located, whether they will be planar or not, or what their exact shape will be. For this reason, shipbuilders who use these traditional CAD modeling tools, tend to think of them as detail and production design tools and use other means to develop the concept and contract design products. Should the shipyard be successful in winning the work, the CAD modeling effort begins anew with the detail design activities. Since the concept and contract products are the means by which the price and schedule are negotiated and established, any discrepancies between the model developed in these early stages and the CAD model developed during the downstream detail and production design stages can lead to cost and schedule overruns.

Using an object-oriented product model, geometry is treated as an optional attribute of the object. With this approach, the shipbuilder works at a higher level of abstraction with objects that closely represent the physical aspects of the ship. The objects can be created early in the lifecycle, with or without geometry as appropriate. Relationships can be established between the objects at this stage that are independent of the underlying geometry. As the design progresses, other non-geometric attributes can be defined and the product model undergoes a gradual refinement throughout the lifecycle until it fully matures into the finished product.

Standard Design Products

The term "standard design products" is used to refer to the re-usability aspect within or across ship design projects. Several studies have cited the fact that U.S. shipbuilders re-engineer a much higher percentage of the ship than our foreign counterparts. In many ways, these efforts could be simply viewed as wasting engineering time and money, however, in the broader sense, they can lead to serious downstream problems since these newly engineered products have never been validated in a production setting. NNS has found that one of the biggest factors affecting the quality of the design and productivity of the end-user was embedding rules into the product modeling system. One of the most critical activities under a "design for production" approach is that of lofting—the art or science of adjusting the part geometry to account for fit-up clearances, weld shrinkage, and/or added (i.e. green) material. This role was traditionally reserved for the most experienced and skilled designers and engineers who took the detailed parts when they were completed and performed their magic on them before sending them off to be manufactured. In many cases, the adjustments were made in the $1/16^{th}$ of an inch range and were not obvious to the naked eye. More importantly, there was no consistency between one loftsman and another and there was nothing but the individual's personal discipline that ensured that all parts had been lofted correctly. This resulted in processes that included activities for checking and validating the parts on the engineering side, activities for checking the parts at the manufacturing site, and accuracy control activities for checking and validating the parts after they were produced. Indirectly, the processes for procuring material had "safety valves" built into them in the form of excess material to re-do parts found to be in error and the assembly schedules were padded to allow for the extra time necessary to correct these errors.

Once the rules were embedded into the product modeling system and validated by the downstream users, many of these extraneous activities became unnecessary. The need for constant checking at each stage of the design process could be eliminated, and the focus of accuracy control shifted from a piece part mentality to an overall process quality mode. The products coming out of the engineering organization were standardized and consistent with the expectations of the production organization, were feasible for manufacturing within the confines of the company's facilities, and represented the best practice, lowest cost alternative. Where there was once a critical skill shortage in the engineering process concerning lofting, the product modeling tool turned every user into a top-notch loftsman. The black magic that was once associated with these activities became a well understood, documented set of rules that became part of the corporate knowledge-base and could be monitored and maintained at a strategic level.

Deferred Part Creation

There are two other observations that are important about these early design stage activities based on the figure. The frequency of change is very high early in the lifecycle, however, the number of "parts" is very low. The term parts is used liberally in this context and an argument can be made that there is a distinction between the business objects that represent the early stages of the ship product model and the business objects that represent the later stages. The authors have chosen to address this distinction by referring to the early stage objects as "systems" and the downstream objects as "parts". Systems can be equated with the logical or functional definition of the ship, while parts correspond to the physical definition. Many idealizations are possible at the system level which simplify greatly the internal data definitions necessary to create an early lifecycle model of the vessel. Experience with such concepts on a product model of a 40,000 DWT commercial tanker resulted in several thousand structural plate and profile systems and about 50,000 plate and profile parts—an order of magnitude difference! The number of geometry objects (i.e. curves and surfaces) representing these parts numbered about 350,000, again another order of magnitude difference. While it may be possible to effectively keep up with changes in the early stages of design, it becomes increasingly more expensive and time consuming to do so in the later stages of the lifecycle. Even though the frequency of change has slowed significantly, the sheer number of parts that may need to be updated can be overwhelming and configuration management becomes the governing factor. Without the ability to capture the knowledge that went into the design beginning in the early stages, it becomes a purely manual, labor-intensive exercise to incorporate change after the bulk of the parts have been created.

Conclusion

The authors have had considerable experience with object-oriented product modeling systems and the concepts described in this paper. While commercial products built on this technology are not anticipated to be available in the marketplace until the year 2000, advanced prototypes and internal pilot implementations have shown great promise in reducing design time, improving the final product quality, and lowering the required skill level of the user. By capturing knowledge through the use of active entities and semantics at the business object level, and by supporting a mechanism for automation using rules, the actual creation of parts can be deferred until very late in the design cycle. The domain experts can shift from a piece-part mentality to a systems engineering mode where they 1) concentrate their efforts on creating the system business objects; 2) establish the appropriate relationships between objects; and 3) associate rules with the business objects to drive automation. The tedious part creation activities can be automatically performed by the system following a validated set of rules, thereby guaranteeing product quality and significantly compressed design periods.

References

1. Sessions, Roger, "COM and DCOM", John Wiley & Sons, Inc., Copyright 1998.
2. National Shipbuilding Research Program, "Strategic Investment Plan", August 14th, 1998, www.nsrp.org.

AN INTEGRATED DESIGN AND PRODUCTION ENVIRONMENT FOR SHIP MACHINERY SYSTEMS

Robert Bronsart and Steffen Gau, University of Rostock,
Institute of Maritime Systems and Fluid Engineering, Rostock, Germany

Abstract

The paper describes the ship machinery systems design and production process chain. Different partner involved are identified and their specific function is explained. Communication processes forming the base for the concurrent and distributed production environment are thoroughly analysed with respect to who exchanges what kind of information and their related interdependencies. Existing standards, especially those under development for the exchange of product data relevant to the worked out communication scenario e.g. STEP AP 217, 226 and 227 are discussed. It is shown that the scope of these APs as defined today overlap considerably. The process integration strategy followed in an ongoing R&D project of a German shipyard and a consortium consisting of engineering subcontractor, classification society, communication technology provider, module manufacturer is described.

Introduction

Until some years ago, almost the complete design and production of ship machinery systems was carried out at the shipyard building the vessel. Today the situation has changed as shipbuilders world wide try to stay competitive. In this context, the same strategic approach as in other industries, like the car industry, is followed by many shipyards: more and more activities of different process steps are performed by subcontractors. Whereas some traditional relationships exist like ship model basin doing the power prediction, the new strategy results in a design and production environment, in which engineering subcontractors are assigned overall responsibility for complete onboard systems e.g. heavy fuel oil, lubrication oil or freshwater system. Based on the functional specification laid down in piping and instrumentation diagrams, the makers list and additional necessary information on the general arrangement as well as the steel structure layout supplied by the shipyard, engineering subcontractors perform the detailed design like routing of pipes, ducts and cables and by this generate fabrication information to be used in the downstream process steps. Beyond this design task, engineering subcontractors also have to take care for the approval by classification societies, the manufacture of components and modules by subcontractors as well as the installation of the systems onboard the vessel at the shipyard.

This "virtual enterprise environment" for design and production of one-of-a-kind products which consist of many and complex systems and the very short time to market has resulted into problems not known before. The necessary transactions of today manly paper based information between the partners involved has partially led in a slow down of the processes and the necessity to control different product descriptions in different versions at different sites. To overcame these problems, the definition of a commonly used product model and the usage of state of the art information technology are regarded as key factors. The following principal benefits are identified by the maritime industry:

- reduction of design and production time which in turn results in the reduction of costs,
- elimination of errors due to inconsistency problems caused by multiple product definitions used by the partners involved,
- support of version control for design variants.

Many research and development resources were and are still allocated for the definition of product models suitable to meet the formulated communication requirements. Kendall and Hasund [1] give a short overview of the STEP application protocols with an application domain related to shipbuilding, Langbecker and Rabien [2] describe the activities under the umbrella of the European Maritime STEP Association (EMSA). The defined business cases worked on in the R&D projects mainly focus on the exchange of hull form and ship structural data.

In this paper, the design and production process chain of ship machinery systems is analysed to identify those business cases with a great potential for an increase of the overall productivity. The technology used will principally be the same as for the above mentioned business cases but based on different product model scope.

Ship Machinery Systems – Design and Production Process

The design and production process of ship machinery systems today is performed in a concurrent and distributed working environment. This overall situation is better described when focusing on a configuration as seen in practice. A consortium consisting of shipyard, engineering subcontractor, classification society and module manufacturer serves as an example. These four partners are involved in a co-operation scenario working on the fuel oil supply system and major components thereof. The entire process chain comprises pre-design, schema approval, detailed design including generation of manufacturing information, material and component ordering and logistics, parts manufacturing and assembly and final approval as well as the installation of complete modules onboard the vessel. Within this situation each partner serves clearly described functions and has to take different responsibilities.

A module in this context is an assembly of components which in general belong to one piping system. The prefabricated unit is installed onboard and connected via pre-defined interfaces. For an example of a module, in this case part of the fuel oil system, please see **Figure 1**.

In the following, the today's situation is described to some detail, please refer also to **Figure 2**.

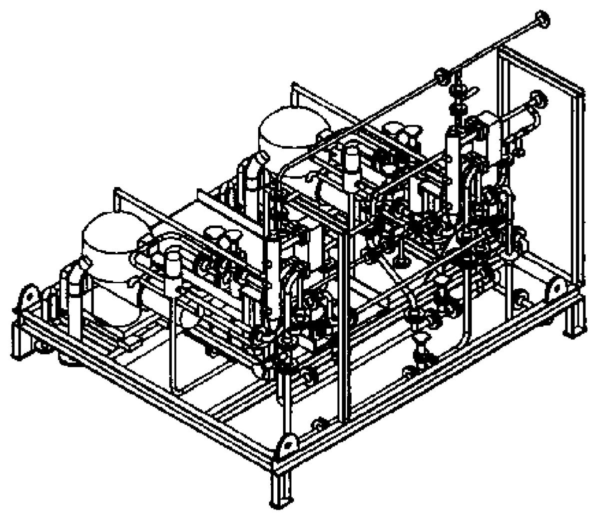


Figure 1. Example of a HFO module (courtesy of MTE)

Shipyard

In accordance with the owner requirements and rules and regulations to be observed the shipyard creates early stage product information like general arrangement, engine room layout and principle piping diagrams of the most relevant systems. These and some other documents form the information set which is called 'classification project' and are sent to the classification society for approval.

The approved documents are sent back to the shipyard where they are stored, copied and transferred together with additional information to the engineering subcontractor. All of these documents may be called the 'technical project'.

Classification society

At the beginning the classification society receives a classification inquiry issued by the shipyard. This results in prescribing a unique registration number to the ship to be built.

The classification project supplied by the shipyard is directed to the departments responsible. An analysis and approval follows. The classification society is utilising its specific regulations as well as internationally valid laws and restrictions. When the classification project documents do not comply with the rules the classification society is requesting changes and/or improvements from the shipyard. This results in an extended communication between classification society and shipyard. In case all documents comply with the rules the classification society will document it's approval by adding the approval stamp and signature to the relevant documents. This stage marks the milestone for further detail work.

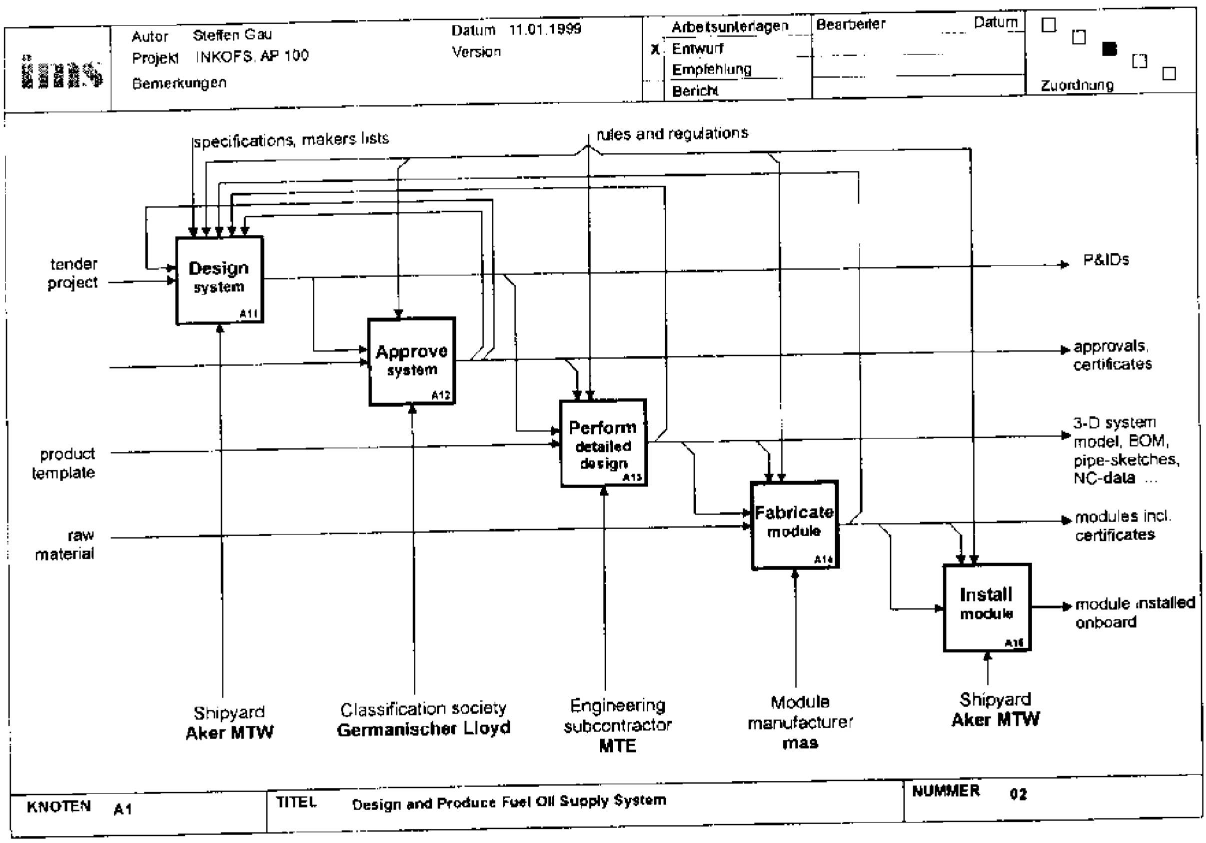


Figure 2. Design and Production Process for Ship Machinery Systems

Engineering subcontractor

Provided with the approved documents and other shipyard specific information as well as owner specific requirements like the makers lists, the engineering subcontractor is carrying out the detailed design of the systems. These activities are done taking classification and as necessary further regulations into account and using additional information from the shipyard regarding other specific systems. A 3-D model of all components and pipes is generated. In co-operation with the shipyard module parameter and interim results are exchanged. An other major task is the generation of fabrication information. This includes pipe sketches, piece lists and even control data for (C)NC machines used by the module manufacturer. Generated data are processed and delivered to the shipyard and module manufacturer and kept in the own project archive.

Module manufacturer

The fabrication of piece parts and assembly of complete modules starts on the basis of the results of the upstream process steps. Out of numerous information sources all necessary documents are to be selected, materials and components in accordance to the makers lists and required specifications have to be ordered. In parallel the existing resources are checked and capacities are assigned to specific fabrication steps in the assembly tree (production planning).

The task do deliver a fully prefabricated ready-to-operate module asks for direct contact between manufacturer and classification society. Several built-in components as well as the completed module need to be approved according to classification society rules and regulations. These approvals have to be carried out by classification surveyors at the manufacturers site, often before the module is completed (without insulation, painting,...).

The delivery of modules and components as well as the complete product documentation including all classification certificates has to be agreed upon between module manufacturer and shipyard. All hardware is installed onboard the vessel at the shipyard, the related documents like approval certificates, operation manuals etc. are handed over to the shipyard to either be kept in the files or put onboard.

When looking at the chain of activities it can been seen that different types of co-operation have to be distinguished. Each is representing a specific configuration, see **Table 1**.

Table 1. Partners and Communication Links

	Shipyard	**Classification Society**	**Engineering subcontractor**	**Module manufacturer**
Shipyard		SC	SE	SM
Classification society	CS		CE	CM
Engineering subcontract.	ES	EC		EM
Module manufacturer	MS	MC	ME	

Communication Link			Partners Participating			Business Case
SC	CS	1	Shipyard	↔	Classification society	I + IV
SE	ES	2	Shipyard	↔	Engineering subcontractor	II
EM	ME	3	Engineering subcontractor	↔	Module manufacturer	III
MC	CM	4	Module manufacturer	↔	Classification society	IV
MS	SM	5	Module manufacturer	↔	Shipyard	-
EC	CE	6	Engineering subcontractor	↔	Classification society	I

According to **Table 1**, 12 communication links exist, looking at each communication from both partners involved. In order to generalise 6 different communication scenarios are identified.

Comparing these scenarios with the process chain depicted in **Figure 2**, it can be seen that link no. 6 does not occur in the actual co-operation activities. Communication links are determined by the participating partners and their communication requirements. These information exchange scenarios may be regarded as business bases which are characterised by the parameter described below.

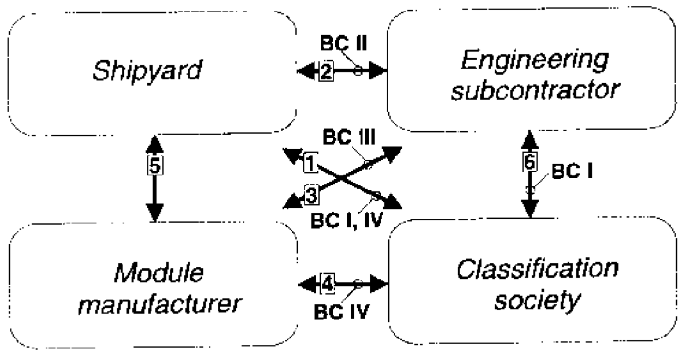


Figure 3. Co-operation Network and Business Cases

Communication Requirements

Business cases as defined above and depicted in **Figure 3** represent inter-company communication processes. In the following each of the identified business cases is described by

- the principle partner involved,
- the activities to be supported,
- the information representation in the data exchange as used today,
- the amount of data (qualitative statement),
- the "direction" of information exchange, if applicable,
- the communication technology used today for the information exchange, if applicable.

BC I Design and approval of ship machinery systems

Example link: Shipyard ↔ Classification society

Task	Information representation	Data volume	Data flow	CT used today
Functional design of ship machinery systems	CAD database, P&ID	very high	S internal	-
Exchange of classification relevant documents	text, drawings	high	S → C	mail
Approval of systems	text, drawings	high	C internal	-
Change requests Communication	oral discussion, text, drawings	low - high	C ↔ S	mail, fax, phone, email
Transfer of certificate (sign and stamp)	text, drawings	low	C → S	mail

BC II Ship machinery module detailed design

Example link: Shipyard ↔ Engineering subcontractor

Task	**Information representation**	**Data volume**	**Data flow**	**CT used today**
Transfer of specification and approved documents	text, drawings	high	S → E	mail, fax, email
Generation of detailed module design data	CAD data, text, drawings	very high	E internal	-
Communication on interfaces to other systems	oral discussion, text, drawings	low - medium	E ↔ S	phone, fax, mail, email
Delivery of design information	text, drawings	very high	E → S	mail, email

BC III Ship machinery production engineering

Example link: Engineering subcontractor ↔ Module manufacturer

Task	**Information representation**	**Data volume**	**Data flow**	**CT used today**
Generation of fabrication and assembly data	text, drawings, NC-files	very high	E internal	-
Exchange of design and manufacturing data	text, drawings, NC-files	high	E → M	mail, email
Communication on manufacturing details	oral discussion, text, drawings	low - medium	M ↔ E	phone, fax, mail, email
Transfer of as-built data	text, drawings	low	M → E	mail, fax, email

BC IV Survey and approval of ship machinery systems

Example link: Module manufacturer ↔ Classification Society

Task	**Information representation**	**Data volume**	**Data flow**	**CT used today**
Request for survey and approval	text, drawings	low	M → C	mail, fax
Communication on survey details	oral, text	very low	C ↔ M	phone, fax
On-site survey of components / module	text, drawings, survey form	very low	C	on-site discussion
Transfer of classification Certificates	certificate	very low	C → M	mail

The primary information elements of the four business cases are listed in **Table 2**. Apart from the exchange of high volume data, an informal communication exists is many cases. Especially in those situations when additional ad hoc information is needed by any of the partners involved, the personal communication is regarded as the most efficient one. This is important to keep in mind when trying to formalise the communication in a "virtual enterprise" which is of highly dynamic nature.

Table 2. Primary Information Elements

Business case	**Information elements in scope of data exchange**
Design and approval of ship machinery systems	– Ship machinery specification – Makers list – Functional system design (P&ID) – Connection of components – Instrumentation and control – Material, stream – Catalogue – Approval/change – External references
Ship machinery module detailed design	– Ship machinery specification – Makers list – Functional system design – Connection of components – Piece parts, assembly – Shape representation 3-D, location – Catalogue – Approval/change
Ship machinery production engineering	– Piece parts, assembly – Shape representation 3-D – Fabrication data (NC) – BOM – Catalogue
Survey and approval of ship machinery systems	– Ship machinery specification – Functional system design – Instrumentation and control – Material – Approval/change – History

Business Cases versus STEP Application Protocols

Currently there are five application protocols under development which focus on the exchange of information in shipbuilding: Ship Arrangements (AP215), Ship Moulded Forms (AP216), Ship Piping (AP217), Ship Structures (AP218), and Ship Mechanical Systems (AP226). In the context of a ship machinery systems application domain, the application protocol "Plant Spatial Configuration" (AP227) which relates to "Functional data and their schematic representation for process plant"

(AP221) have to be looked at as well. In the following an overview of the scope of the APs is given.

Ship Piping (AP217)

The scope of the application protocol for ship piping is defined to cover the life-cycle phases: functional design, detailed design, production engineering, fabrication and assembly as well as testing, see **Figure** 4. According to the committee draft for comments [3], the following are within the scope:

- data required to support the definition of the operating flow states of a piping system for the purpose of analyzing flow conditions, computing required pipe sizes, and for documenting the operational conditions of piping systems for shipboard personnel;
- data required to support the definition of the geometry and rigidity of a piping system for the purpose of evaluating stresses in the system obtained by applying loads;
- data sufficient to describe the geometry and location of equipment connected by piping and other distributive systems;
- data that defines the geometry of piping components, the equipment to which they attach, and envelopes surrounding these objects sufficient to enable interference analysis;
- data that defines the sequence of bending operations needed to bend a fabricated pipe;
- data describing the assembly operations necessary to assemble a piping assembly;
- data that defines the test procedures that evaluate the proper operation of a piping system or subsystem;
- data necessary to support the extraction of a bill of material data for a piping system or piping assembly;
- data necessary to document the configuration status of one or more piping components;
- data that defines the maintenance requirements, history and status of a piping system or collection of piping components.

The conformance classes defined are grouped into the subsets:

- functional piping design,
- detailed piping design,
- production engineering design,
- piping test data,
- piping maintenance and repair data

with additional variations in each subset with respect to shape representation and configuration management information.

Compared to the other APs described below, this is a very broad scope definition.

Ship Mechanical Systems (AP226)

In the working draft of this application protocol [4], the scope is defined as follows. Lifecycle phases to support are specification, design/selection, approval, installation, commissioning, acceptance, operation, in-service inspection and maintenance, decommissioning and disposal.

Systems and components to support are air supply to the engine room, exhaust gas, fuel oil treatment, lubrication oil and engine cooling, propulsion drive line, main and auxiliary engines, thruster units, pumps, heat exchangers, air compressors, boilers, deck machinery, ... For each of these systems and components, the product definition information is in scope:

- functional and physical connectivity including connectivity to ship structure
- functional description such as performance and operational characteristics
- geometric representation

- technological information such as material, tolerance, ...
- data necessary to track lifecycle and operational history such as specification, inspection and maintenance data.

Declared out of scope are: piping arrangements, electrical distribution systems and control systems not integral to the machinery unit, ship's heating, ventilation and air conditioning (HVAC), cargo refrigeration, ... , data related to the manufacturing of systems and components.

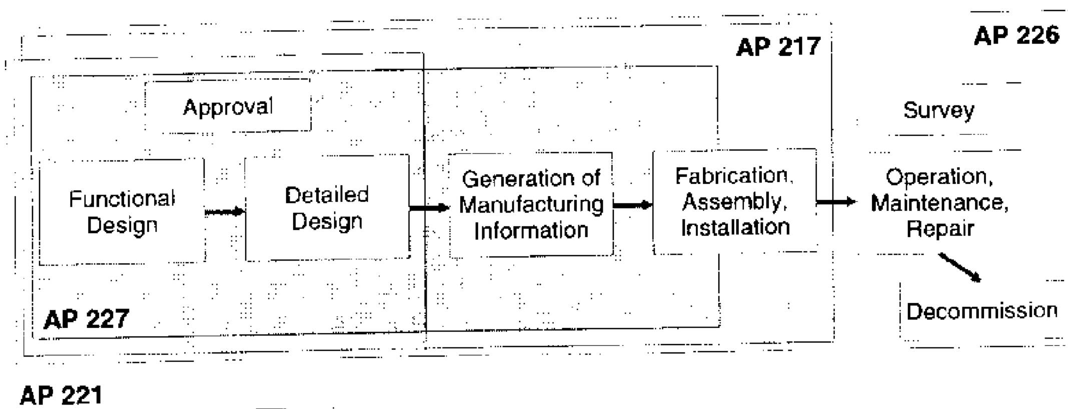


Figure 4. Life Cycle Activity Coverage

Plant Spatial Configuration (AP227)

According to the introduction in the draft international standard (DIS) document, this application protocol [5] is to be used "for the exchange of the spatial configuration information of process plants[1]. The information includes the shape and spatial arrangement characteristics of piping system components as well as the shape and spatial arrangement characteristics of other related plant systems (i.e. electrical, instrumentation and controls, and structural systems) that impact the design and layout of piping systems. In the design and fabrication of a piping system, the piping layout must be evaluated with respect to the spatial characteristics and arrangement of these related plant systems, and the requirements for clearances between systems. The complete specification of these other systems is not needed, but enough spatial information is needed to support the layout of the piping system.

This AP specifies additional requirements for the exchange of information required for the design and fabrication of a piping system. This includes information on the piping material, process stream fluid, and the piping system functional characteristics. A process and system design specifies process requirements for a piping system that includes pipe size, design temperatures and pressures, and insulation class. The physical design uses these process requirements for the design of the piping system.

The application protocol also identifies and provides a functional specification of the components of the plant piping system. The design information for a piping system may specify a pump capable of maintaining a pressure and flow rate. The design will also specify the shape limitations or requirements and the location of the pump in the system, but not sufficient information for the fabrica-

[1] A process plant is defined as "an assembly of one or more plant systems and plant items that can, or is intended to perform, a chemical, physical or transport process. A process plant is identified as a single unit for the purpose of management and ownership. A process plant has both physical and functional aspects." (definition 3.3.37 in [5]).

tion of the pump. The principle focus of the AP is on piping systems and the shape and spatial arrangement of systems including plant items required to ensure the physical integrity of piping systems."

Furthermore, the functional view on the piping systems, traditionally documented in the piping and instrumentation diagrams (P&IDs) is supported. The representation of these diagrams (drawings) however are not supported by the AP. The shape of items making up the plant may be represented at various levels of abstraction, i.e. from an encompassing envelope to a detailed design description. The requirements of the following business cases are to be satisfied by this AP:

1. Exchange of requirements from a plant owner to an architectural engineering (AE) firm;
2. Exchange of process requirements for the plant piping system from a process engineer to a system design engineer;
3. Integration of designs created by different engineers;
4. Detection of physical interferences of plant piping system components with components of other plant systems;
5. Exchange of construction specifications between AE and construction firms.

The conformance classes are defined:

Class 1 - Provides piping system functional information,

Class 2 - Provides equipment and component spatial information,

Class 3 - Provides plant layout and piping design information,

Class 4 - Provides piping fabrication and installation information.

It has to be noted, that compared to the shipbuilding application protocols, this AP has already DIS status and though can be regarded fairly stable with respect to it's contents.

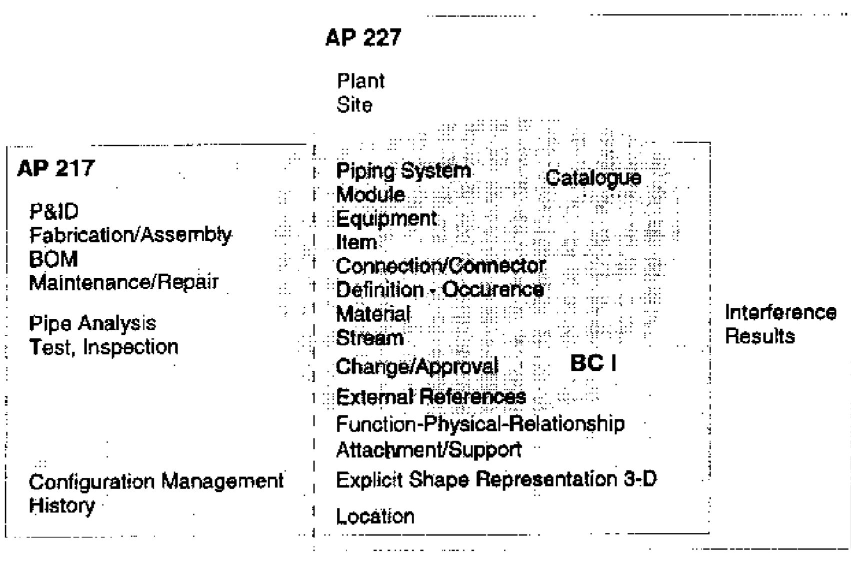


Figure 5. Scope Overlap Between AP 217 and AP 227, BC I

The application protocol "Functional data and their schematic representation for process plant" (AP221) focuses on the exchange of functional data and their schematic representation for process plants. It complements AP 227 in that P&IDs in form of a graphical representation, are in scope whereas the spatial configuration and shape as well as production information of pipe parts and components are out of scope. Apart from this major difference, the large overlap in scope will result in a decision to be made on which AP implementations for an information exchange will be realised.

Comparison of Life-cycle Coverage and Scope

Figure 4 shows how the different life-cycle phases are supported by the APs. Whereas the usage of the application protocol for "Plant Spatial Configuration" is principally restricted to the plant engineering life-cycle phases, the AP for "Ship Piping" also supports information exchange in the production and testing phases. AP 226 however is planned to support the whole life-cycle of the vessel. According to the above outlined scope definition, the application protocol for "Ship Mechanical Systems" differs from the other two in that the level of detail for the product definition is far less. Another important difference is to be seen in the focus on the equipment rather than on the piping components. In **Figure 5** the two APs with mainly focus on piping systems are depicted, indicating the concepts defined part of the corresponding scope. It can be seen, that there is a considerable overlap between AP 217 and AP 227. As one example, the concepts to be used in the first business case, "design and approval of ship machinery systems" are highlighted.

Conclusion

The analysis of the design and production process chain for ship machinery systems in a distributed environment results in four inter-company communication scenarios with different information exchange contents. The existing application protocols, namely AP 217 and AP 227, are able to fulfil (most of) the formulated requirements. The considerable overlap in scope of the two APs and the DIS status of the application protocol "Plant spatial configuration" might lead to the decision to make use of AP 227 for the exchange of data of ship machinery systems. A thorough analysis of this AP with respect to the requirements of the above outlined communication links will be performed. This approach is also stimulated by the experience gained in the implementation of AP 217 (ARM level). The work done by KCS in the MariSTEP project has shown that major modifications to the ship piping AP are necessary as the existing version does not allow for an efficient data exchange of pipe components.

According to the results of the ongoing R&D project, it has to be kept in mind that for the realisation of an IT based communication infrastructure, the highly dynamic nature of the overall scenario has to be taken care for. The potential for an increase of productivity is dependent on the amount of information to be exchanged between the companies involved as well as on the frequency of the communication.

Acknowledgement

The work presented in this paper is supported by the Federal German Ministry of Education, Science and Research and Technology. The authors acknowledge the valuable discussions with the members of the project team from the shipyard Aker MTW, from the engineering subcontractor MTE, from Germanischer Lloyd, from the module manufacturer *mas* and from the information technology supplier Scheller.

References

1. John Kendall and Knut Hasund, "Product Data Integration and Exchange for Ship Design and Operation", in *The International Conference on Computer Applications in Shipbuilding*, Yokohama, 1997, Volume 1, pp. 569-587.

2. Uwe Langbecker and Uwe Rabien, "Practical Use of STEP Data Models in Ship Design and Analysis", in *The International Conference on Computer Applications in Shipbuilding*, Yokohama, 1997, Volume 1, pp. 553-568.
3. Application Protocol 217: Ship piping systems, ISO TC184/SC4/WG3 N494, 1996 ➤ www.nist.gov/sc4/step/parts/part217
4. Application Protocol 226: Ship mechanical systems, ISO TC184/SC4/WG3 N730, 1998 ➤ www.nist.gov/sc4/step/parts/part226
5. Application Protocol 227: Plant spatial configuration, ISO TC184/SC4/WG3 N600, 1997 ➤ www.nist.gov/sc4/step/parts/part227
6. Application Protocol 221: Functional data and their schematic representation for process plant, ISO TC184/SC4/WG3 N600, 1997 ➤ www.nist.gov/sc4/step/parts/part221

The authors' email are: robert.bronsart@mbst.uni-rostock.de
steffen.gau@mbst.uni-rostock.de

Product Modelling

ELECTRONIC COMMERCE AND EDI
NEW STRATEGIES FOR ENHANCED SHIPBUILDING SUPPLY CHAINS

Joachim Brodda, BALance Technology Consulting, Bremen, Germany

1 Introduction

Investments in the last 10 years within shipyards world-wide led to increasing levels of automation and process integration with substantial improvements of productivity in "blue collar" areas. Most advanced shipyards developed from craft-skill-based workshop technologies towards highly robotised shipbuilding factories. Beside the consequent development of all CAD/CAM processes the shipyards also improved their organisational and logistical functions. This was necessary to serve the needs of those highly integrated production areas, but also to improve the productivity of "white collar" functions.

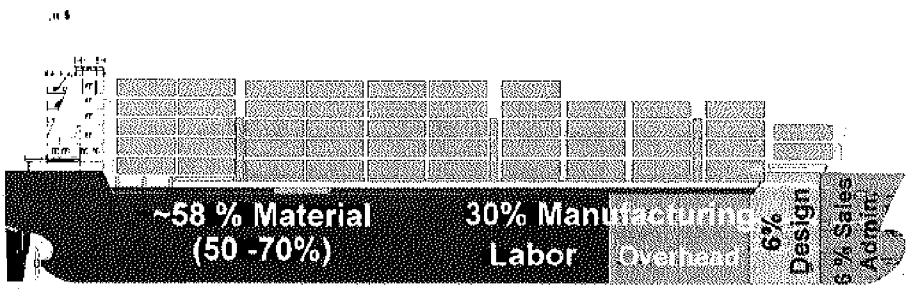
Most developments were oriented to improve functions and processes inside the individual companies. Whereas, new and advanced Information Society Technologies (IST), namely, Electronic Data Interchange (EDI) and Electronic Commerce by means of Internet and Multimedia technologies, open up new fields for business improvements through inter-company process integration. These technologies will sustainable change and enhance shipbuilding supply chains and co-operative engineering processes. In the future, shipyards and their co-operational partners will develop to powerful temporary networks, so called Virtual Enterprises (VE) which on the basis of integrated processes by means of information and communication technology (ICT) will behave like one large entitiy. Since material supplies and external engineering services count for 50-70% of the ship cost the potential for improvements in this field is comparable high.

This paper will discuss the state-of-the-art in EDI and Electronic Commerce technologies and its application in shipbuilding supply chains. On the basis of actual trends in the shipbuilding industry towards more co-operative working, outsourcing and system supplies, the potential application areas of EDI/EC technologies will be discussed including its commercial implications on the process, i.e. cost savings and lead time reductions. Results from European research projects, e.g. EDIMAR (EDI for the European Maritime Industry) and MARVEL OUS (Maritime Virtual Enterprise Linkage – Open User Syndicate) have contributed to the discussion and definition of standards. These projects have proven the functionality in different test scenarios by adapting technology ready for application, i.e. available products and services. This draws also the scenario for ongoing projects activities and on future development needs.

2 Motivation

Competition in shipbuilding is constantly increasing in the shipbuilding industry for many years. After long phases of recession in the 70ies and 80ies the markets are improving. However, through new capacities coming into the market the competitive position of many shipyards has not improved. Searching for alternative solutions to improve the competitive position the shipbuilding industry is following world wide a trend towards further outsourcing of processes and services to benefit from a better specialisation and cost performance of smaller and self-responsible companies, i.e. marine equipment manufacturers and service companies. Compared to the total cost of a ship this trend leaves the shipyards today with a share of 50 – 70% material cost, 20 35% manufacturing cost (labour and overhead), 5–10% engineering/design cost and 5-10% sales and administration cost (Figure 1).

Figure 1. Typical ship cost structure (Example)



© BALance Technology Consulting

Some of the potential advantages of outsourcing have been bought in for the price of dramatically increasing overhead and management cost. This is because many companies have not properly prepared themselves to keep control of services and processes which they so far performed by themselves. On the other hand shipyard personnel often are not used to adequate management techniques and tools to manage external resources and the purchasing process effectively. Therefore, material and services overhead cost are summing up to 5 – 8 % of the total ship cost not saying anything about the time losses through incomplete and inadequate information in the process.
This substantial share of cost, which is very often not clearly visible, but hidden in other cost positions, provide a very good motivation and a large potential for cost savings.

3 Understanding the Maritime Supply Chain

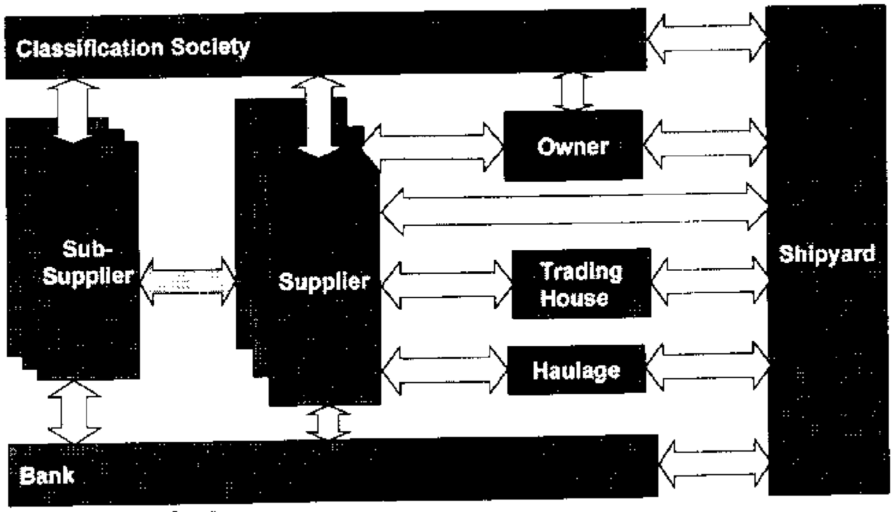
3.1 Elements of the Supply Chain

Elements of the maritime supply chain can be found in almost all processes of ship newbuilding. The respective work and effects cannot be limited to the work of the purchasing department, but has to be considered as an integrated element in almost all functions throughout the shipbuilding process. It already starts with the suppliers seeking for markets and acceptance of their products and it continuous after the ship delivery through life cycle supporting functions in the area of maintenance and repair. Besides the shipyards and the suppliers maritime supply chains involve many external partners directly and indirectly (Figure 2) and requires an extensive communication process including specially applied management procedures.
More specific the different elements of the supply chain may be distinguished in the following six business processes:

Marine Equipment Type Approval

Before entering into the market marine equipment suppliers have to obtain type approvals from classification societies for their products. This is a major entrance barrier into the market and as well a time critical, lengthy and expensive process for the suppliers. It mainly involves the suppliers and the classification societies. The procedures involve laboratory tests and complex administrative procedures. For some products testing of the individual products are requested beyond type approval.

Figure 2. Partners in the Supply Chain



Pre-Selection of Equipment and Materials (Engineering and Design Process)

A process of vital importance for the supplier is the engineering and design process of the shipyard. The optimal situation for a supplier is to be no. 1 choice on the ship-owner's maker list, which requires after good quality and reliable products continuous marketing and sales efforts to maintain the position in the market. But even then it is necessary to be "at hand" for the designer either in form of good catalogues and fast accessible technical information, technical advisory and support services and maybe through good personal relations.

Procurement Process

Often in parallel, sometimes in a sequential order or even before the detailed design process has started, purchasing activities begin with inquiries in the market by requesting quotations from suppliers. This is not necessary for those products where the shipyards have negotiated framework contracts with the suppliers including fixed price structures. The bidding process is followed by the order process including sending the formal orders and receiving respective order responses including confirmations respectively changes to the order. Since shipbuilding is a very dynamic process with highly concurrent engineering processes order changes are frequently necessary. In some special cases the purchasing process will be handled through trading houses, which may receive better prices than single shipyards through some special framework contracts and access to different markets.

Material Delivery Process

The purchasing process itself is followed by close tracking of the order, especially with respect to the delivery date. For some supplies the timely delivery is of vital importance for the shipbuilding schedule. Therefore a continuous contact to the suppliers will be maintained and even shipping and transport are subject of close tracking. However, all the paperwork including transport data, delivery notes, despatch advices, invoices and storage intake control are subject of this process. It may also comprise the handling of certificates which may accompany the supplies. These have to be handled and administered carefully and become later an element of the overall documentation for the ship.

On Site Assembly, Functional Testing, Approval

Whoever is responsible for the assembly of materials and components (shipyard or supplier) will rely on good documentation for the assembly procedure. Access to remote information through direct contacts or advanced media can be important for fast and reliable mounting on site. This is followed by functional testing of systems which may involve again the classification societies. Beside the technical testing this is also a very formal procedure with excessive amounts of documentation and administrative procedure. Again the availability of all required information, certificates etc. is important for fast and reliable procedures.

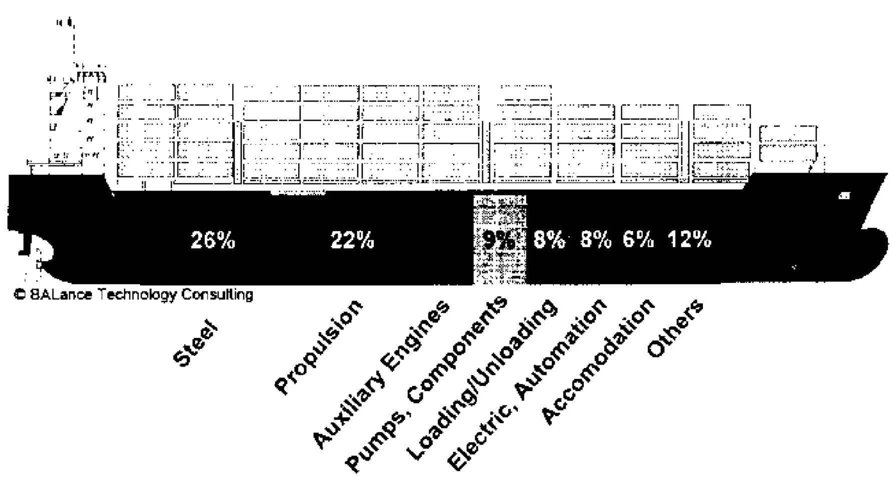
Guarantee Process, After Sales Services, Maintenance and Repair

After the delivery of the ship normally all involved parties, i.e. the owner, the ship, the classification society and the shipyard file comprehensive sets of all documentation for the ship. For all events like potential guarantee processes, renewing the class, regular maintenance or emergency repair it is essential to have fast and reliable access to all documentation of the ship. Documentation needs to be small in size, but comprehensive and supportive in its content.

3.2 The Supplier Base of Shipyards

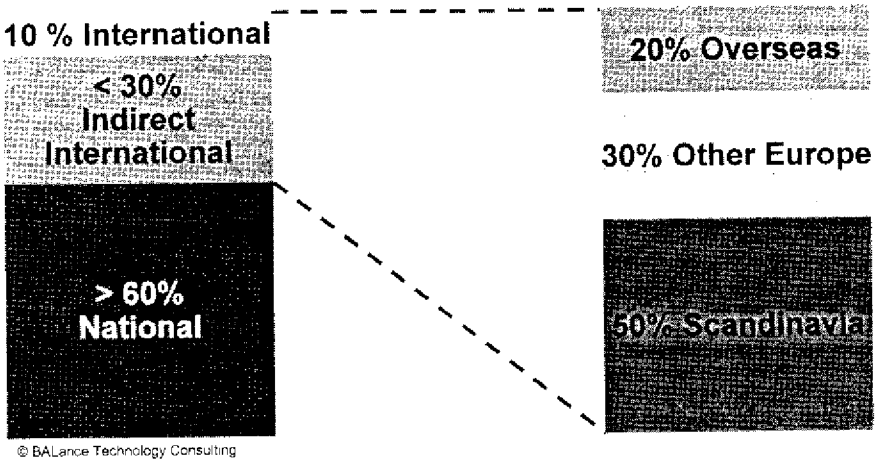
Depending on the ship-type the relative value of the manifold product groups of ship equipment may be very different. The cost-share of accommodation for a cruise ship has a totally different dimension than for a normal cargo ship. The cost-share of electrical equipment or electronic components for naval vessels are incomparable to those for tankers and so forth. Figure 3 gives an example for a typical distribution of material cost for a container-ship from a North-European shipyard. The relative value of material and other external cost against the ship total cost depend on the level of outsourcing and also on the regional employment cost of the shipyards, i.e. in regions with low employment-costs the total share for material may be relatively higher.

Figure 3. Typical distribution of material cost (Example)



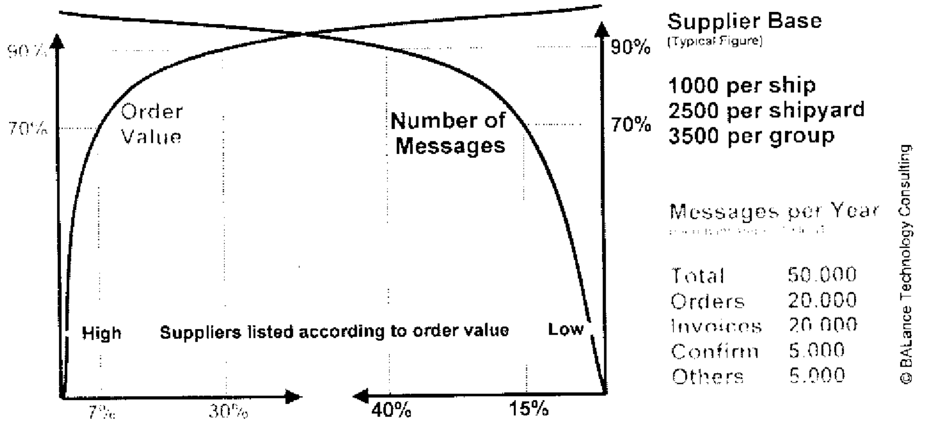
The total number of suppliers for a large shipyard or a shipbuilding conglomerate may sum up to around 3.500 companies. For the building of a general cargo ship the shipyard may employ about 1.000 of them. To understand the structure of the suppliers, their meaning for the shipyard and to develop suitable ways to treat them in the right way, respectively to develop strategies for improvements in the future relations the shipyards need to carry out extensive analysis. Beyond information about the products, country, reliability, owner structure, solvency etc. which need continuous updating, data about order volume, special abilities, flexibility and continuous exchange of messages is necessary. The treatment of supplier data in relational databases have facilitated this work in the last years tremendously. Targeted evaluation and interpretation of data is easily possible. Results are for instance the international level of material sourcing (Figure 4) or the identification of key suppliers by order value or of those with the highest numbers of messages to be exchanged.

Figure 4. International Sourcing (Example)



Examples from shipyards show that about 7% of the companies involved absorb about 70% of the ordering value (30% companies absorb 90%). Compared to that the shipyard exchanges with 15% of the companies about 70% of all messages (respectively with 40% about 90% messages). But, companies with high order values not necessarily represent those with high numbers of messages. In most cases it is the contrary (Figure 5)

Figure 5. Extended Supplier Analysis



Understanding the different supply chain processes and working on the supplier base to derive most valuable information on the meaning of the different suppliers for the shipyard is essential to develop and evaluate new and advanced strategies. Some shipyards have developed complimentary strategies to treat different supplier groups differently. This can just be done on the basis of detailed knowledge. It must be stated that many shipyards do not have this knowledge and therefore cannot launch appropriate measures respectively cannot control the impact of their respective investments.

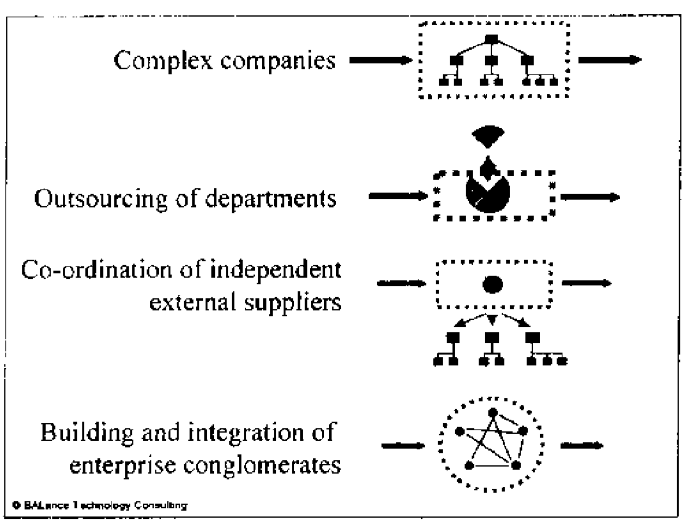
4 Actual Trends and Strategies

Beyond outsourcing strong tendencies towards more global sourcing of materials, more system related purchasing (reducing the supplier base!) and more collaborative engineering can be identified within the major shipbuilding groups world-wide. In spite of high potentials for cost reduction in these strategies, there are risks, which may jeopardise the anticipated success. Major concerns are with the delivered quality of supplies, the timely delivery of ordered materials and components and, to come in control of the entire process, the need for new and advanced management skills in combination with a better information and communication system environment.

Benchmarking studies in the shipbuilding industry for comparable ship types show margins of 15 % and more between prices for external material cost. The shipyards have started to realise those margins by putting more price-pressure on the suppliers, but also by the standardisation of technical solutions, teaming-up with other shipyards to achieve stronger buyer-power and by reducing their supplier base through more system related enquiries.

On the other hand these trends in general require also good strategic concepts from the marine equipment manufacturers for the future. Since the trend towards outsourcing in the first place effects the configuration of strategic alliances with marine equipment suppliers in geographically close regions, the building of global co-operation alliances with some material/component key-suppliers and service-providers may be more adequate. The general step-by-step process of outsourcing through shipyards is shown in Figure 6.

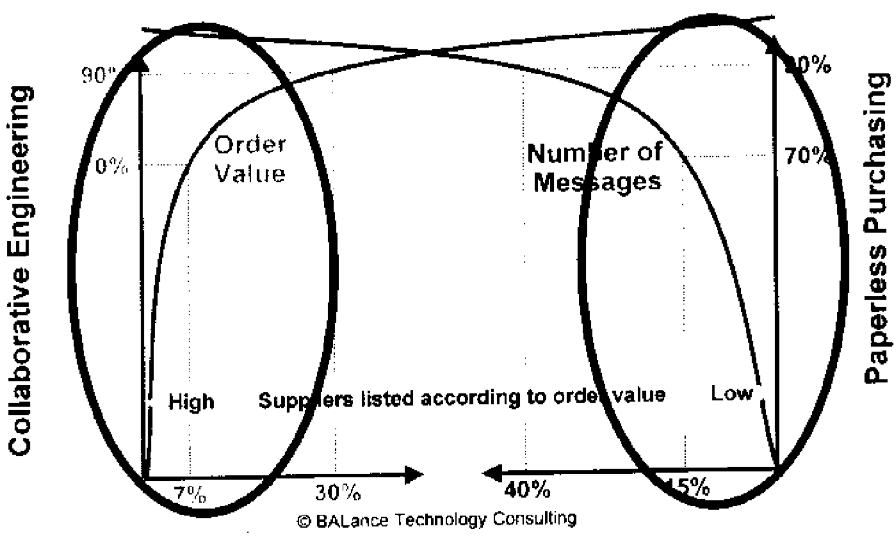
Figure 6. Migration towards a virtual enterprise



Beneath the effect to achieve commercial benefits through the outsourcing process itself, e.g. through higher workloads and relatively lower overhead costs, the availability of new and advanced information and communication technology allows the cost-effective realisation of those concepts. As an ultimate consequence shipyards, theirs suppliers and others integrated into the process may migrate towards a Virtual Enterprise (VE). A Virtual Enterprise by definition is a set of temporary linked individual companies clustered to fulfil a timely restricted business process and behaving for that business process as an integrated enterprise enabled by advanced technologies.

On the basis of the individual supplier analysis of a shipyard it is essential to apply the right strategy with the right partners (marine equipment suppliers) and to assign the right technology for the implementation of co-operative working solutions. An analysis of the supplier base as shown earlier (Figure 5) can be used to make appropriate decisions. From this analysis indicators and strategic decisions can be derived with which partners technical solutions towards paperless purchasing and/or collaborative engineering should be developed on high priority. Those suppliers of shipyards which provide high value components with a low amount of purchasing documents, but a high amount of technical data to be exchanged must be considered for collaborative engineering solutions. Those who supply "bulk materials" or standard products which require frequent and periodic exchange of purchasing documents, may be better considered for paperless purchasing procedures (Figure 7).

Figure 7. Typical Supplier Analysis and Recommended Co-operation Fields



5 New Technologies and Management Practice

New technologies, basically information and communication technologies, tremendously effect the way of working within and between companies. Since this is true for almost 40 years now, the shipbuilding industry seems still to be at the starting point for an organisational revolution caused by this. The given background of newly available technology, e.g. internet, EDI, geographically distributed client server environments etc., allow far reaching concepts for integrated supply chain management for shipbuilding applications. Principle examples for this can be found in other industries, e.g. just in time delivery solutions in automotive and aircraft industries.

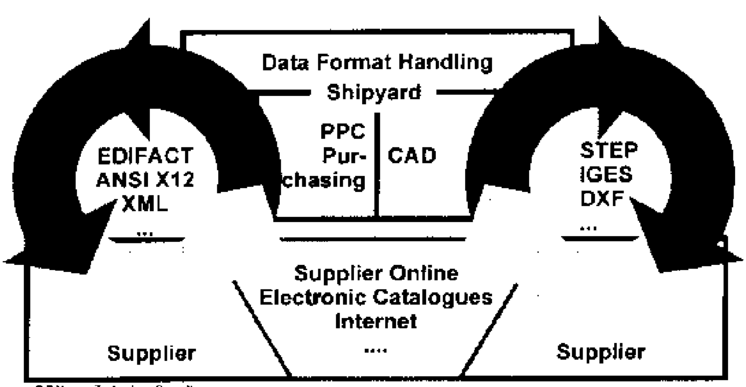
The technology to be applied is very often described by the name "Virtual Enterprise Technology" (VET). VET is anticipated to play an increasing role in the support of emerging collaborative networks. Virtual enterprises are defined as a set of temporary linked individual companies clustered to fulfil a timely restricted business process and behaving for that business process as a integrated enterprise enabled by advanced information and communication technology. For example a shipyard and a set of key suppliers team-up to fulfil a ship newbuilding contract. Virtual enterprises can be build-up for different applications, e.g. supply chains, design chains, distributed manufacturing and assembly processes. Because of their temporarily nature and a heterogeneous portfolio of participating companies the mechanisms applied need to be flexible, standardised and easy to reconfigure. An open communication infrastructure, openness of software systems, agreed data exchange standards, harmonised workflow and a dynamic intra- and interorganisational reorganisation process are vital success parameters for VEs.

To achieve the expected benefits, it is further essential that the shipyards and their suppliers build up new and powerful managerial skills combined with continuous qualification programmes for their employees.

In the last ten years the European shipbuilding industry including also suppliers to some extend started to prepare the baseline for this kind of future working. Within individual projects international standards for the exchange of data have been always in focus. For example, major contributions have been made to the international discussion and development of maritime application protocols of the ISO standard STEP and to the definition of industrial specific message types according to the UN standard EDIFACT. Through the establishment of EMSA (European Maritime STEP Association) in 1994, Special Interest Groups for Networks, Electronic Data Interchange (EDI) and Product Data Management (PDM) and many bilateral international co-operations the industry achieved a better understanding for the needs and benefits of co-operative working. The industry is now at the point to show an increasing interest for putting VE-technology into operation.

Mainstreams for collaborative electronic based working is highlighted in Figure 8. Most likely a hybrid solution combining the advantages of different technologies will be implemented for individual business process needs. The major challenge is to find the right configuration by using all enabling technologies and to keep the application flexible for fast adaptation according to a fast changing technological platform.

Figure 8. Mainstreams in EDI



The problem is that the structure and the level of technological development may be extremely different with the co-operating partners. The portfolio ranges from fully developed and computerised companies which also apply appropriate management abilities to the change of the processes to those companies who have hardly applied any computerised solutions for their own process handling. The shipyards have to think about the right way to build up solutions which allow most of these companies to be integrated into advanced co-operative working concepts. The way as chosen by many other industries to just put enough pressure on suppliers and force them into solutions which are favourable for the customer but complicated, inadequate and expensive for supplier cannot be applied by the shipyards. This is because the shipyards are a very heterogeneous group of companies itself which does not create enough market-power. Even co-operational agreements between bigger shipbuilding groups in Europe are not consequently used to create and us this market-power. Further, the supplier base is comparable big and consist of many small and medium sized enterprises which are by far not prepared for advanced working concepts.

New very promising technologies which have been developed for Internet applications may help to overcome the old problem of too high cost for the application of classical EDI solutions to small and medium sized enterprises. As a result it can be stated that the availability of technology for all level applications is as such that they almost offer solutions also for the incorporation of these companies. Different projects performed under the framework of European support programmes etc. have substantially contributed to some of these developments. MARVEL OUS (Maritime Virtual Enterprise Linkage - Open User Syndicate) draw a baseline to the situation of standards for Maritime Virtual Enterprises and edited a basic book on standards. EDIMAR (Electronic Data Interchange for the European Maritime Industry) developed some adapted EDIFACT messages to the need of maritime purchasing applications, contributed to the definition of STEP AP 226 and developed and adapted workflow tools and concepts to shipbuilding purchasing applications. The new project MARIFLOW (A Workflow Management System for the Maritime Industry) is now working on workflow applications to quality data chains including applications of EDI functionality for the exchange of quality certificates for steel plates. In these projects and more others functionality and potential of different technologies and standards have been proved by setting up demonstration networks and scenarios covering Computer Supported Co-operative Work (CSCW), Workflow Management Systems, Classical EDI and Extensible EDI (XDI) concepts by means of internet technology. It is now about time to continue working on these platforms and to create numerous and manifold reference applications to verify and develop commercial benefits and to create new starting points for further developments.

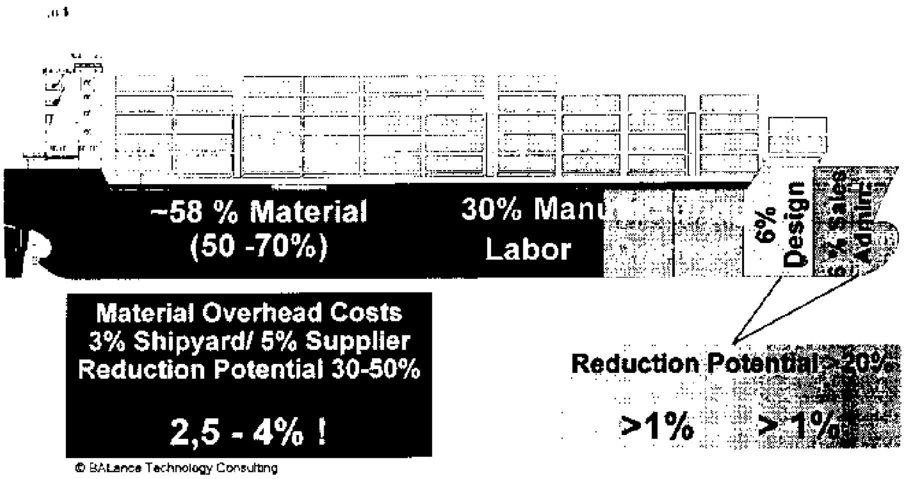
6 Commercial Potential

As said above the cost involved in the complex functions of supply chain management are substantial and very much worth to seriously think about an improvement. Nevertheless these cost are very often hidden in other costs and therefore, often not easily to identify or to separate. Overhead cost by nature the potential for cost reduction is difficult to prove and just be approached through consequent managerial efforts and cross-departmental thinking and re-organisation. However, a considerable overall cost-reduction for ship newbuildings can be anticipated by the application of new technologies and new ways of collaborative working if the technology is seriously taken and consequently applied.

The commercial benefits maybe generated at both ends of the collaboration, the shipyards and the suppliers. In comparison to investments in the manufacturing area the risk ratio (ratio of investment to potential earnings) is lower, but, to say this again, needs more managerial involvement and consequent reorganisation. The benefits through the application of advanced Electronic Commerce and EDI technology in the different processes of the supply chain may sum up to about 5%

of the total cost of the ship, which is a considerable share. The biggest share can be realised through decreasing material overhead cost at the shipyard and at the suppliers. However major achievements can be expected in the design area as well as in the area of Sales and Administration (Figure 9).

Figure 9. Potential for cost reduction

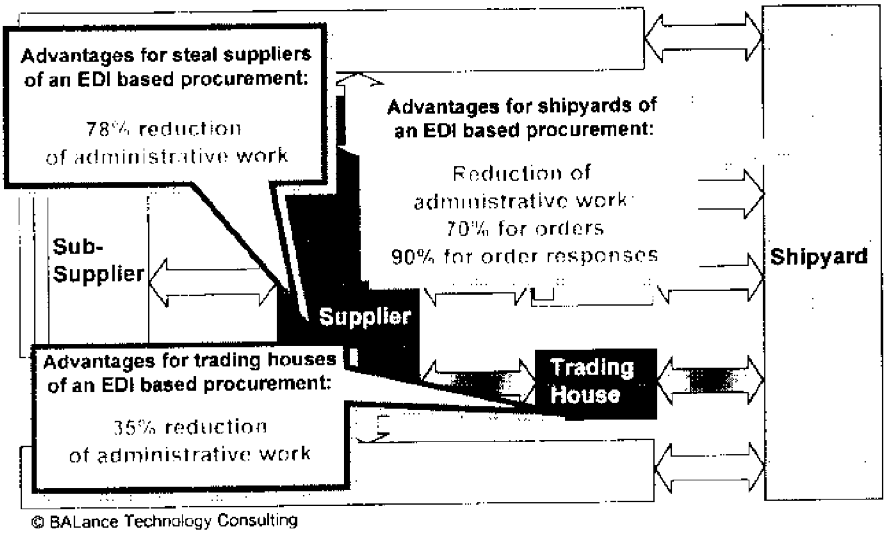


At the supplier side some more effects can be generated through the use of advanced multimedia marketing instruments, i.e. electronic catalogues with technical information elements either on CD-ROM and/or through suitable Internet representations. This may include product descriptions in standard formats which are ready to build in for the designer in his product model. Further the consequent building of organisational interfaces towards the shipyards including building the ability for the structured exchange of data by means of EDI can also create very positive effects for the opening of new markets and the maintenance of existing one's.

Potential effects at the suppliers are the building of a special differential advantage in the market, possibilities for direct marketing (without regional sales agents), a faster penetration of the market with new products, a faster/cheaper change of the marketing strategy and a closer link to the customer once the new links have been positively established

Pilot installation and demonstration cases have shown that the anticipated targets for cost reductions could be achieved easily. This is also confirmed by some comparable applications in other industries (Figure 10). However, the full benefit of the investment can just be achieved if new and EC/EDI based shipyard/supplier relations can be build up fast and consequently so that as many suppliers as possible can be linked in a short time frame. Risks can be minimised if shipyards and suppliers in the starting phase do not invest in own systems, but use commercial data clearing services (e.g. BAL.DIS the BALance Data Integration Service). As long as the number of interlinked suppliers or other co-operational partners is low or the amount of data to be transformed is limited those commercial services offer the economical solution. By developing more co-operational links one or all partners can successively migrate into an own system which can be slowly build up in parallel to an already working solution.

Figure 10. Benefits through EDI based Steel purchasing.



Other benefits have been proved through tremendous time savings during the entire purchasing process, but also in other follow-up processes. Better data accuracy, less mistakes through data coding and a better basis for decision making processes have been reported to be other major achievements which can be forecasted, but just proved through the consequent application and use of new and advanced technology and management methodologies. There is no lack in technology, even if for many special applications suitable solutions still need to be developed. But for initial earnings the technology is ready for application.

7 Conclusions

The different and manifold business processes with maritime supply chains create a substantial amount of cost in the entire shipbuilding process. Emerging advanced information and communication technology and respective developments from recently performed R&D projects allow the consequent attacking of those cost with a high potential for commercial achievements. Especially the reorganisation of the purchasing process and the logistical process between shipyards and their suppliers through the application of Electronic Data Interchange and Electronic Commerce Technology show promising potentials. Very actual new developments in the area of Internet programming languages (XML) which are allowing also the exchange of structured data on this basis are very promising to also overcome cost problems of classical EDI for small companies, which do not even run own purchasing systems. Nevertheless, all applications in this field have to consider the heterogeneous structure of the shipyard's supplier base. Therefore all solutions needs to be flexible to the outside, but streamlined to the inside. The problem to start respective applications in shipyards is not a problem of the availability of technology. A management decision has to be taken, pilot applications have to be implemented and necessary re-organisation and qualification programmes have to be started in parallel to the configuration of the technical solution. Once the technology is in place and the organisation has learned to handle it the potential to quickly earn commercial benefits is very high.

8 References

1. Ahlers, R. et al (1999), "EDIMAR - Electronic Data Interchange for the European Maritime Industry", *Internal Project Report No. 2 Esprit Project EDIMAR*, January 1999
2. Ahlers, R. (1997), "Installation Method for Electronic Data Interchange in the Maritime Industry", *Proceedings of the 9th International Conference on Computer Applications in Shipbuilding (ICCAS 97)*, October 1997, Yokohama, Japan
3. Brodda, J., Ahlers, R. (1995), "Effective use of EDI, including MARIS programme relationships and interactive partnerships between shipyards and their suppliers", *Proceedings of the CESA (Committee of European Shipbuilders' Association) Conference "European Shipbuilders World Class Performance through new Partnerships with Suppliers"*, November 1995, Brussels, Belgium.
4. Buxmann, P. (1999), "The Future of EDI – XML as a basis for building bilateral business processes", *Brochure of Competence Center XML*, University Frankfurt, Germany 1999
5. Kendall, J. (1998), "Standards and Protocols for the Realisation of Virtual Enterprises in the Maritime Industry", *Public Document Esprit Project MARVEL OUS*, January 1998
6. Özel F. (1996), "Analysis of Cost-Benefit through the Application of Electronic Data Interchange in Shipbuilding", *Master Thesis at the University of Bremen*, Germany 1996

ELECTRONIC COMMERCE

Than N. Lam, NSWC, Ship Systems Engineering Station, USA

Introduction

The word "Commerce" is combined by the word *com* (together) and merx (merchandise). The Random House College Dictionary defines that "Commerce" is the buying and selling of goods. The world "Electronic" pertains to devices, or systems developed through electronics. Electronic Commerce is interpreted differently by different people. The first thing to understand is Electronic Commerce covers any form of computerized buying and selling, both by consumers and from company to company. It is viewed by many to be the technological revolution of the late 1990's and early 21st century that will likely transform the operations, systems and efficiency of business, industry, and government.

Here is the other useful description:

The Automotive Industry Action Group in North America defines it as "the enablement of a business vision supported by advanced information technology to improve efficiency and effectiveness throughout the trading process."

History

The term Electronic Commerce has really emerged in the last ten years, and can be regarded as a broadening of the term Electronic Data Interchange (EDI). However, there have been links between computers since the 1960's, so the routes of electronic trading lie much further back even than when EDI became a buzzword during the 1980's.

Categorization

With modern telecommunications infrastructures, we have been provided with the means to exchange data instantaneously. Electronic Commerce is about using these data-flows in the most effective way. There are commonly three phases of implementation of Electronic Commerce:

- Replace the manual and paper-based operations with electronic alternatives
- Rethinking and simplifying the information flows
- Using the information flows in the new dynamic ways

Simply replacing the existing paper-based systems will reap few real benefits. It may reduce administration costs and improve the level of accuracy in exchange data, but it does not address the questions of whether a business is operating efficiently. Electronic Commerce applications can help to reshape the ways to do business and have often acted as a catalyst for reengineering.

Reengineering is about reassessing the ways to conduct business. It is about looking at the flows of information through production processes. It is about optimizing the use of resources. Thus, it is necessary to understand the electronic commerce applications available in order to think about how they can best assist the existing system. Electronic commerce applications:

- EDI, Hybrid EDI, Interactive EDI
- Email
- Internet Applications

- Enhanced Fax
- Voice Applications
- CALS
- File Transfer
- Computer Aided Design/Manufacturing
- Multimedia
- Bulletin Boards
- Teleconferencing
- Automatic Identification

Significance of E-Commerce

Electronic commerce is important because it has the potential to advance the global economy by changing business styles. For business to survive the Information Revolution, business must take advantage of electronic commerce.

In a souvenir shop in San Francisco, there are many T-shirts of the Golden Gate Bridge, Cable Cars, Fisherman's Wharf, etc. Most of those T-shirts are made in China at a very low cost but are sold at about five dollars or more. In an electronic commerce network were established between T-shirt factories in China and San Francisco. Chinese T-shirt makers would be able to get orders directly from tourists by simply displaying their catalog on the Internet.

With those simple ideas in mind, you may understand the impact of electronic commerce. As long as you have a good product and a competitive price, you don't need to travel across the oceans to find retailers and to create marketing networks.

Example of Electric Commerce - Amazon.com

Amazon.com is reportedly the Internet's largest bookstore. Customer accounts rose from 610 on June 30, 1997 to 3.1 million as of June 30, 1998. The product line was recently expanded to include music (June '98) and has also expanded into the European market by its purchase in April of Bookpages, Ltd., Telebook Inc. and Internet Movie Database, Ltd.

Most recently in July, 1998, Amazon.com announced an agreement to participate in ModaCDA's Virtual Shopping Mall e-business fashion solution. ModaCAD's CEO commented that their virtual shopping experience would not be complete without the presence of an established retailer of books and music.

Amazon.com is considered a pioneer in the electronic commerce marketplace. For this news release from Amazon.com, the following comments were made regarding the agreement which identify the current considerations for e-commerce.

Factors that could actual results to differ materially include the following:

- the timely completion of the development of the company's software products, including ModaCAD's e-commerce and consumer software and enhanced or updated versions of the company's electronic merchandising and CAD software products
- unforeseen technical or other obstacles in the development or production of such software

- acceptance of ModaCAD's e-commerce and consumer software by the publisher/distributors of such software and the release and marketing plans of the publisher/distributor
- customer acceptance of ModaCAD's e-commerce and consumer software products and updated or revised versions of the company's electronic merchandising and CAD software products
- the company and its publisher/distributor's ability to produce its products on a cost-effective and timely basis
- factors not directly related to the company, such as competitive pressures on pricing, market conditions in general, competition, technological progression, product obsolescence and the changing needs of potential customers as well as the software, textile, apparel, home furnishings and home design industries in general

The announcement also contains the following statements regarding Amazon.com's concerns for risk and uncertainties that include:

- Amazon.com's limited operating history
- The unpredictability of its future revenues
- Risks associated with capacity constraints, management of growth and new business opportunities.

Strengths of E-Commerce

E-commerce is another step closer to a goal that has started with grocery stores that stay open 24 hours 7 days a week: the quest to be able to get everything, anything and from anywhere in the world. The electronic marketplace never shuts down and the number of on-line available goods is ever increasing. This makes on-line shopping very convenient: no need to leave the house, no need to think of opening times, no need to remember you can buy what.

In addition, companies can set up personal profiles of on-line shoppers by tracking their moves on their web sites and create personalized offers based on previous customer preferences.

Interactively, the ability of the buyer to enter some form of dialogue (e-mail, questionnaires, etc.) with the vendor is another advantage of Internet based trading over traditional long-distance trading (e.g. mail order catalogues). This way vendors and manufacturers get valuable direct feedback from their customers.

Virtual stores can replace bricks-and-mortar stores and thereby save costs. This in return allows small companies to compete effectively with bigger corporations without the need to set up a wide system of stores and expensive logistic and distribution systems. E.g., Amazon.com completely changed the way the people can buy books. Being a new company it can effectively compete against old timers like Borders and Barnes & Noble because of the E-Commerce on Internet/WWW. The old paradigm "all business is local" does not hold true anymore. Buyers/sellers are no longer geographically confined.

Finally, E-commerce reduces costs by eliminating traditional brokers or middlemen, thereby shortening the distribution chain. The redesign of business processes forced through the new technology save money due to the greater effectiveness of the way business is done.

Weaknesses of E-Commerce

E-commerce still has a few weaknesses, too, that shouldn't be ignored:

From a buyer's point of view, the Internet offers an overload of information that can be difficult to sort through, especially for the inexperienced user. Web browsers provide a good deal of help, but even with these the hit of 214 sites containing the searched term can be overwhelming. Therefore the problem of finding the right site, to navigate in the vast amount of information can pose a problem. Another weakness is still the unsolved security issue (see also next paragraph) that results in a low trust level when it comes to financial transactions and the exchange of credit card information, for example.

From a vendor's point of view, the Internet is a difficult place to grab and hold a customer's attention, since it is so easy for the customer to just click to the next web page. For the same reason it is difficult to build and keep up a brand loyalty as it is known from traditional retail. Also, the fact that buyers can easily find the lowest prices for a specific product forces the vendors to reduce their profit margins in order to stay competitive. Lastly, depending on the kind of business there can be rather high initial investments for hardware and software in a cyberspace business. High up-front costs combined with low profit margins can therefore pose significant hurdles for market entrance.

Current Limitations

- *Security*
    1. Customer to Web server security i.e. Web browser to Web server security. This is currently being addressed by various new technologies like SSL, SET and encryption but this technology is still evolving and takes some time to mature.
    2. Web Server to Merchant's Desktop security. Most small businesses don't have their own web server; they rent web space from an Internet service provider, and those who want to sell directly on Internet, rent "secure" web space (that is, equipped with SSL) so customers can communicate securely with their Web server.

- *Bandwidth and latency limitations due to various factors including technology*

    When the Internet was created 25 years ago, it was called the ARPANET and was used primarily by U.S. researchers and scholars for file transfer and E-mail. The bandwidth (measure of capacity of data transfer mediums) deployed from 9.6 Kbps to 56 Kbps, which was sufficient to support activities at that time. Today, individual and corporate users are flooding onto the Internet via the WWW in record numbers. These users demand higher bandwidth-consuming technologies (such as multimedia and voice) to perform electronic commerce. This usage is causing serious bandwidth and latency problems.

Technical Challenges to Address

- *Security*

    A widely held belief is that security issues continue to hinder the growth of electronic commerce over the World Wide Web. There are a number of technological and legislative efforts underway to ensure that adequate security measure exists for these Web-based transactions that include

    * Encryption
    * New communications technology like secure sockets layer (SSL), and Secure Electronics Transactions (SET)
    * Digital certificates/Authentication

- *Bandwidth and Latency*

  This problem can effectively be addressed by using various new technologies like

  - Data compression to reduce the amount of data to be transferred
  - New hardware technologies like TV cable modem, fiber optics and satellite links
  - Better packing routing technology like Resource ReSerVation Protocol (RSVP) and IP version 6 (Ipv6)

- *Connectivity*

  The currently dominating media TV and Radio will eventually merge with the PC; not necessarily in the content offered (entertainment on the TV, information of the PC, a mix of both on the Radio), but at least in the point of access. One device will be used to access video, audio and text-based information and entertainment. This in return will allow for improved transmission of information and even easier access. In order to get their current applications will have to be easier to connect and "Plug 'n Play" has to grow from a marketing slogan to reality.

Future Trends

- *Improved market information for sellers*

  New technology will facilitate expansion of rapidly growing database marketing trends. This expansion will also be fueled by developments like more extensive research, which will enable better understanding of consumer behavior and coordination of old and new media, which will enhance an organization's ability to communicate with consumers, the ultimate results will be one-to-one marketing.

  With implementation of new technology future marketers can instantaneously exchange information and can assess effectiveness of marketing strategies, particularly in promotions and pricing, which is critical to maintaining a competitive edge.

  Sellers will make decisions based on individual or household demographics, responses to promotions and brand loyalty. Marketers who master this more complicated environment as a result of the enormous amount of data available, will be rewarded with customer loyalty based on increased customer satisfaction.

- *Emergence of New market Intermediaries like information warehouse*

  With the dramatic increase in information, new market intermediaries like information warehouses will emerge, which begin by providing storage capabilities and later evolve into entities which integrate massive amounts of data and sell them to organizations to enhance their marketing efforts. A parallel effort will be the emergence of intelligent hardware and software like the search engines which rely on artificial intelligence that go beyond current notation of keywords. For example, a search for the words "Casual apparel" will reveal products related to casual without the word "casual" appearing in the title, i.e., the search engines know that products associated with certain activities are considered casual.

  Market intermediaries will address the fragmentation of formerly integrated selling and buying functions. For example, electronic marketers selling at the retail level. Once the sale is made electronically, they can place an order directly with the supplier who will ship directly to the customer. This specialty technology-based service is that of a broker having no person-to-person contacts, nor store atmosphere. This will gain importance at the retail level as selling is separated from other retail functions.

- *Expanded and more technology-based channels of physical distribution*
  Physical distribution through electronic channels will become widespread. Electronic channels will distribute any kind of product or service that can be converted to a digital format and transmitted either through fiber optic cable or satellites. These products may include movies, music, newspapers, magazines and books, money, tickets, stage plays or any other forms of entertainment or market research information. Thus products that can be digitized will be available in various formats from a variety of sources.

- *Additional technology-based consumer services*
  Time management difficulties will continue as customers seek more control over their lives, both personal and professional. This will be particularly true at the retail level as baby boomers move into their 50s and search for ways to simplify their lives. Obvious services like bill paying, time and activity organizers, callback reminders, "To Do" lists, trip schedulers and so forth will emerge first. But others, such as food purchases or recommendations on product/service purchases on the Internet will not be far behind.

- *More worldwide sourcing*
  National boundaries will fall even faster than they have in recent years. Technology will enable individuals and groups to bypass existing sources of products and services and choose those that most closely meet their needs. Competition will be facilitated not only by the Internet, but by other technologies like ISDN, HDTV and interactive voice response. Some traditional channel distribution will disappear, whereas those that provide service through the Internet will be most in tune with customer needs and desires.

- *Further price reductions on hardware side*
  The continuing decline of prices for computer hardware will further increase the availability of communication technology in the form of PCs, Laptops, Personal Assistants, and will also support the trend towards miniaturization. Smaller and cheaper devices will be used by a wider audience for a wider array of tasks and therefore, further support the rapid growth of e-commerce.

PRACTICAL USE OF ELECTRONIC COMMERCE – CASE STUDIES

Case One: Building a Customer Web Site

The main objective is to redesign business processes to simplify information flow, and replace manual and paper-based operations with a cheaper and faster electronic alternative. Cost effectiveness through the reduction of personal contact based transactions can be re-directed to other business objectives. (See Appendix A presentation slides).

Background

The Clinical Affairs department at HBOC established a goal to provide the customer base (600+) with clinical documentation on the Web in 1998. Additionally, a large, influential client suggested that the company provide the top 100 client questions with solutions on a Website.

The Clinical Affairs department has two offices (Cambridge, MA and Malvern, PA) representing the HBOC business unit - Payor Solutions Group. The Corporate office is in Atlanta, GA. The targeted information for this initiative are MS Word documents which describe clinical logic for edits within the knowledge base/product. The documents are currently provided to clients by Internet email or PC fax.

The Process

The Clinical Affairs staff must respond within 48 hours with a solution for 70% of the questions submitted. Clients rely on the company's ability to respond to these questions on time since reimbursement to physicians is often at stake. The remaining questions are answered within 10 working days. Meeting this response time is not related to reimbursement directly, but does affect client internal policies regarding reimbursement to physicians. Any delays experienced with this process create customer satisfaction issues.

Clients typically call the Client service organization Call Coordinators who route both product and clinical questions to Client Service Consultants. This group attempts to answer clinical questions for clients based on their knowledge of the product and available client training manuals. While some clinical questions can be answered using manuals, the majority requires further review by the Clinical Staff.

The call tracking system is the front end to this process. It serves as both a client call tracking system and a client invoicing system. It contains a database of customer information which includes questions asked about products. Clients are able to access the call tracking system to check the progress of questions. This database is centrally managed at the corporate office.

Business units do limited customization of the call tracking database. Each unit is expected to import data from the system to another database (e.g. Access), and supplement it with data and reporting relevant to business unit needs. Often, a business unit will duplicate the information from the call tracking system in order to report metrics for client and departmental activities.

Clinical Responses

Clinical questions arrive in the Clinical Affairs group once routed through the call tracking system (annual volume 2000 - 2500). In order to assure that information is captured on the client and the date a question is received and tracked through completion, the Clinical Affairs group duplicates the data entry of key customer data elements captured in the call tracking system. This information is routed to MS Excel for reporting purposes (the call tracking system does not provide reports specific to departmental processing).

The status of the question is monitored in the call tracking system which must be updated by every user assigned to the call. Clinical responses are compiled in MS Word documents, stored by client name, reviewed for quality by the clinical group, and then either faxed or emailed to the client. The call tracking system is updated to indicate the call is closed.

A historical file of clinical responses is stored in the clinical affairs directory and researched when a question is submitted. More than 40% of the calls received can be answered with history file responses which are cross-referenced in the call tracking system, but are not available to clients.

Issues

- Re-creating the call tracking database and clinical responses (i.e., those in the history file) is inefficient and causes delays in the overall processing of clinical responses.
- These inefficiencies cause customer dissatisfaction with respect to timely turnaround for solution.

Infrastructure

- The company has an Internet Web site and customers have capability to access product information (see Appendix B. HBOC Organization/Departments Home Page, HBOC Intralink Home Page Product Portfolio, EC200 Product Home Page re electronic commerce, and Customers Only Home Page).
- Remote offices are client server with communications to and through Atlanta.
- The company is moving to MS Outlook from Lotus' CCMail, and for Intranet applications, from Lotus Notes to MS Office (see attached Monthly CIO Newsletter).
- Current technical support operations are problematic for normal email communications across remote sites (see attached IS Home Page).
- Internet communications go through Atlanta and firewalls are in place, as needed.
- Traffic on pipeline is heavy
- Infrastructure enhancements by the Information Systems group are in the works to improve timeliness of these communications.

The company is well positioned to provide support to the Clinical Affairs group for the development of a Website to house a database of clinical responses/solutions to meet client needs.

Departments Involved

This initiative is spear headed by the Clinical Affairs group with representation from key departments to support the design and implementation of the Website and to define future enhancements.

- Clinical Affairs - Driving the business need based on client feedback.
- Client Services - Interface with client base.
- Corporate Information Systems - Central developers of the Website.
- Marketing - Group responsible for liaison to corporate for Website development.
- Technical Support - Group located in Malvern and Cambridge serving as liaisons to the corporate office (Atlanta).
- Product Management - Driving product direction with Clinical Affairs to meet client expectations.
- Database Tools - Support to Clinical Affairs and link to corporate regarding development of the clinical response database for the Website.

Procedural Issues

This initiative requires a review of current processes that will be factored in to the design of the Website. The clinical Affairs directive is to provide a Website with a meaningful set of client solutions. Review and analysis needs to consider the scope of this effort so that clients will not be dissatisfied with the result; i.e. KISS.

Some procedural issues to consider include:

- Text documents - how will these documents be converted to a database for the Website? What is the company standard?
- Managing the Website and current call turnaround times
- Integrating business systems (i.e., call tracking) into the process
- Resource constraints (system and human)
- Coordination of Malvern and Cambridge offices
- Update and Maintenance of text files
- Replacement of current system or is this an adjunct to the existing process; i.e., replace only the distribution process (fax and email)
- Priority of this project over others
- Longer term objectives of Clinical Affairs and other departments

Technology Issues

The following issues must be considered as processes are re-engineered:

- Dynamic links
- Updates and maintenance
- ORACLE database structure and MS Word
- Store text in HTML
- Search Engine
- Object Linking and Embedding (OLE)
- Client readiness to access the Internet (i.e., dumb terminals or PC based)
- Acquisition of resources to support the initiative
- Current sluggishness of email and Internet communications
- Acquisition of hardware or new software
- Testing
- New technology with uncertainty of expert resources

Security and Confidentiality

The company is diligent about maintaining security for its Intranet and Internet Home Pages. Security remains one of the biggest concerns for Internet users in general. The concerns that are most pressing with the creation of the clinical Website component include:

- Company information accessible to competitors (indirectly through existing clients)
- Defining customer level of access
- Overall uncertainty of Internet security

Strengths of the Initiative

- The development of the clinical response system Website will reduce labor intensive processes that grow more complicated with increased client volume.

- The clients can access clinical responses by accessing the Web in lieu of making phone calls and being routed through two (2) customer service staff members.
- The clients will experience timelier processing for their needs.
- Internal resources can be re-directed to other department/company objectives.
- Current delays in faxing or emailing clinical responses caused by "pipeline" traffic will be reduced.
- Provides the foundation/experience for future technology alternatives.
- Saves human resource dollars as process improvements occur.
- Eliminates monthly mailing of client newsletters.
- All clients benefit from the initiative vs. Those who typically request more solutions (statistics show that 12 clients in the Malvern client list ask the majority of questions year after year).

Limitations of the Initiative

- Security and confidentiality of company information remains a concern.
- Communication issues associated with decentralized offices.
- Staff training and education
- Availability of the technology staff

Case Two: Vendor Managed Inventory via EDI

Vendor managed inventory (VMI) is when the vendor (or supplier) provides not only their products, but also performs the inventory management of those products for the customer, deciding which products need to be replenished, the quantities, and at what time. VMI focuses on the manufacturer taking responsibility for the customer's inventory levels.

Vendor managed inventory makes good business sense and presents a tremendous opportunity. A supply chain partnership between trading partners normally has two driving objectives, increase sales and reduce operating expenses in the supply chain. VMI can contribute to both of these goals. With the customer providing information, and the vendor performing the inventory planning, both parties together can increase supply reliability and reduce supply costs.

Electronic Data Interchange (EDI) is used to transmit structured data between computer applications of business partners. The mechanism for the delivery of the data depends on which method the business partners decide to use. Computers can be linked directly, via modems and phone lines, or through a private network. The most common communication method for EDI is to use a private network, or Value Added Network (VAN) to carry the data. VAN's are run by third party network operators. They provide a "store and retrieve" service for business trading partners. Each user has a mailbox on the VAN so that they can retrieve and process messages in a batch mode. VAN's offer added values such as a high level of security, a help desk when problems arise, an audit trail so that the sender knows when a message has been received and read. They also support most communication protocols and data formats.

In 1979, the American National Standards Institute (ANSI) chartered the Accredited Standards Committee (ASC) X12 to develop uniform standards for the use of EDI transactions. Committee members develop and promote EDI standards that streamline business transactions. X12 standards facilitate these transactions by establishing a common, uniform business language for computers to communicate around the world. Currently there are more than 275 transaction sets, that are used to communicate business to business operations.

The VMI process (attachment C) uses transaction sets to transmit information to and from the customer and the vendor via EDI. The customer sends product activity (warehouse usage and withdrawals) data (852) to the vendor on a daily basis. This information contains the customer's current stock keeping unit (SKU) inventory, as well as any SKU inventory on order (currently in-transit). This product activity data is downloaded into a VMI software application such as a demand planning tool. Also transmitted to the vendor to anticipate the expected demand by product, as well as any deals or promotion the customer may be planning in the future. This product forecast information can be provided in weekly, monthly, or quarterly buckets. This is also downloaded to the VMI software application demand planning tool. This information is then used by the vendor in the demand planning tool (as time-phased DRP) to determine when it is necessary to plan and ship inventory replenishment orders to the customer's distribution centers. Once the demand planning tool determines that a replenishment order is needed, it creates a planned purchased order. A planned PO acknowledgment transaction (855) is sent from the vendor to the customer, to advise them of an expected PO. The actual planned order is sent to the vendor's order management system as an order open to ship. Once the order is picked, packed, and shipped by the vendor, another EDI transaction is sent to the customer. It is called the ship notice/manifest (856). This transaction advises the customer of the contents of the shipment, as well as product description, packaging marking, and carrier information. This information is used by the customer's receiving location. Lastly, the vendor sends the invoice transaction (810) to the customer for payment via EDI.

Electronic Commerce embraces all sorts of data, however, it does not possess the level of definition that EDI has. In fact, EDI has taken on new life within the broader framework of Electronic Commerce, and there are many benefits that can be achieved by using EDI in an organization.

Summary

Electronic Commerce as a concept is not something new, but just recently caught on fire with the emergence of World Wide Web. World Wide Web, with its easy-to-use browser interface, made interaction over electronic network (the Internet in this case) as practical and easy as a click on the mouse button.

This paper took a comprehensive approach to examine the evolution of the Electronic Commerce. We first looked at it from its definition and its history in order to understand where the concept came from and how it developed over time to become today's phenomenon. We then analyzed the significant role that Electronic Commerce was playing in today's world and how it has changed our lives and the world around us.

Secondly, we investigated several key components of Electronic Commerce, particularly in the context of the Internet and World Wide Web. This includes the technical solutions and infrastructure required, strengths and weakness comparing to conventional trading methods, and limitations of Electronic Commerce. Furthermore, we also looked into the future and gave our best shot at predicting where the Electronic Commerce was going.

Finally, we took two real-world examples to show how the Electronic Commerce were being used in today's organizations and how it had helped them to gain competitive advantages.

References

1. Johnson, G., "Vendor Managed Inventory," APICS The Performance Advantage, June 1998, pg. 30-31
2. Kulwiec, R., "Vendor Managed Inventory," Modern Materials Handling, October 1997, pg. 13-14
3. World Wide Web, "http://www.harbinger.com/resource/X12"
4. Associated Press, "Amazon.com Sales Soar to $116M," July 23, 1998, http://biz.yahoo.com/
5. Business Wire, "Amazon.com Anchors ModaCAD's Virtual Shopping Mall," July 27, 1998, http://biz.yahoo.com/

ELECTRONIC COMMERCE FOR SHIPBUILDING SUPPLY CHAINS

Thomas Gulledge, Ronil Hira, John Liuzzi, and Rainer Sommer
George Mason University, Fairfax, VA

Introduction

Years ago the international business community embraced the Electronic Data Interchange (EDI)[1] as the "wave of the future." The prospect of electronically transferring large volumes of data appealed to many companies, for it would not only allow them to gain a competitive advantage by speeding up the delivery of business transactions, but also decrease IT expenditures by standardizing their information systems on the ANSI X.12 and the UN EDIFACT transaction formats. In theory, EDI was going to change the way companies interacted[2]; data was to flow unencumbered between integrated systems and across international borders. The reality however was somewhat different.

By the 1990s, many of the early EDI proponents had to admit that the exuberance demonstrated during that earlier time was somewhat optimistic. Although EDI had made significant inroads in some large high-volume businesses, the implementation had not proven a great benefit to many small-and medium-sized enterprises (SMEs). We were aware of the problem primarily from our many interactions with SMEs through our U S. government-sponsored electronic commerce center, but the problem was also being discussed widely in trade publications [see, for example, Mann (1996) and Smith (1996)].

The emerging opinion is that the benefits of traditional EDI (from a large supply chain perspective) are focused on large corporations at the top and the 1st and 2nd tiers of their supply chains (Table 1). These corporations command significant market share and product volume to gain a competitive advantage from EDI.

Table 1: EDI Penetration in Supply Chain Tiers

Tiers	Significant Penetration	Slight Penetration	Very Low Penetration
Top Tier	X		
First Tier	X		
Second Tier		X	
Third Tier		X	
Lower Tiers			X

Moving further down the supply chain, EDI provided little or no competitive advantage and the EDI implementation rates at these tiers were very small. While some may be surprised by this assertion, it is not surprising to those who work extensively with SMEs. These suggestions are completely consistent with our shipbuilding supply chain project[3], and Shunk's (1996) supply chain

[1] Electronic Data Interchange is the transmission of business transaction information in computer-readable form between organizations in a standard format.
[2] Colberg, et al. (1995), provides a comprehensive review of EDI and EDI implementation practices.
[3] This project is described in detail in a later section.

research. The obvious question then follows: What caused this disparate level of EDI acceptance and what has been its effect?

This analysis is an outgrowth of our experiences obtained from working for two years in support of a large effort to integrate a major supply chain using VAN-based EDI. We reached an understanding of the extent of EDI penetration after many months of attempting to educate SMEs on the use of VAN-based EDI. This paper describes these experiences in some detail, explaining why VAN-based EDI is inappropriate for most SMEs, and also making recommendations for what is more appropriate: Internet-based Secure Transaction Processing.

The paper is organized as follows. The following section summarizes the basic concepts of EC that are relevant for our large supply chain integration project. Then, we describe the project and our experiences from the implementation effort. We also describe, with supporting references, an alternative to VAN-based EDI that is more likely to succeed with SMEs. Finally, we describe in some detail how we are focusing our implementation efforts (Phase II of this project), given our experiences that are explained in this paper.

EDI Architectures

We have used the following basic EDI architectures to understand EDI implementation impediments and enablers (Figure 1).

- Standalone: Store-and-Forward - EDI messages are delivered in a Value-Added Network (VAN) environment that offers 24-7[4] EDI message transaction handling services between suppliers and customers. Messages could also be received through an EDI translator module and printed. The printed orders and the message acknowledgment are often manually processed.
- Integrated EDI: Store-and-Forward - EDI messages are delivered in a Value-Added Network (VAN) environment that offers 24-7 EDI message transaction handling services between suppliers and customers. Messages are received through an EDI translator module and routed to a corporate information system. The transaction and the message acknowledgment are processed automatically by the corporate information system and electronically forwarded back to the trading partner.

Alternatives to Traditional EDI: A Look at EC-based Business Transactions

Despite the simplicity and stability offered by traditional EDI, attempts to implement these architectures were not as widely successful as had been envisioned. Many corporations implemented test case Standalone **Store-and-Forward** solutions. These were quickly outgrown, but security concerns and corporate policies prevented corporations from implementing the integrated EDI option. In such organizations, EDI transactions were employed only with those special customers that insisted on electronic business data; other customers were serviced in a more traditional paper-based manner. Most small companies who acquired standalone solutions, however, found that the actual transaction volume could not justify the cost of a **Store-and-Forward** solution.

[4] 24-7 allows messages to sent and received 24 hours a day for seven days per week. The VAN stores messages until customers and suppliers are ready to receive them.

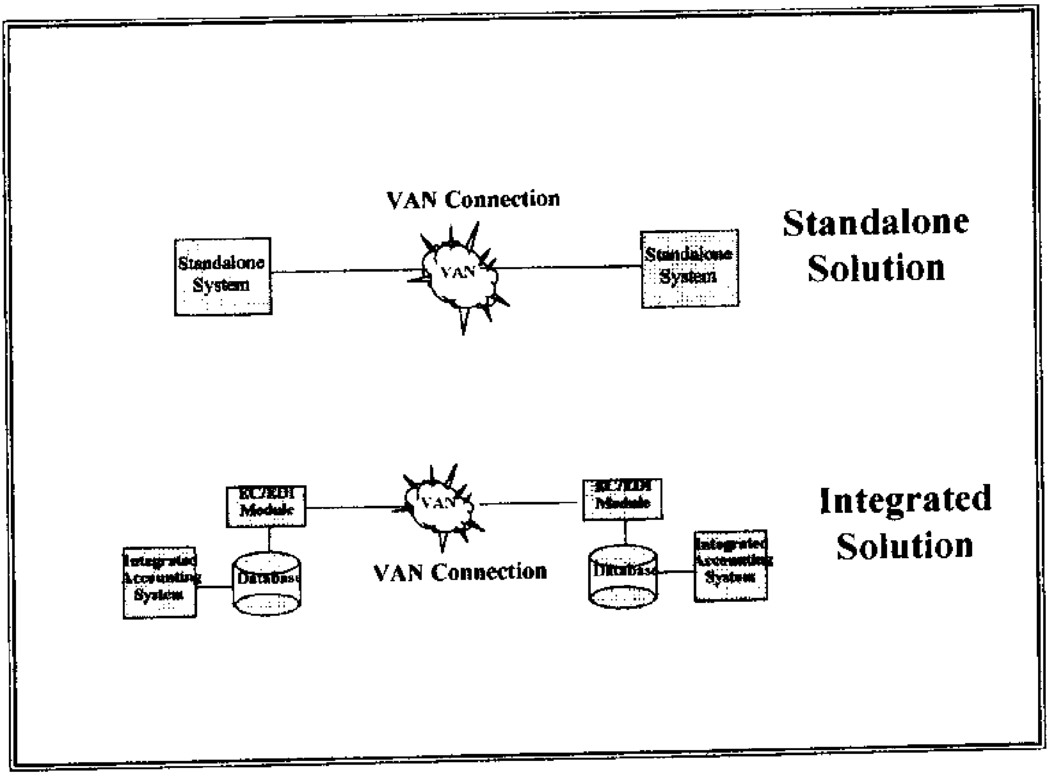


Figure 1: Basic EDI Architectures

By the late 1980s, the EDI service industry was having difficulty positioning software and services that could meet the needs of many small and medium sized businesses. At the same time, many large corporations implemented proprietary Electronic Commerce[5] solutions that were often based on private leased line (X.25 packet switched networks) and transaction standards other than X.12 or EDIFACT.

Factors that influenced this growing trend of EDI implementation reluctance include:

- Pricing and complexity of EDI software - Users found it difficult to justify the software, installation and VAN service costs, and
- Software and service interoperability - Users found that the software and VAN service implementations that were based on the X.12 and EDIFACT standards often resulted in incompatibilities between competing software products and services. Each company has its own implementation conventions, and VAN service providers do not automatically communicate with each other.

Although there are many documented cases of successful EDI implementations across the US and throughout the world, the impediments are well known. For example, Smith (1996) states directly that "despite 15 years of government and industry advocacy, EDI has been static for years because of the inflexibility of the technology and the high cost of the dedicated, proprietary value-added networks (VANs)."

[5] Electronic Commerce is the secure bi-directional flow of business data.

The overall effect of these encumbrances was that many companies had given up on the EDI standardization process and began to implement their own proprietary solutions. Although this strategy proved expensive, these companies realized a significant competitive advantage after deploying trusted leased line networks[6]. By gaining direct access to corporate data, high volume top and first tier suppliers were able to reduce their delivery cycle time significantly, while top and first tier customers reaped the cost and customer satisfaction benefits of a Just-In-Time (JIT) inventory replacement model.

Unfortunately, this trusted leased line network option was not cost-effective for smaller companies, nor was it made available to the lower tier suppliers who were forced to do business with traditional paper-based methods. It was not until 1993 that emerging Internet technologies would offer these companies an affordable and efficient EC solution. With the advent of **Internet-based Secure Transaction Processing** small businesses were able to take advantage of "open" World Wide Web solutions that were cheaper than proprietary leased line solutions, more economical and scaleable than traditional EDI, and provided reasonable security that could be integrated very easily into existing accounting and Enterprise Resource Planning (ERP) systems[7] (Figure 2).

Since 1993, the influence of the Internet and the "open" nature of the TCP/IP WWW protocols have prompted many SMEs to view Electronic Commerce as a relatively low level distributed secure transaction processing activity rather than as part of a more complicated EDI solution. This transition in technologies has opened the door for SMEs to participate in EC, and it has placed tremendous pressure on suppliers of traditional EDI services and the proponents of the EDI model. Even the largest proponents have not been isolated from the pressures, as evidenced by the rejection of the federal government's acquisition network (FACNET), which mandated a van-based EDI solution through the implementation of the Federal Acquisition and Streamlining Act of 1993 [see, for example, Deller (1997) and Slabodkin (1997)].

In fact, many companies have totally abandoned EDI in favor of WWW based Extranet solutions that link customers and suppliers to secure corporate Commerce Servers (Figure 2). This is specifically the case with Mobil Oil (Mullich, 1997), and the trend in other companies is significant (Waltner, 1996). Proxy servers provide a controlled external mirror image of a portion of the actual corporate data base that allows secure transactions to be processed in real-time without giving customers (or potential "hackers") access to internal corporate data. Data on proxy servers is only posted to the corporate database once it has been analyzed and verified. This verification and posting process is in most cases automated and invoked at regularly scheduled intervals (i.e. once every 30 minutes).

Three Emerging EC/EDI Transaction Approaches

As today's business environments become increasingly competitive, firms are required to rapidly reconfigure their business processes. Traditional enterprise boundaries no longer exist. To remain competitive and realize lower costs, each enterprise seeks to establish closer relationships with its partners, customers, and suppliers. Apart from the traditional EDI mechanisms outlined in figure 1, this close relationship with external enterprises can be established through a number of new supply-chain integration mechanisms. The exponential growth of the Internet along with the well-timed

[6] Many case studies are now appearing in the popular press. For example, when Mobil Oil moved from VAN-based EDI, the savings were substantial (Mullich, 1997). "Mobil encountered many of the problems that have stymied the growth of EDI. VAN charges for using the hard-wired networks topped $100,000 a year. Maintenance was burdensome; every time Mobil changed a business rule, new software had to be sent to each dealer and installed on their desktop."

[7] Turbide (1997) presents the ERP integration issue.

advance of integrated Enterprise Resource Planning system software packages is causing many innovative solutions to appear in the market [Varney & McCarthy (1996) and Tucker (1997)].

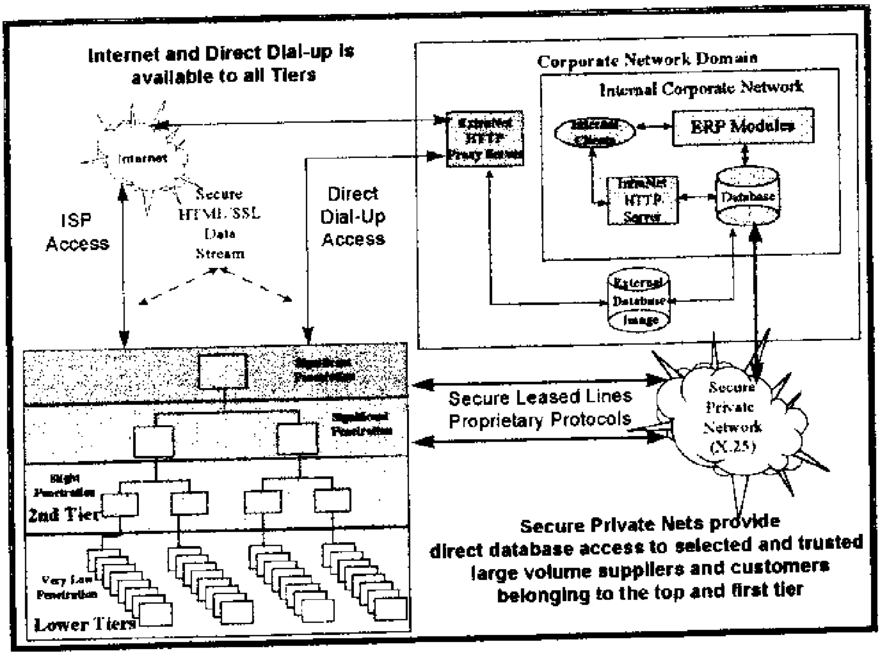


Figure 2: Internet and Private Network Supply Chain EC Concepts

While it is unlikely that all SMEs will adopt a fully integrated approach, there will be an increasing variety of implementation options and degrees. Rather than facing a limited choice between no EDI, stand-alone non-integrated EDI, or Integrated EDI, SMEs may have a choice of varying degrees of integration, allowing them to choose the one appropriate given their "business case."

The Shipbuilding Supply Chain Project

In 1997, our project team entered into an agreement with Newport News Shipbuilding (NNS) to help them integrate their supplier base of 24,000 firms. The initial focus was on electronic procurement, with an eventual inclusion of technical data exchange. This was a dynamic period for NNS, as many internal change initiatives were underway. NNS had just separated from Tenneco (their parent company), and had once again become the largest private employer in the state of Virginia. NNS was also in the early stages of implementing major change initiatives, including company-wide client-server ERP solution, a new Product Data Management (PDM) solution, a new Component and Supplier Management (CSM), a new CAD environment, and a new Procurement Execution System (PES).

Our primary focus was not on the internal operations of NNS, but their supply chain. NNS had a plan for bringing all active[8] suppliers into an electronic procurement environment using VAN-based EDI. Our task required basic understanding of NNS's internal operations and supply chain interfaces,

[8] About 4,000 of the 24,000 suppliers were active at any point in time.

but our primary focus was on the SMEs that supply the prime contractor, and indirectly NNS's major customer: The U.S. Navy.

Approach

Our team worked with NNS to construct a database of characteristics for their top 1000 suppliers. The geographical location of these suppliers is presented in Figure 3.

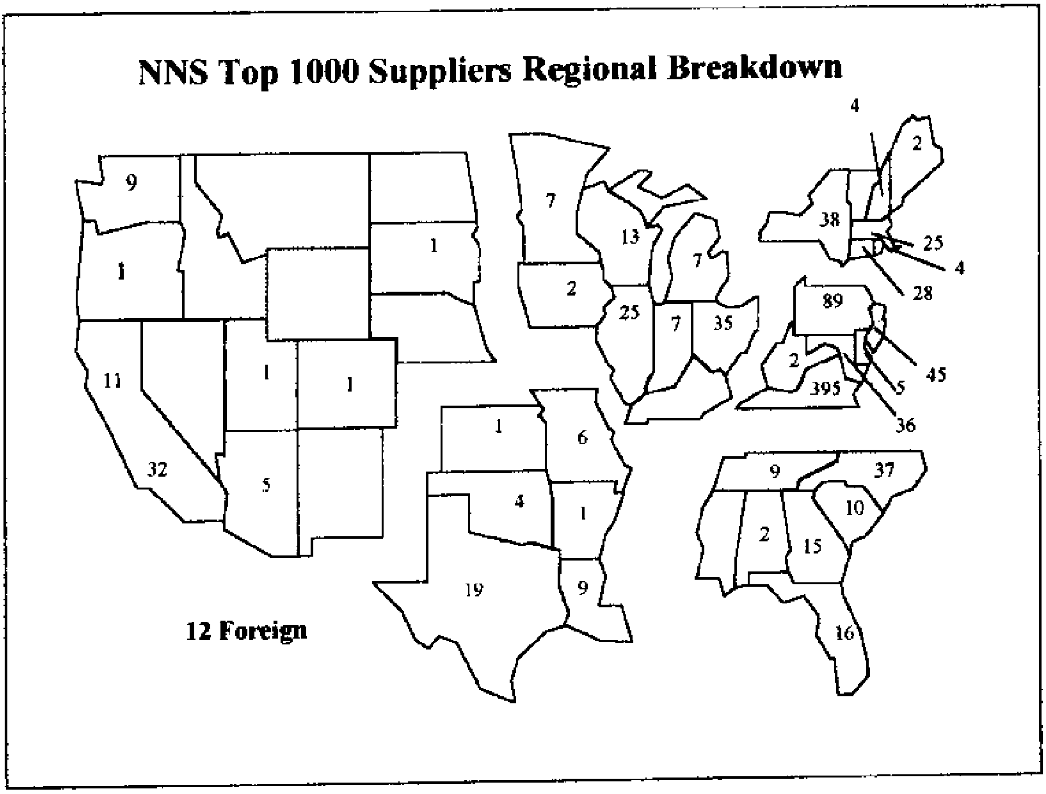


Figure 3: Geographical Dispersion of NNS Suppliers

We also confirmed that these 1000 top suppliers represent the bulk of the procurement transactions that are executed by NNS; i.e., our sample represented about 95% of the total purchase orders, even though it was a small portion of the total number of firms in the NNS database.

Our plan was to develop focused supplier training sessions, and invite all suppliers in targeted geographic areas to participate. The training, which was designed mainly to promote "awareness," detailed NNS's plans for requiring their suppliers to move to VAN-based EDI for procurement. Given the dispersion in Figure 3, we focused our efforts east of the Mississippi river. In the end, we personally invited[9] 72% of the top 1,000 suppliers; this figure represented 82% of the purchase order volume. This coverage statistic is depicted below in Figure 4.

---

[9] Each company was contacted by telephone, and an effort was made to locate the responsible person for electronic procurement within the company. As will be seen later, we feel that this significantly enhanced our response rate and attendance at the training sessions.

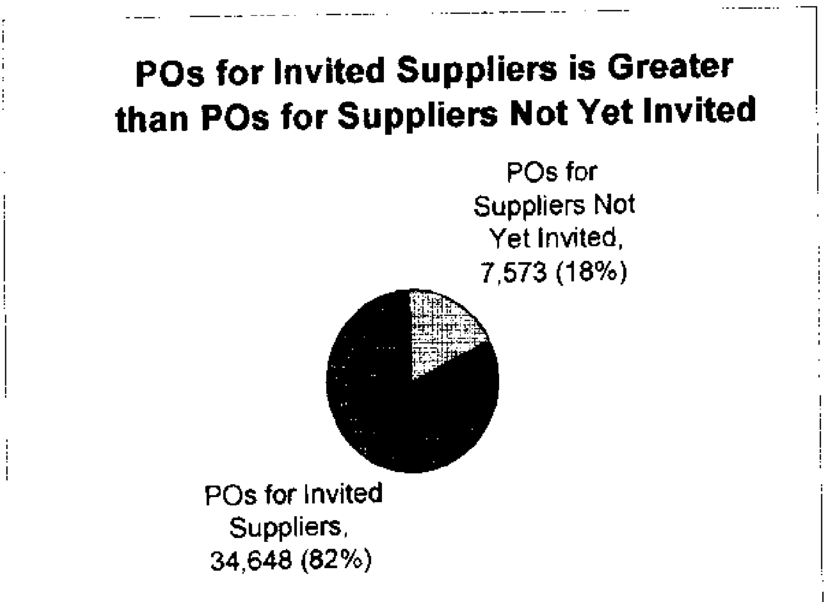


Figure 4: Purchase Order Coverage for Selected Firms

After the training part of our approach, we planned to offer "Technical Support" to those firms requesting specific help in the implementation of EDI. That is, the target SMEs were given information about EDI in general, (and EDI with NNS in particular). Those firms with the appropriate knowledge could procure the necessary hardware, software, and services; and enter into certification testing with NNS on their own. Others could request that we help them, and this service was provided at no cost to the SME.

Of the firms invited, 25% attended the training sessions. This amount still represented 45% of all purchase orders. We attribute this high response rate to persistent telephone efforts to identify the appropriate person within the target SMEs; that is, the manager responsible for electronic procurement and/or sales to NNS.

Evaluation Criteria

To evaluate the supply chain integration engagement, we established a set of evaluation criteria. We developed the following four broad characterizations for assessing effectiveness:

- ***Coverage*** - Are we reaching both a broad cross section of firms as well as the most important firms?
- ***Understanding of SMEs*** - Do we understand the technology level of the SMEs? We constructed a survey instrument to provide a snapshot of the technological capacity of NNS suppliers.
- ***Efficacy of Training*** - Has the training provided sufficient information to the vendors in order for them to implement VAN-based EDI with NNS? Case Studies also are used as supporting evidence
- ***Integration to Other Objectives*** - What technical support leads and follow-up consultation work resulted from the training sessions with the NNS vendors?

Supplier Coverage

The approach gives some indication of the coverage of our efforts. However, it is important to make sure that the high volume customers (with respect to purchase orders) are being reached. Figure 5 provides a picture of the NNS suppliers. Basically, there are a large number of suppliers that receive a few purchase orders. However, it should be noted that there are a significant number of suppliers at a second mode who receive a relatively large number of purchase orders. The overlay darker bar in Figure 5 indicates that these SMEs were eager to attend the sessions, as much as the "many companies/low POs" group. We were somewhat surprised by the attendance from those firms with small transactions base. After talking with them, however, it was clear that they were searching for ways to increase their business base with NNS. Some in fact made the investment in VAN-based EDI in anticipation of more business with the shipyard, only to be disappointed when the business did not materialize. This implications of this phenomenon will be discussed in more detail in a later section.

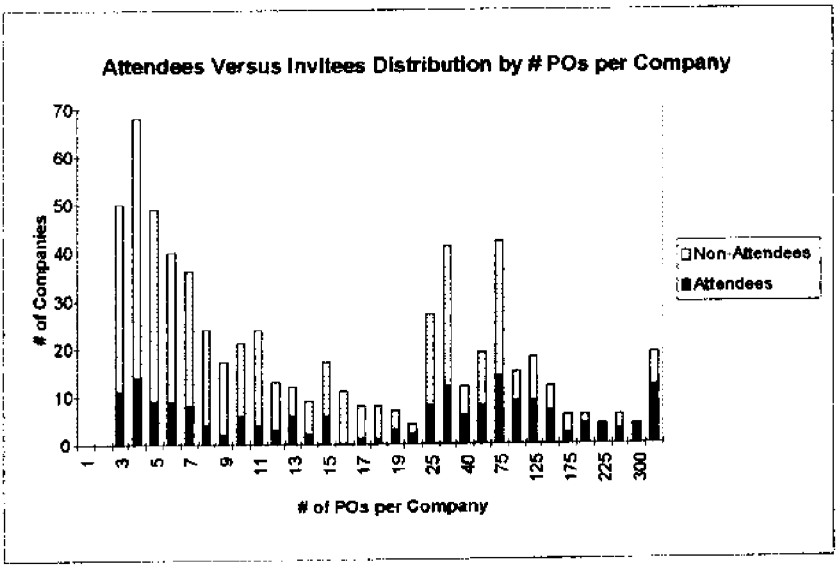


Figure 5: Attendees at Training Sessions by Purchase Order Volume

We were pleased with our coverage. We have certainly learned lessons that would alter our approach for future endeavors[10], but on the whole, given the information available before the training session, we are satisfied that appropriate care was taken in constructing the sample of SMEs.

Understanding the SMEs

To properly assess the effectiveness of our efforts, we felt that we must attempt to quantify in a meaningful way the technological sophistication of the firms encountered. For example, if the training appeared ineffective (i.e., few firms adopted VAN-based EDI), it might be argued that lack of SME technology and IT infrastructure posed an obvious barrier to adoption. To obtain a better understanding of SME technological ability, we constructed a survey instrument that was distributed to the firms by fax after identifying the appropriate contact person. If the firm attended a training session and had not completed a survey form, they were contacted again to complete the form.

[10] These lessons-learned will be presented in a later section.

The firms were asked to place themselves in one of the following categories: 1) not using EDI, 2) currently using stand-alone non-integrated EDI, or 3) currently using integrated EDI[11]. The results of this survey are summarized in Figure 6.

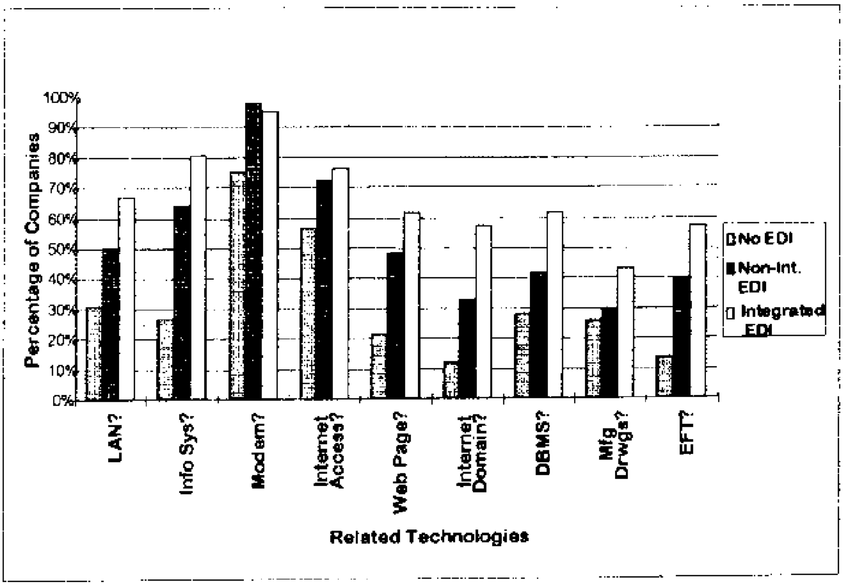


Figure 6: Technological Sophistication of Firms in Sample

The results were surprising. Each category in Figure 6 contains a simple question: Do you have this technology capability in your company? The first bar in each category represents companies categorizing themselves as not using EDI, the second bar those involves in some basic form of EDI, while the third bar represents companies with a fully integrated EDI solution. As with any sample, there are some inconsistencies, but the striking feature is the number of firms with modems and Internet access. It is also worth noting that a large percentage also maintain corporate web pages.

This evidence points to the fact that technology is not a significant barrier in EDI adoption. These firms have access to what is currently considered high technology. If they elect not to participate through VAN-based EDI, then there must be other, more fundamental business reasons for this choice. While integrated EDI may not be fully achievable for these firms, they should be capable of at least stand-alone non-integrated EDI.

Efficacy of Training

The training that we performed was primarily directed toward "awareness" of EDI. In order to measure the efficacy of this training, we sent a follow-up survey to participants and queried them as to whether they have adopted EDI and if not, why? While this analysis is not yet complete, initial results seem to show that after our training nearly all participants were aware of EDI, and some had gone as far as to adopt it. Those who have chosen not to adopt EDI have overwhelmingly said that it was because they could not make a "business case" for it, and they were waiting for more cost-effective

[11] The distinction is as follows: with integrated EDI, incoming and outgoing data is mapped directly from the EDI translator into the firm's database or the application logic of the ERP system. Non-integrated EDI is where the firm responds to transactions in a hands-on mode through software residing on a PC. The incoming and outgoing data require manual intervention for processing.

solutions to present themselves. Our results showed that very few SMEs who use EDI have their software integrated into their other business applications or databases, and there was little indication that they planned on doing this in the near future.

Integration to Other Objectives

The positive externalities created by our work with this supply chain were numerous. Benefits included gaining access to a large number of SMEs. This is paramount to us as they are often very busy organizations with few slack resources that can be devoted to conceptual planning and looking outside the organization for solutions. By working with NNS, we were able to gain meaningful access to the SMEs without incurring a large marketing expense.

We also gained a much better understanding of the needs, capabilities, and decision-making processes within SMEs. This knowledge enabled us to better understand and fine-tune our role as intermediaries in the technological advancement process.

Finally, we accumulated general knowledge on how to understand and approach a large supply chain. It is clear that particularly with a large prime like NNS, the supply chain is extremely heterogeneous. The dimensions of size and technological sophistication of individual supply chain members are obvious, but there are other subtle distinguishing attributes such as the percentage of sales that are derived from an individual company or industry, or a particular type of firm (distributor, manufacturer, etc.). A solid grasp of these important dimensions enables a better understanding of the business relationships between prime and supplier and the leverage and incentives that interplay between the two.

Lessons Learned from the Supply Chain Integration Effort

The average corporate picture that emerged from the self-selected sample of firms is a firm that has the necessary technology resources available, including internet, to install and run EDI, and has been educated in the uses and nature of EDI. But they still do not use EDI. The question then is why?

Our analysis of the situation concentrated on two areas:

- An thorough analysis of the product that was made available to the SMEs; i.e., VAN-based EDI, and
- The economics (i.e., the business case) for implementing and maintaining the VAN-based EDI product.

After many discussions with the firms regarding these topics, two clear conclusions could be drawn:

- The EDI product was not being used effectively in this shipbuilding environment. EDI was primarily used for trading process management; i.e., declaring a "winner" in a competitive bidding process and establishing a contractual relationship. The gains from implementing VAN-based EDI are primarily from executing routine delivery orders in an integrated environment; not the establishment of contractual relationships, which still requires hands-on intervention[12].

[12] It is interesting to note that the U.S. government employs this same inefficient use of EDI. Trading process management (i.e., the awarding of a contractual relationship) provides some efficiencies, but it is the routine execution of delivery

- The low volume of transactions (see figure 5), combined with the technology sophistication required to implement integrated VAN-based EDI, creates a "business case " barrier that is difficult for the firms to overcome. Economic justification is difficult, even for the non-integrated environment where the supplier is trying to preserve a business relationship with the prime.

To test the validity of the above conclusions, we searched for similar implementation experiences. Knowing that shipbuilding is a unique environment[13], we focused our efforts on other intra-industry attempts to implement EDI in shipbuilding.

The European Experience

The European experience with EDI in shipbuilding provides similar insight into the nature of implementation barriers discussed in the previous section. The views in this section are based on interviews with participants in a large European Union effort called EDI for the European Maritime Industry (EDIMAR). The EDIMAR project is funded by the Commission of the European Communities under the ESPRIT Program (Project No: EP 20.624). Its objective is to intensify inter-organizational cooperation within the maritime industry and to increase productivity through use of Electronic Data Interchange. Four major European shipyards are participating in the project:

- Astillerso Españoles (Spain),
- Chantiers De l'Atlantique (France),
- Ficantieri (Italy), and
- Howaldtswerke-Deutsche Werft (Germany).

The EDIMAR project is focused on the exchange of business and technical data, and it is a two-year effort that began in January 1997.

When the results of our NNS engagement were presented to the EDIMAR implementers, there were not surprised by the data. In fact, they were amazed that we had such success moving SMEs into production EDI, and in such numbers. This came as a surprise to us, since our Defense Logistics Agency sponsor had recently expressed a feeling that our NNS numbers were too low and not what they could be.

The EDIMAR implementers noted an interesting characteristic of shipbuilding industry:

> A large number of firms receiving a small number of purchase orders means that the 'automotive industry' model of EDI is not appropriate. For the automotive industry, the volume and frequency of orders is much higher than in shipbuilding. The automotive buyers even provide a 'preview' of what they intend to order in the future, allowing the suppliers to quickly respond once the order is released – this accommodates just-in-time manufacturing. Since shipbuilding purchase patterns are very different (i.e., infrequent and small numbers of orders with no advance previews), it was clear that as strategy was needed that allowed SMEs to build the business case for participation in this market." [Personal interview (February, 1998)].

Given such an environment, the EDIMAR participants sought to modify traditional EDI to make it more palatable to SME economic tastes.

(release) orders after the establishment of a contract that yields the largest benefits to both parties. The establishment of the contractual relationship requires manual intervention by one or both parties. Release order execution requires no intervention and can be accomplished in a fully automated environment.

[13] For example, we have determined that the automotive industry is very different in its implementation of EDI. There is a larger volume of transactions, and the focus is on release orders to support just-in-time manufacturing.

The participant shipyards adopted an EDI clearing-house approach that makes electronic exchange easier to implement in SMEs. BALance Technology Consulting, a member of the EDIMAR project team, developed this approach, known as BAL.DIS. The idea is simple. BAL.DIS is an EDI broker service, accepting proprietary SME data, translating it into the EDIFACT standard, and forwarding the data (in any format) to another company. That is, the BAL.DIS service handles all EDI mapping and technology issues, allowing the firm to communicate directly from its legacy business systems. This is accomplished at a cost that is justifiable by the SMEs, since the SME does not have to invest in integrated EDI solutions.

While this is not the "pure" and perfect technology solution, it is appropriate in situations where the transaction volume does not support the investment in integrated EDI. Also, the solution should not be confused with the services provide by a traditional VAN service provider or an EDI service bureau. BAL.DIS provides more than the exchange and tracking of transactions. The SME organizational processes are analyzed as part of the solution, and the SME is relieved of expensive internal mapping and integration costs.

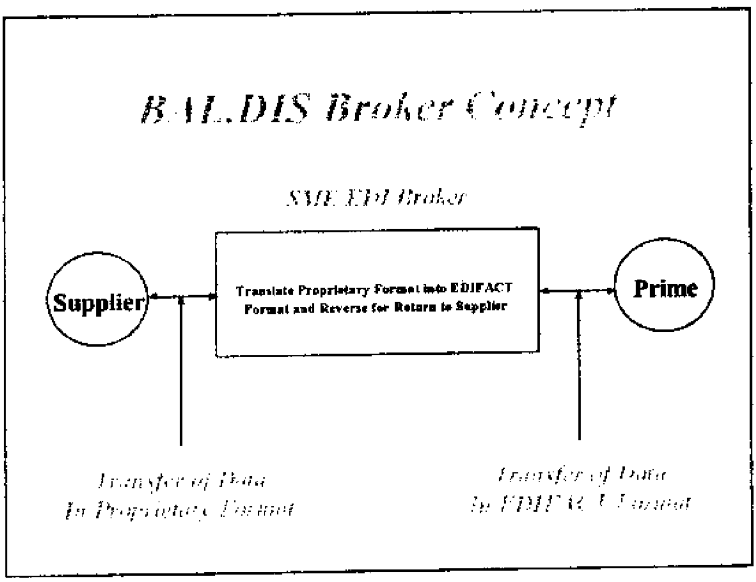


Figure 7: The Concept of an EDI Broker Service

As a final point, the EDIMAR implementers confirmed our contention that, in general, the problems of SMEs in shipbuilding supply chains, are not technology related. The EDI broker concept can overcome the technology problems. The larger problem is the inter-organizational ability to adopt to change. Shipyards in Europe have downsized, just like those in the United States. As noted in our interviews, "The newly hired and the senior employees have departed. The yards are left with the non-technologically sophisticated middle." Unfortunately, these are the most difficult employees to adapt to the new high technology-enabled environment.

Review of Lessons Learned

Supply chain issues with regard to EC are increasingly important. Understanding the supply base is crucial to any successful implementation of a supply chain EC effort. To be sure, one needs to understand the technological capabilities of the suppliers and the match/mismatch of potential technological solutions; however, even more important is sound knowledge of the nature of the business relationship between customer and supplier.

Technological solutions that allow for fine degrees of divisibility in adoption may be more suitable for the wide range of suppliers that are often encountered in a supply chain. The business case for adoption by the SMEs must be strong if there is any hope for widespread adoption. SMEs are willing to expend some effort to maintain customer satisfaction, but they are not willing to expend a large amount of money/time for solutions that that lack a clear, economic benefit.

The success of our engagement was largely due to the openness of both NNS and a large number of their suppliers. We were able to maintain this open dialogue by clearly stating that we were an independent organization that was not being paid by NNS. This helped to assure the suppliers that we were seeking to educate, not indoctrinate.

A final lesson learned was that solutions that are industry-specific would probably not be viable in the end. The technologies are rapidly changing, and any solution that is developed specifically for the shipbuilding industry will probably not advance at the same rate as general EC technologies.

Proposed Solutions for SMEs and Lower Supply Chain Tiers

The reality is that most SMEs at lower tiers will opt not to integrate their business transactions with their internal business databases. Any potential solution must accommodate this reality by understanding the economics of those suppliers' choices. They want a solution that has a low cost and requires little or no training and technology sophistication. These firms are interested in making money, not implementing technology. However, if there is money to be made by implementing technology, then SMEs have a much greater motivation to participate.

As new mapping technologies, such as Extensible Markup Language (XML)[14], begin to enter the marketplace, there are expanded opportunities for viable and cost-effective solutions for SMEs. These technologies should be directed to some extent by the lessons learned from the EDI experience. Still, no technologies will be implemented unless the SME can use the technology to expand its business base.

Conclusions

Our successful engagement with a supply chain illuminated many of the ground level issues facing SMEs in the movement towards Inter-enterprise supply chain integration. It is clear that the drivers of EC/EDI adoption are business and cultural, not technological. In order to overcome these barriers to adoption, the business case must to be articulated by the SMEs.

Our research on a large shipbuilding supply chain indicates that the barriers are not technological. Most of the firms have access to the Internet and many maintain their own Web domain. While ease of use (or lack thereof) was a big issue, the primary concern was expanding the business base and making more money. In this supply chain, it was difficult to justify VAN-based EDI as the key to reducing costs or expanding business. Given the level of transactions in shipbuilding, the additional costs of VAN-based EDI do not exceed the benefits.

---

[14] See the article by Drummond and Spearman (1998).

References

1. Colberg, Thomas P., Nicole W. Gardner, Kerry J., Dennis M. McGinnis, Phillip W. Mclauchlin, and Yuk-Ho So (1995), *The Price Waterhouse EDI Handbook*. New York, John Wiley & Sons.
2. Deller, Robert (March 20, 1997), "FACNET Faces Uphill Battle for Acceptance." *Washington Technology*, Vol. 11 #24, p. 13.
3. Drummond, Rik and Kay Spearman (May 1, 1998), "XML Set to Change the Face of E-Commerce," *Network Computing*, pp. 140-144.
4. Fabris, Peter (June, 1997), "EC Riders," *CIO Magazine*.
5. Fulcher, Jim (April, 1997), "A Sound Foundation," *Manufacturing Systems*, pp. 36-48.
6. Mann, Steve (June, 1996), "Good-Bye EDI, Hello Internet: The Role of the Internet in Supply-Chain Management," *Manufacturing Systems*, Vol. 14 #6, pp. 16A-17A.
7. Mullich, Joe (April 14, 1997), "Mobil Is Moving EDI to an Intranet," *PC Week*, Vol. 14 #15, p 35.
8. Radosevich, Lynda (January, 1997), "The Once and Future EDI," *CIO Magazine*.
9. Reda, Susan (May, 1997), "Internet EDI," *Stores*, pp.18-21.
10. Shunk, Dan L. (1996), "And Wallace Patterson, Electronic Commerce Facilitates Time Compression," *1996 International Industrial Engineering Conference Proceedings*. Norcross, Georgia, Institute of Industrial Engineers.
11. Slabodkin, Gregory (June 2, 1997), "Defense Wants of Bypass FACNET and Use Alternate Buying Methods," *Government Computer News*, p. 65.
12. Smith, Howard (July, 1996), "Caught in the Web: Can Value-added Networks Survive Internet EDI?" *EDI World*, Vol. 6 #7, pp. 44-48.
13. Tucker, Michael (April, 1997), "EDI and the Net: a Profitable Partnering," *Datamation*, pp. 62-69.
14. Turbide, Dave (July-August, 1997), "The New World of Procurement," *Midrange ERP*, pp. 12-16.
15. Varney, Sarah and Vance McCarthy (October, 1996), "E-Commerce: Wired for Profits," *Datamation*.
16. Waltner, Charles (November 11, 1996), "Web Commerce: Intranet Business Applications," *Communications Week*, pp. 59-66.
17. Wilder, Clinton and Tom Stein (Oct. 6, 1997), "Application Integration Chain," *Information Week Online*.

Applying STEP Technology to Shipbuilding

Dr. Matthias Grau, KCS Consulting GmbH, Hamburg, Germany
Dr. Thomas Koch, KCS Consulting GmbH, Hamburg, Germany

Introduction

In the last several years, computerisation in the maritime industry has reached a stage where digital product information is created, maintained and explored using a broad range of software solutions throughout the product life cycle. This holds for all stages of the design process, from production planning and controlling to the handling of data during the operational phase and includes the exchange of data with classification and on-shore authorities.

Ships are complex systems and information of very different nature is created and handled during its life cycle. Started as very specialised solutions, the isolated islands of the product life cycle covered by software have gradually grown together during the years. Today's systems are capable of covering several life cycle stages by having a number of modules (like structural design and production planning) seamlessly integrated and operating on the same database.

What remains is the problem of transferring information between different systems dealing with successive life cycle stages and the problem of transferring information between collaborating companies using different software. In these cases either costly proprietary solutions must be developed or the data exchange between the involved systems is restricted to geometry data where standards such as IGES already exist.

The ISO 10303 (STEP) standard is being developed in order to overcome this type of problem by providing the capability to exchange virtually any product data by mapping industry specific information to a neutral and generic product information model.

It consists of several families of subsequent documents that are developed and published separately under a certain part number. From

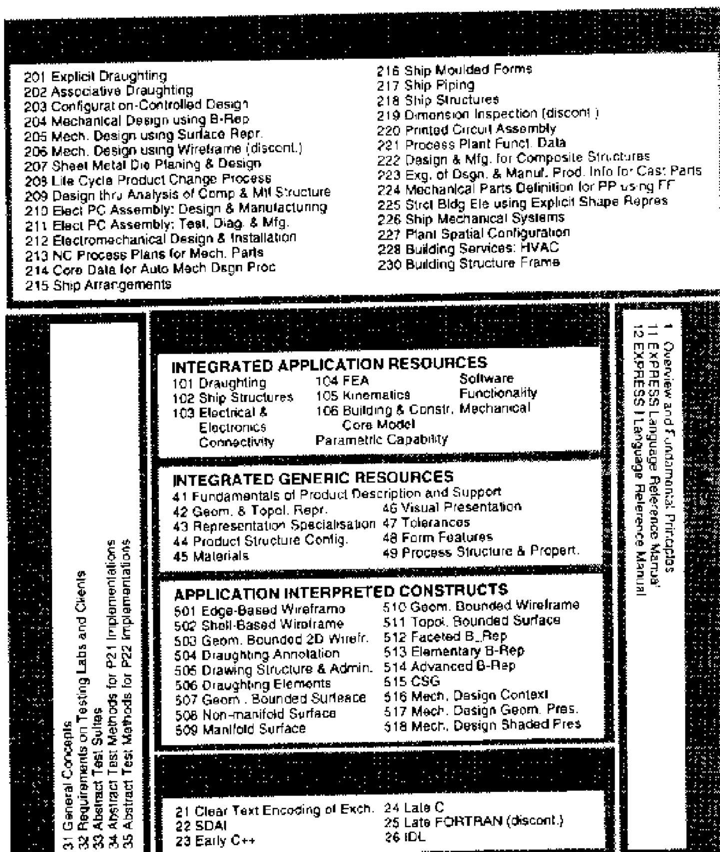


Figure 1: STEP Structure

the viewpoint of practical application of STEP, three part families can be seen as the most important:

- The application protocols (APs) formulate the requirements of the product data to be handled in the scope of a certain industrial area. These are documented in the Application Reference Model (ARM), which is part of each AP. To achieve this, the terminology of the industry is used. APs provide the basis for the implementation of STEP as they describe what data is to be exchanged. Part numbers of APs range from 201 to 299.
- The integrated information resources (IRs) describe a generic data model of a product. Parts of this data model are instantiated in order to fulfil the requirements formulated in an AP. Therefore the requirements of APs must be expressed by constructs of the IRs. The process of expressing the requirements of an AP this way is part of the AP development process and is called interpretation. The IRs can be seen as the 'language that STEP speaks'. Parts belonging to the IRs are in the range of 41 to 99 (integrated generic resources), 101 to 199 (integrated application resources) and 501 to 599 (application interpreted constructs).
- The third important part family within STEP are the implementation methods. These describe the way an AP is represented in a software implementation. This is either as a binary data repository having a well-defined access interface – the Standard Data Access Interface (SDAI) – or as a human readable ASCII file – the STEP Physical File. APIs formulated in different programming languages such as C, C++, Java, or IDL give an application access to the binary repository. Implementation methods are described in parts ranging from 21 to 29.

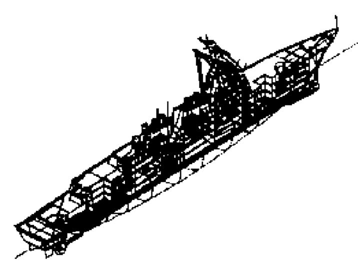
The implementation of STEP means the implementation of an AP into a piece of software using one or several of the implementation methods.

Shipbuilding Application Protocols

The suite of APs currently available or under development covers the requirements of general mechanical design and draughting, automotive industry, electromechanics and electronics, process plant, building and construction as well as maritime industry. Six APs represent the area of maritime industry.

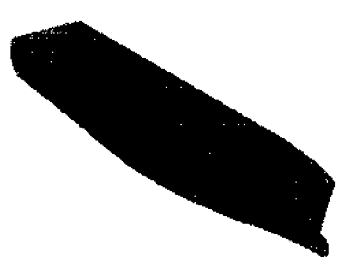
Ship Arrangements (AP215)

The AP "Ship Arrangements" [8] describes the requirements of the spatial subdivision of a ship into zones and compartments. In addition, all design related information like function, geometry, topology as well as data about usage, access and occupancy is covered. AP215 is currently in CDC stage and will undergo a second wide industry review this year.



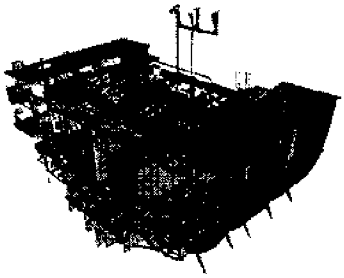
Ship Moulded Forms (AP216)

The geometric definition of the ship hull is the subject of the AP "Ship Moulded Forms" [9]. This includes the outer hull form as well as the major interior surfaces of decks and bulkheads. This is complemented by main dimensions and general characteristics such as building ship yard, classification society and owner designation as well as hydrostatics data. AP216 is currently under interpretation and is expected to be out for CD ballot this year.



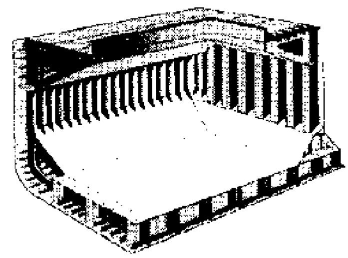
Ship Piping (AP217)

All aspects of piping system design are provided by AP "Ship Piping" [10]. The main focus is on the hierarchical breakdown of pipe systems into pipe parts as well as the connectivity at the different hierarchy levels. Both diagram and 3d solid model representations are available allowing different views of the pipe structure. AP217 will be interpreted likely beginning sometime this year.



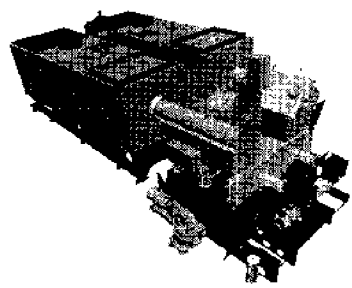
Ship structures (AP218)

AP "Ship Structures" [11] deals with everything about the ship made of steel. This covers the hierarchical breakdown, from the design as well as production viewpoint, into structural parts. The approval at different stages of the design process is addressed as well as the design for manufacturing. AP218 is currently in the process of interpretation jointly with AP216, and will be circulated for CD ballot starting later this year.



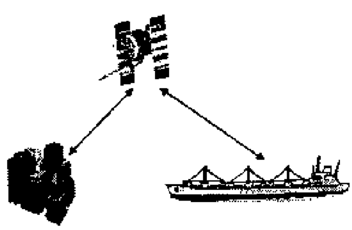
Ship Mechanical Systems (AP226)

Data structures about the mechanical systems of a ship are defined in AP226 [12]. This includes all types of main and auxiliary machinery. Special attention is drawn to the connectivity of the mechanical systems and components. There is a potential overlap with AP217 that needs to be addressed. The AP has passed the wide industry review and interpretation is planned to start this year.



Operational Logs, Records and Messages (AP234)

The newest AP in the maritime area is Operational Logs, Records, and Messages. It defines standard tags related to the operational data such as navigational and hull stress monitoring data, environmental observations and tank surveillance. Those are collected and processed by the ship's onboard control systems and exchanged with for example the ship operator and classification societies ashore. This AP is approved as a New Work Item (NWI) within ISO TC184 / SC4.



Building Block Approach

The development of all shipbuilding APs is carried out within working group 3 task 23 of ISO TC184 / SC4. In order to avoid redundancies and to make these APs integrated parts of an overall ship product model a special modelling technique has been introduced [13]. The ARMs of most of these APs are configured from relatively small data model units written in EXPRESS – the modelling language defined within STEP [1] – and called building blocks. This special modelling approach was taken by the shipbuilding AP developers to allow distributed modelling and to share parts of the data models across APs. The modularization approach that is currently discussed for STEP in general would be the continuation of the building block approach as it intends to define data model parts on both the ARM and AIM level.

The AP developers have access to the building blocks via a server that is operated by the European Maritime STEP Association (EMSA) [25]. While this has turned out to be an efficient way to develop

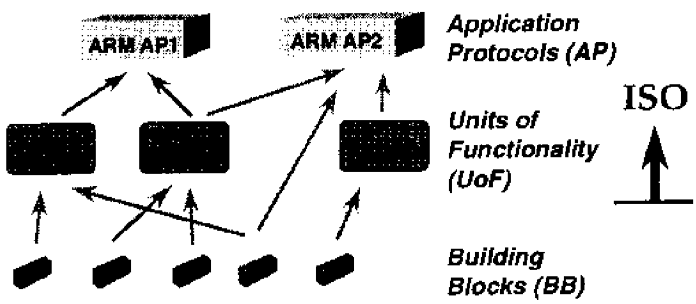


Figure 2: Building Block approach

related APs in parallel, it also has the big advantage that a fully attributed EXPRESS model of the AP's ARM can be generated out of the building blocks using off-the-shelf tools. Although this is not required by ISO it allows for pilot implementations before the AP has finished its interpretation, i.e. before the standard data model for the AP exists. This enables extensive validation of the APs by implementation projects long before they become a standard.

The Challenge for STEP

Over the last several years research projects in the shipbuilding area, partly funded by national governments and international organisations, supported this type of validation by implementation. Several of them were also to support the development of one or more of the ship APs so that they acted as a kind of iteration loop between model development and model validation. Apart from the important validation momentum, this has given the participating companies the opportunity to deal with the technology of STEP implementation and to gain experience in applying the STEP approach.

This approach is mainly characterised by the fact that product information is passed through a neutral and standardised protocol. As STEP has the capability to address any product-related information, its application cannot be compared with neutral geometry exchange protocols. The latter cover an area of mathematically well-defined data that can be transformed between different types of representations, sometimes with loss of accuracy.

The big challenge for STEP is that most of the information about a product is generally not of geometrical nature. Examples are the hierarchical breakdown of the product, topological relationships between the parts of a product, configuration management information, material properties, design, and manufacturing information, etc. Like geometry, such product information is represented differently in different systems - especially at different semantic levels – but here no exact transformations exist between the different representations. An example is a stiffener on a panel whose design can be defined

- by explicit geometry – possibly by referencing geometry on a more global level like the shape of the bulkhead this plate is part of,
- by referencing adjacent structural elements - by this defining the boundaries in a more implicit way,
- by some parameters – perhaps taken out of a profile library or
- even by a combination of these.

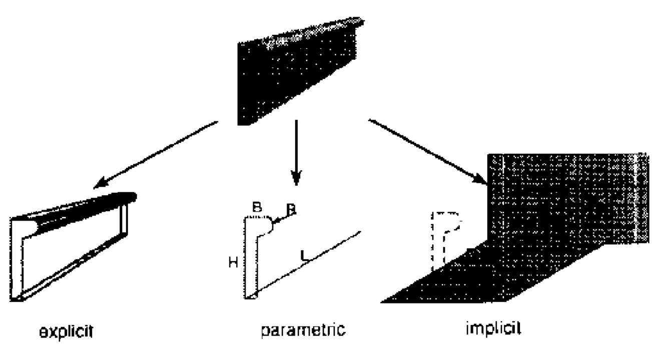


Figure 3: Defining the design of a stiffener

The AP that deals with steel structures for ships has to address all these possibilities in order to satisfy the requirements of a broad range of software systems that handle this data. The reason for the semantically different representation of the same piece of information is the independent development and the distinct nature of the different systems. To stay with this example, the information about a stiffener is handled

- by the CAD system of the shipyard's design department in which it is initially defined,
- possibly by the different CAD system of an external design office that got a subcontract to do the detailed design of the section of the ship containing the stiffener,
- by a system of a classification society that is applied to check the design against the classification rules,
- by a production preparation system to define the manufacturing design,
- by a material stock administration system that handles the ordering and supply of raw profiles
- by the software of numerically controlled machines that do the cutting, bending, positioning and welding,
- by a production planning system that is used to control the manufacturing process,
- by the ship owner's survey and maintenance system, that tracks its condition during the operational life of the ship.

The practical application of these systems when making and operating a ship – especially the need to pass information between the different steps in the life cycle – creates the input for the requirements that are documented in an AP. They also define the scenarios that are to be realised by its implementation.

Implementation History

The various projects mentioned above have addressed several such scenarios in implementing the ARM models of many of the ship APs. Results have been presented by on many occasions, for instance at ICCAS conferences in 1991 [21], 1994 [17, 18, 19], and 1997 [14, 15]. The focus within this paper is on the aspects of the practical application of STEP by such projects.

Early Initiatives and Projects

While the STEP effort itself started as early as 1983, relevant participation of shipbuilding related industries did not occur before 1986, when the NIDDESC [23] project has started in the USA. However, this project was not directly targeting STEP in the beginning, but as the PDES standardisation effort moved towards STEP, NIDDESC started to redirect its activities accordingly. First data modelling results were presented to the ISO working groups in 1989.

In Europe, one of the first projects to investigate the implementation issues of STEP for shipbuilding applications was the CAD-FE project [20]. It was targeting major improvements in the design and analysis cycles involving the creation of Finite Element models from CAD models. The relevance of validation through implementation was one of the key insights which led to the development on numerous implementation support tools, which did not generally exist at that time [21].

NEUTRABAS

This was the first ESPRIT project, funded by the European Commission which had a scope that combined the application of STEP (and the underlying technology) with information technology solutions for shipbuilding [22]. Apart from extensive data modelling contributions for the ISO work, this project included a range of prototype implementation tasks.

The project used (for the first time in this context) an implementation approach that used a late-binding interface style on top of a relational database. The complexity of the data model and the processing overhead resulting from the combination of the generic data storing (due to the late binding) and the relational technology lead to important conclusions. These have had a strong impact on following implementation attempts.

ITiS-B and Maritime

The first implementations of ship APs were done in two projects from 1992 to 1995. One of them – ITiS-B – was supported by the German Ministry of Research and Technology as part of the national ITiS programme. The second project Maritime was funded by the European Commission under the ESPRIT programme. On the STEP implementation side, they were both looking into the exchange of steel structure data between a shipyard and a classification society for the purpose of hull cross section approval. To achieve this, subsets of the AP218 ARM were identified that were able to accommodate the hull cross section data. This data included the structural system breakdown of the hull cross section into plates and profiles together with a respective geometrical representation. In addition, approval information was instantiated by the classification society as a result of the rule checking process.

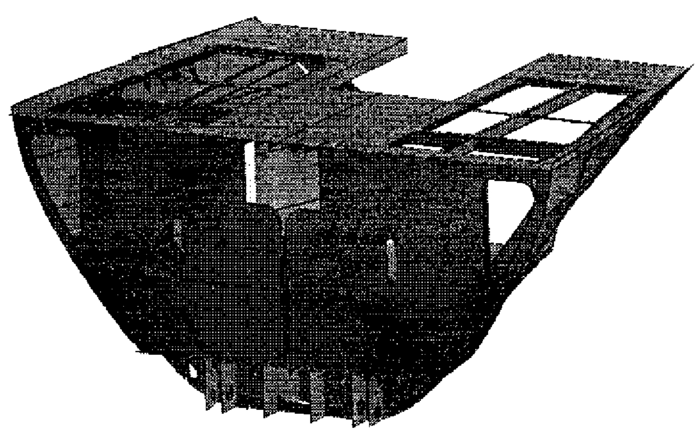


Figure 4: Hull Cross Section

In ITiS-B, structure information was exchanged between the former Bremer Vulkan yard using TRIBON Hull and Germanischer Lloyd using their Poseidon system. 2-D geometry was used to represent the hull cross section. A pre-processor was implemented to access the TRIBON database and extract the relevant information which was than exported into an ASCII file formatted according to part 21 of STEP (STEP Physical File). The generated file was passed to the GL system that had an implementation of a post-processor able to import it and put the data into Poseidon's native data storage. Once the hull cross section was checked against the class rules a report was generated and sent back.

The basic data model was shared between ITiS-B and Maritime. The latter however used a 3-D geometric representation of the structural parts within the hull cross section. Data exchange was done between TRIBON Hull used by Bremer Vulkan and the PILOT system, the former rules checking system of Det Norske Veritas. Both systems had a pair of pre- and post-processors implemented, allowing for a round trip exchange; yard – classification society – yard. The project had also produced a Visualisation Tool that allowed for an analysis of the received STEP Physical Files before the information was loaded into the native databases. This allowed to identify which plates and profiles were rejected by the class society due to non-conformance to their rules. For those parts, information about the re-

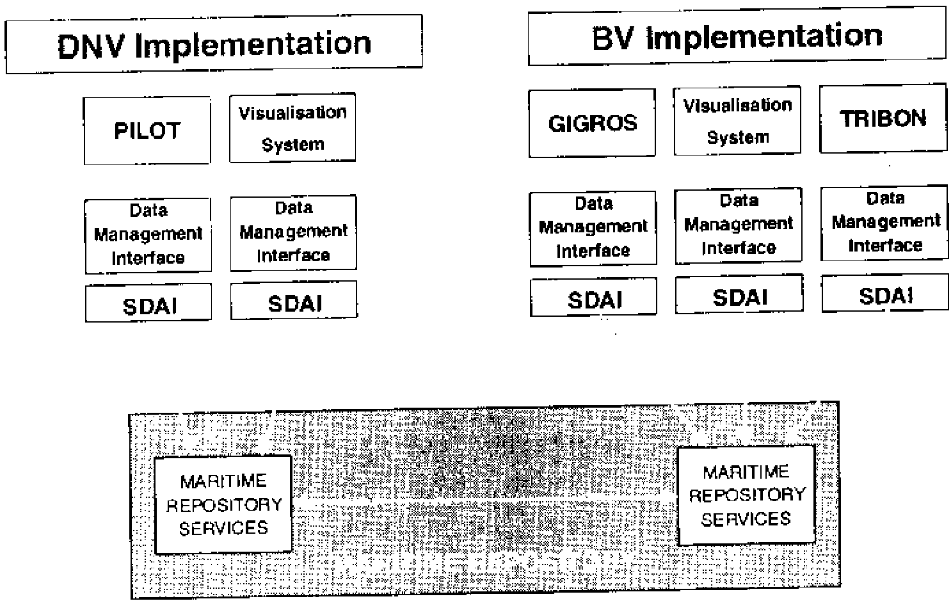


Figure 5: Maritime Scenario

quired changes (e.g. required plate thickness) was sent by the PILOT system as part of the data exchange and the receiver could consider updating the original data with the required changes.

ShipSTEP and Calypso

The two research projects ShipSTEP (1994 – 1996) and Calypso (Brite/EuRam, 1996 – 1999) addressed the exchange of hull form geometry based on AP216 (ship moulded forms). The main objective was the integration of hull form design systems and systems that are used to perform CFD analysis. In a typical scenario a shipyard has done a hull form design and wants this to be analysed by a model basin in order to optimise the shape for minimal resistance and best propeller current. The resulting optimised hull geometry is sent back to the yard where it is taken as input for the steel structure design as well as input for a survey system operated by the ship owner or a classification society. In the scope of these projects implementations of pre- and post-processors were done for systems like NAPA, TRIBON Initial Design, GMS (MARIN), FLOWTECH and SEASAFE (LR).

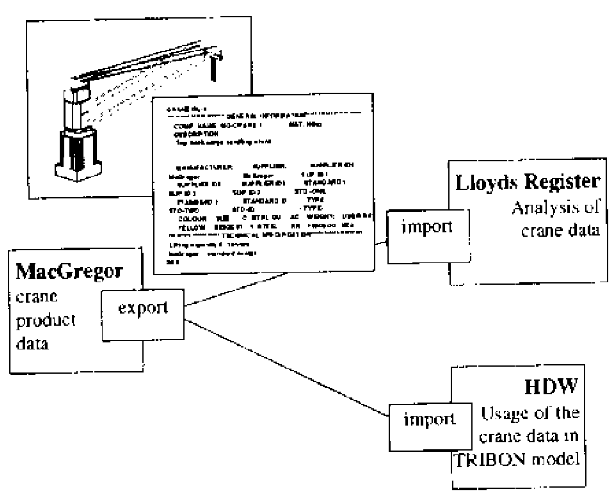


Figure 6: Edimar Scenario

Edimar

In combining business with engineering data Edimar – another EU project funded from 1996 to 1998 – has tried to combine both

EDIFACT and STEP. This was done by transferring STEP based information as the content of EDIFACT messages. The transported data – a STEP Physical File based on an AP226 subset with information about a deck crane – was sent from the crane manufacturer to the yard where it could be used as an externally supplied component during the design process. For that purpose, a post-processor was implemented to pass the crane information into the component database of the TRIBON system.

Maristep

Maristep is a project sponsored by the US DARPA MARITECH program targeting prototype implementations of the shipbuilding APs 215 through 218. It started in 1996 and involves the major US ship yards together with their CAD system vendors such as Electric Boat Corp. (using CATIA), NASSCO (using TRIBON), Newport News Shipbuilding (using Vivid and GS-CAD) and Ingalls (using CADDS 5). Over three years significant effort was spent to enable data exchange capabilities between the yards to improve their ability to co-operate. The main scenario in the project is the joint collaboration of the shipyards, which requires data exchange at all stages of design and production. For this purpose all CAD system suppliers implemented pre- and post-processors for subsets of the four APs.

Seasprite

The latest European initiative is the Seasprite project also funded under the ESPRIT programme. As basis for the processor implementation, it uses four exchange scenarios:

- exchange of hull form geometry between shipyard and model basin (AP216 based)
- exchange of steel structure early design data between shipyard and classification society (AP218 based)

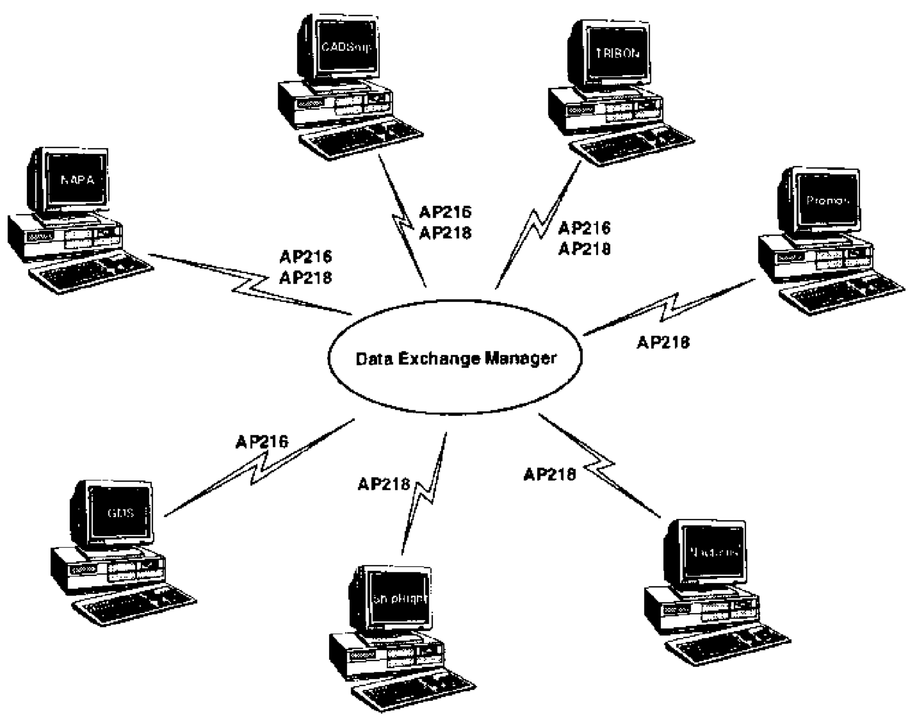


Figure 7: Seasprite scenario

- exchange of steel structure detailed design data between shipyard and engineering or production subcontractor (AP218 based)
- exchange of steel structure "as built" data between shipyard and ship owner

The European Maritime STEP Association (EMSA) initially defined these scenarios as Business Process Descriptions before they were temporarily handed over to Seasprite for further development and application. Their implementation involved several systems including CAD, classification, and CFD software.

The limited length of this paper does not allow for a complete presentation of all STEP applications in the shipbuilding area. It is worth mentioning that there were activities in the Japanese shipbuilding sector (GPME, NCALS) doing implementations of AP203 and AP216. A Korean project (KS-STEP) has started in 1998 and aims to implement an AP218 subset. Follow-on projects in Europe (Seanet, Marvin) are currently applying further the results of the projects described above.

Implementation Technology

The examination of the STEP technology has offered a deeper insight and understanding of the underlying mechanisms. Before developing STEP based software, several decisions must be made with respect to the implementation strategy. These decisions influence the toolkit choice because the toolkits available today do not support all possible options defined by the STEP Implementation Methods.

Principle configuration of a STEP based application

The principle configuration of a STEP based application consists of three main components:

- the repository that holds the instances of the data model (as defined in an application protocol)
- the binding to the repository providing the Standard Data Access Interface (SDAI) functionality as an Application Programming Interface (API) in a certain programming language [3, 4, 5, 6, 7]
- the application that uses the SDAI functionality in order to manipulate the instances in the repository

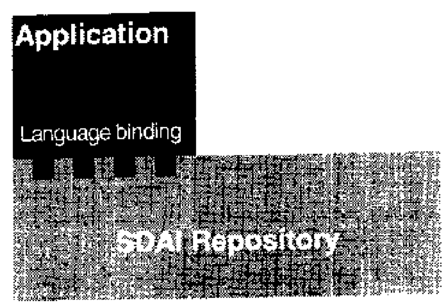


Figure 8: Principle configuration of a STEP based application

STEP Implementation Scenarios

From the theoretical viewpoint, two different basic implementation scenarios exist for STEP:

- data sharing and
- data exchange

In the data-sharing scenario potentially many applications access the same repository concurrently and share the data model instances. This occurs for example when the repository acts as the data storage for these applications.

The data exchange scenario is the file-based approach. The data model instances are exported to or imported from an ASCII file that is formatted according to part 21 of STEP [1] (known as STEP Physical File) and reside only temporarily in the repository. Due to its nature, this file is machine independent and can thus serve as an exchange medium between two STEP based applications.

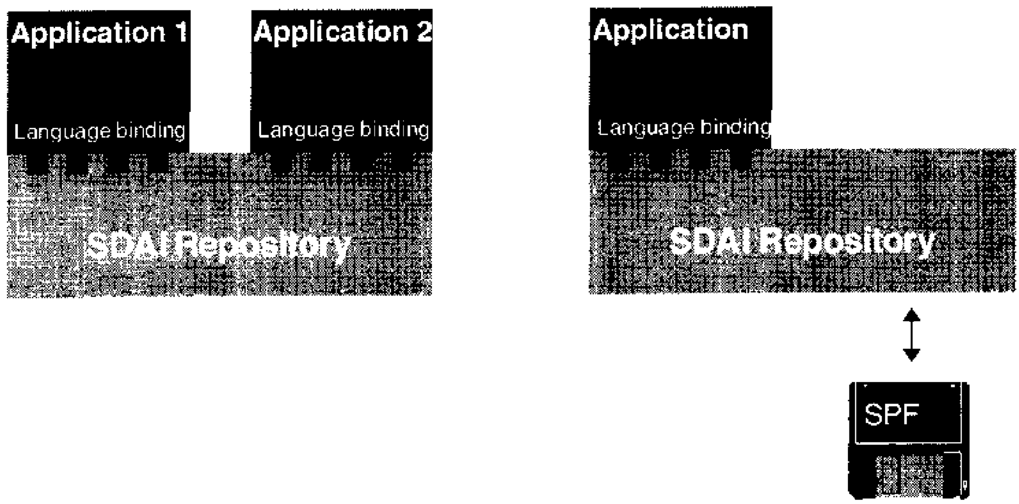


Figure 9: STEP Implementation Scenarios

In fact, all STEP based implementations described above read and write STEP Physical Files and thus follow the exchange scenario.

Persistent or volatile data storage

The second decision to be made is with respect to the type of storage used below the SDAI interface layer. Basically, two different approaches are available:

- persistent data storage and
- volatile data storage.

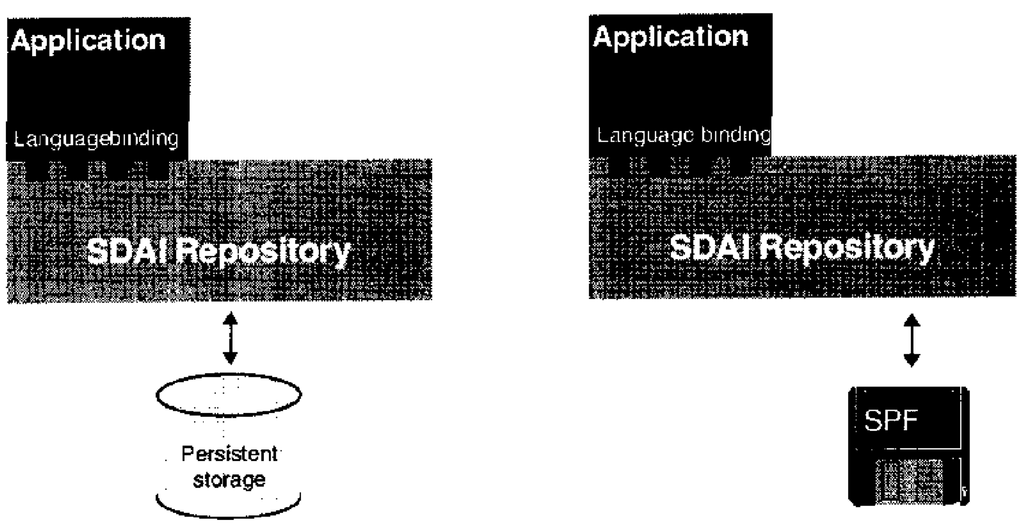


Figure 10: SDAI based application with and without persistent storage

Today's operating systems provide virtual memory management that allows the allocation of much more memory than is physically available in the system and already this ranges to hundreds of megabytes for standard computer hardware. This makes it unnecessary to persistently store data model instances if they have only a temporary lifetime in the repository. Therefore the second decision mainly depends on whether the implementation follows the data sharing or the data exchange scenario.

In case a STEP processor accesses a CAD system's database and creates instances of the data model in order to write them into a STEP physical file, a persistent storage of this data would unnecessarily slow down the process. This would make the processor less handy to use and more expensive. However, for a solution where the repository is located in the CAD system's database, persistent storage is of course required.

Early or late SDAI binding

Thirdly, the kind of the language binding to the repository is to be decided. STEP defines two types of language bindings:

- early binding and
- late binding.

Early or late binding refers to the time when the relationship between the data structures defined in the application protocol and the software is created: either during compile time or during run time.

Early binding should be used if the application protocol is known and remains the same during the lifetime of the software or where no dynamic change of data models during runtime is required. In this case the application protocol is 'hard coded' within the application. The advantage of this approach is that the SDAI will be implemented as an API that 'speaks' the language of the implemented application protocol as it is completely generated from the EXPRESS model.

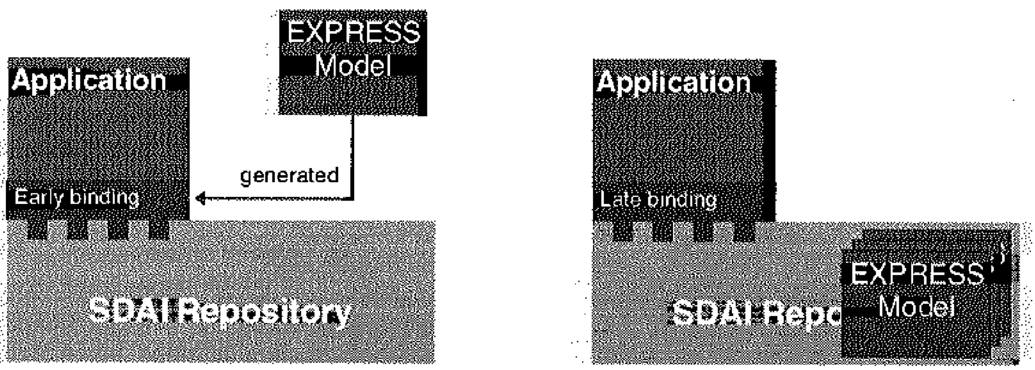


Figure 11: SDAI based application with early or late binding

The early binding approach does not allow to implement generic applications such as model independent browsers, since the API exposes a specific data model. However, meta-data information as such can be obtained.

Late binding, in contrast, offers the capability to create software that is flexible with respect to the contents of the application protocols in use. Here, the application protocol is not known during compile time. An SDAI late binding is a generic API that provides functionality to query for meta-model information of the currently loaded application protocol and to create or manipulate instances of it.

The usage of the late binding technique for a STEP processor development would consequently be not really late binding because the processor must have information about the application protocol in order to be able to create or evaluate instances of it. However, once information about an application protocol is hard coded, it can't be changed to another one without updating the processor implementation. The flexibility of the late binding approach would be lost.

Implementation language

The implementation language of the SDAI binding layer is the fourth decision to be made. That is the programming language that is used to access the repository. STEP currently offers four languages for the SDAI:

- an early C++ binding (part 23 [4]),
- a late C binding (part 24 [5]),
- an early IDL binding (part 26 [6]),
- an early Java binding (part 27 [7]).

The development of a Fortran binding (the former part 25) was discontinued in 1997.

The binding decision can therefore influence the implementation language and vice versa.

EXPRESS-X

Sometimes it can be necessary to transform the information represented by instances of one data model into those of another. For this reason, a data mapping capability on EXPRESS level is under development. The mapping language describes the transformation between two EXPRESS models. It works in such a way that once an EXPRESS model is instantiated, another one can be instantiated automatically by applying the transformation rules described using EXPRESS-X.

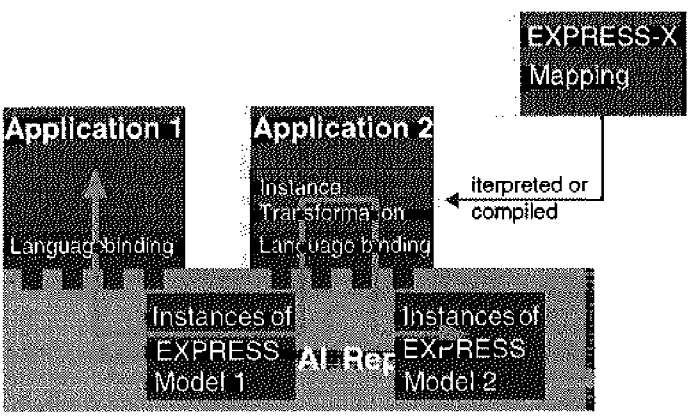


Figure 12: EXPRESS-X Approach

Possible scenarios are the implementation of the ARM of an AP and a mapping to the AIM or the implementation of an (non-standard) EXPRESS model of the native database of the system and a mapping to the (standard) application protocol.

Apart from the fact that the EXPRESS-X implementations today are often working as interpreters this approach requires the same set of information to be instantiated twice. This raises the efficiency question for data models that are not reasonably small. The future will have to show if this - admittedly interesting - approach will become practically relevant for processors.

Conclusions

The participation in the various research projects presented the opportunity to vary the implementation strategy and investigate different configurations. Two binding types, late C and early C++ were used for implementations where the authors have been involved.

The implementations of AP215 and AP216 based processors utilised the late C binding approach. It was combined with the EXPRESS-X methodology to do a mapping from the native data structures of TRIBON Initial Design to the implementation subsets of the two APs and vice versa.

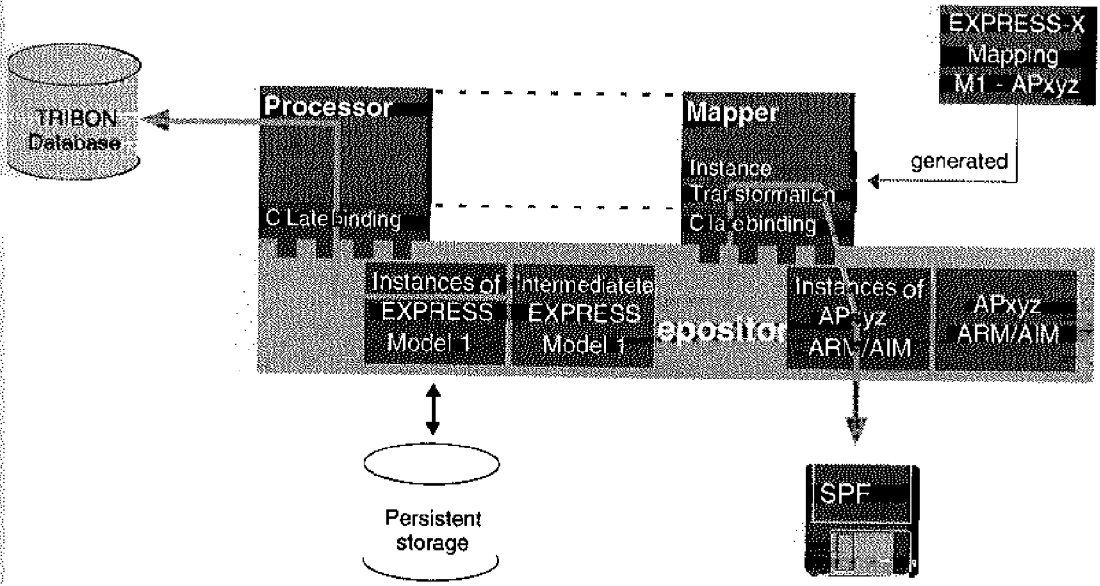


Figure 13: KCS Implementation using C-late binding SDAI and EXPRESS-X

The early C++ approach was applied for the implementation of AP217, AP218, and AP226 based processors. Both, transient and persistent storage of the repository was used. The persistent re-

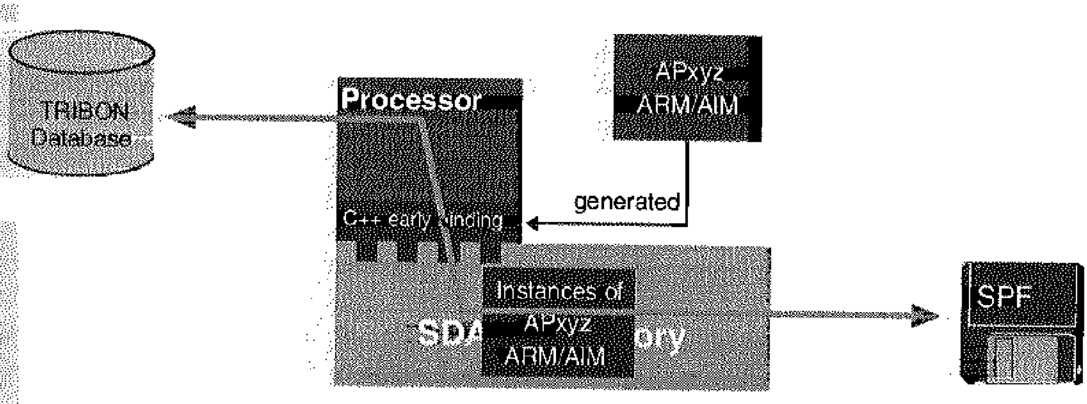


Figure 14: KCS Implementation using C++-early binding

pository based processor implementations proved to be significantly slower than those using a volatile repository.

A mechanism similar to EXPRESS-X was tried out for the early binding based implementations providing a mapping capability between two C++ APIs. One of the APIs was the native database interface of TRIBON, the other one an early bound SDAI. This approach combines the flexibility of a mapping language with the performance of a compiled programming language using an early binding and proved to be very efficient.

This paper has given an overview of the status of the STEP development in the shipbuilding area as well as different aspects of its application. It has shown that the development of the standard is still not finished but in a state where it can be practically applied. Several of the companies that are involved in current implementation projects have recently reported inquiries and requests from their customers for STEP based data exchange solutions. This shows that a commercial exploitation of the technology has started and that the application of STEP in the maritime industry is no longer in its infancy.

Abbreviations

AP	Application Protocol
AIM	Application Interpreted Model
ARM	Application Reference Model
CD	Committee Draft
CDC	Committee Draft for Comments
EMSA	European Maritime STEP Association
NIDDESC	Navy / Industry Digital Data Exchange Standards Committee
IR	Integrated Recourses
SDAI	Standard Data Access Interface
STEP	Standard for the Exchange of Product Data (ISO 10303)

References

1. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 11: EXPRESS Language Reference Manual
2. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 21: Implementation methods: Clear text encoding of the exchange structure
3. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 22: Implementation methods: Standard data access interface specification
4. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 23: Implementation methods: C++ language binding to the standard data access interface
5. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 24: Implementation methods: C language binding to the standard data access interface

6. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 26: Implementation methods: Interface definition language binding to the standard data access interface
7. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 27: Implementation methods: Java programming language binding to the standard data access interface
8. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 215: Application protocol: Ship arrangement
9. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 216: Application protocol: Ship moulded forms
10. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 217: Application protocol: Ship piping
11. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 218: Application protocol: Ship structures
12. ISO 10303 Industrial automation systems and integration – Product data representation and exchange Part 226: Application protocol: Ship mechanical systems
13. ISO 10303 Industrial automation systems and integration – Product data representation and exchange: AP Development Guidelines for Shipbuilding
14. Rabien, Uwe & Langbecker, Uwe, "Practical Use of STEP Data Models in Ship Design and Analysis", ICCAS '97, Yokohama, Japan, 1997
15. Kendall, J. & Hansund, K., "Product Data Integration and Exchange for Ship Design and Operation", ICCAS '97, Yokohama, Japan, 1997
16. Fowler, J., "STEP for Management, Exchange and Sharing", Technological Appraisals, 1995
17. Metha, S & Lehne, M., "Product Data Technology Benefits – A Perspective of Yard and Classification Society", ICCAS '94, Bremen, Germany, 1994
18. Wooley, D., "Configuration Management of a Ship Product Model", ICCAS '94, Bremen, Germany, 1994
19. Langbecker, U, Nowacki, H., Bühr, W. & Kress, H., "A reference Architecture for Shipbuilding Product Models", ICCAS '94, Bremen, Germany, 1994
20. Grafe, W., "Datenextraktion und Bereitstellung im Neutralformat", Schlußbericht zum BMFT Verbundforschungsvorhaben Zukünftige Dimensionierung und Gestaltung der Schiffskonstruktion, Technische Universität Hamburg-Harburg, Arbeitsbereich Schiffstechnische Konstruktionen und Berechnungen, Hamburg, 1993
21. Koch, Th., "STEP-based Modelling of Ship Product Definition Data", ICCAS '91, Rio de Janeiro, Brazil, 1991
22. Nowacki, H. (Ed.), "NEUTRABAS", Springer-Verlag, Berlin, 1991
23. Geraldi, M. L., "Application Reference Model for Ship Structural Systems", Working Draft of Part 102 of ISO 10303, ISO TC184 / SC4 / WG1, 1990
24. ISO SC4 web page: www.nist.gov/sc4
25. EMSA web page: www.emsa-bbs.org
26. Seasprite web page: www.lrtpdd.demon.co.uk/SSPWWW.HTM#Partners
27. Maristep web page: www.intergraph.com/federal2/projects/step
28. Edimar web page: www.biba.uni-bremen.de/projects/edimar/index.html

SHIP PRODUCT DATA INTERCHANGE TO SUPPORT SHIP DESIGN PROCESSES

John Kendall, Lloyd's Register of Shipping, London, UK

Introduction

The past few years have seen significant advances in the development of the data exchange and sharing technologies that are available. The industry must take advantage of these to enable the increasing need for co-operative working, concurrent engineering and decision support.

The paper will demonstrate the results of two related virtual enterprise development projects. The EU funded SEASPRITE project and its successor SEANET not only aim to introduce technologies to reduce the lead time for design, but also to minimise the cost implications during the design and manufacturing processes of changes in earlier stages. The systems, which have been developed and implemented in commercial systems, are all based on open technologies such as the internet, the ISO Standard for the Exchange of Product Model Data (STEP), CORBA and COM.

The underlying requirement for an efficient implementation is the development of a complete Ship Product Model to act as the basis for the data exchange. The STEP development work in SEASPRITE has resulted in considerable advancement of the standard within the shipbuilding domain to provide a stable basis for implementation. The business case descriptions for these models have resulted in their publication by the European Marine STEP Association as EMSA Protocols. These are STEP based industry standards for the exchange of information within narrow application domains.

Translators have been built, based on these Protocols, in commercial products, by companies both within and without the project. These translators have then provided the basis for the work in SEANET which aims to produce an open data sharing environment based on the storage of information in a neutral format. The core of the implementation is the PETS(TM) Product Data Management System, which has been configured with the STEP Ship Product Model. The project then provides application views of the data via the STEP translators and interoperability views with other standards based on EXPRESS-X mapping technology. Workflow and security are also core technologies in this implementation.

The systems described will enable the efficient management and transfer of ship hull data throughout the ship lifecycle; from initial design through model basin tests, plan appraisal, detailed design, production and survey.

Data exchange

STEP Development

The development of STEP within the shipbuilding industry has taken nearly ten years to reach the current position where the core group of APs are all reaching the first ISO ballot stage. The majority of this development has been performed by the US NIDDESC consortium, and by several EU funded research projects. This work has been described in previous papers, and I do not intent to re-iterate it here [3].

The current project undertaking the work on the development of APs to cover the exchange of ship hull information is SEASPRITE. It is due to complete in June 1999, and will see the three APs for Moulded Forms, Ship Arrangements and Ship Structures nearing the Committee Draft stage of development.

The recent work to achieve this goal has focused on the development of the Application Interpreted Model. This requires that the information requirements that have been developed be mapped onto the generic resource parts of STEP. This process produces the standardised model and is known as interpretation. It involves the participation of external experts who not only know the resource parts of STEP very well, but also know how they have been mapped in the past. This will ensure the objective of interpretation, that of 'AP interoperability'. That is to say, the ability for APs from one domain to work with APs from another.

In this way the geometry created by a CAD application for the automotive industry will be able to be read by an application in the shipbuilding domain. This is important when parts vendors supply many industries, and also when projects cross industry borders. One particular example is the manufacture of FPSO (floating, production, storage and offloading) vessels. These are ships that have oil and gas facilities on the topside. A total data exchange solution for an FPSO would therefore require data exchange according to AP215, 6, and 8 from the shipbuilding domain, as well as AP221 from the process plant domain.

The majority of the interpretation work to date has been in the generation of the mapping tables that specify how the information requirements are mapped onto the STEP generic resources (so called Part 40 series). The shipbuilding STEP development team have had a distinct advantage over other groups in the field of STEP development, namely the building block (BB) approach to AP development and the Ship Common Model (SCM) [1].

The former allows the mapping table work to be distributed amongst those people who have developed the BBs. The latter allowed a series of mapping templates to be developed that ensured that once a construct had been mapped, other similar constructs could use the template mapping

This interpretation process is however still a long one, and will mean that standardised STEP exchanges in shipbuilding are some years away. However the companies involved in the development of shipbuilding STEP have adopted an implementation strategy so that they can gain business benefit from the work immediately.

STEP Implementation

The main STEP implementation efforts at the moment are the EU funded SEASPRITE project and the US MARITECH funded MariSTEP project. Both are focussing on the core APs for Arrangements, moulded forms and structures, with MariSTEP additionally implementing AP217 – Ship Piping Systems. The implementations however are being based on the information requirements models. In order that these can become stable and internationally accepted, the business cases supporting the implementation efforts were handed over to the European Marine STEP association and will be published as EMSA protocols [1].

EMSA protocols are the information requirement s for data exchange within narrow business case domains. EMSA protocols are base don exact subsets of the official STEP APs and will be withdrawn when STEP publishes APs covering their scope.

The EMSA protocols currently under development are exchange of ship data to support:

- Hull form optimisation;
- Hull design approval;
- Subcontracting during design and manufacturing;
- Hand-over of as-built condition;

These EMSA protocols specifications are explained in greater detail in by T.Turner [1]. The companies currently implementing these EMSA protocols and the current state of the art will now be described in the following sections. For the SEASPRITE project we are using as a test ship the Eleo Maersk built by Odense Steel Shipyard, the first double hull VLCC.

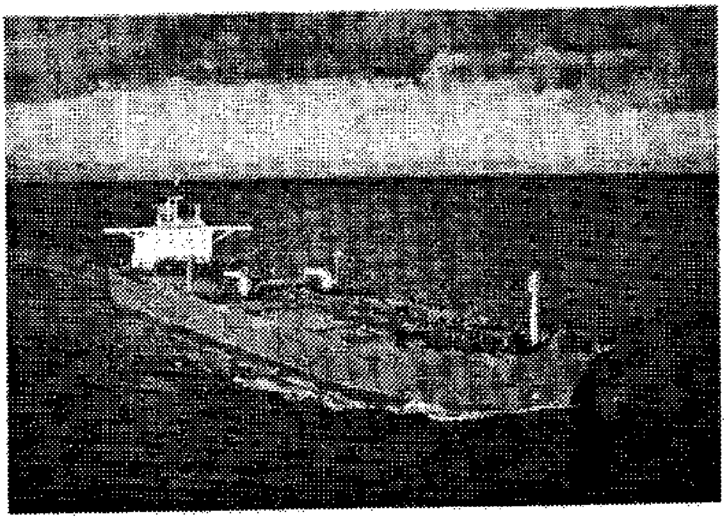


Figure 1. Eleo Maersk

Hull Form Optimisation

The business case behind this protocol is the exchange that is needed between the shipbuilder or design agent and a test model basin or CFD package during the design of the hull form.

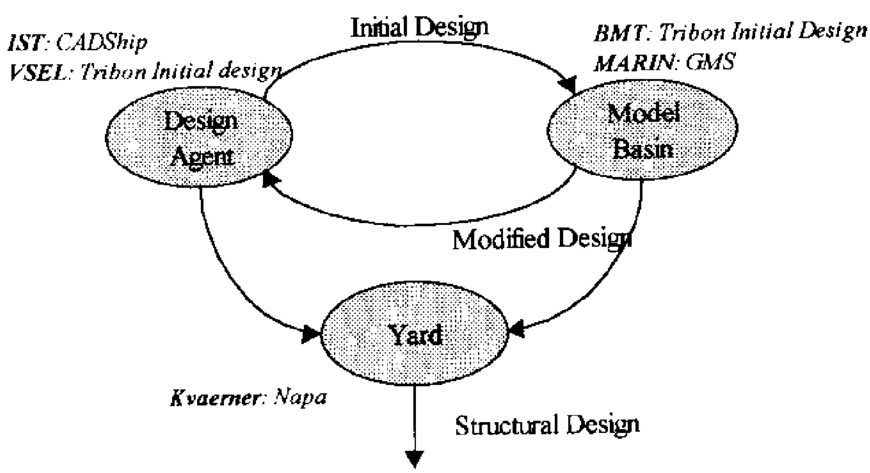


Figure 2. Hull form optimisation scenario

This business case based on AP216 for ship moulded form. The companies who are imlementing this business case within SEASPRITE and their roles are listed in table X below.

Table 1. Hull Form Optimisation Implementations

Company	**Role**	**Software**	**Platform / Operating system**	**Capability**
KCS	Software vendor	Tribon Initial Design	PC/NT	Import/Export
MARIN	Model basin	GMS	SGI/Irix	Import
Napa	Software vendor	Napa	PC/NT	Import/Export
BMT	Model basin	Tribon Initial design	PC/NT	
IST	Software vendor	CADShip	PC/NT	Import/Export

The systems listed above all have working translators to import or export the hull form specification and associated parameters.

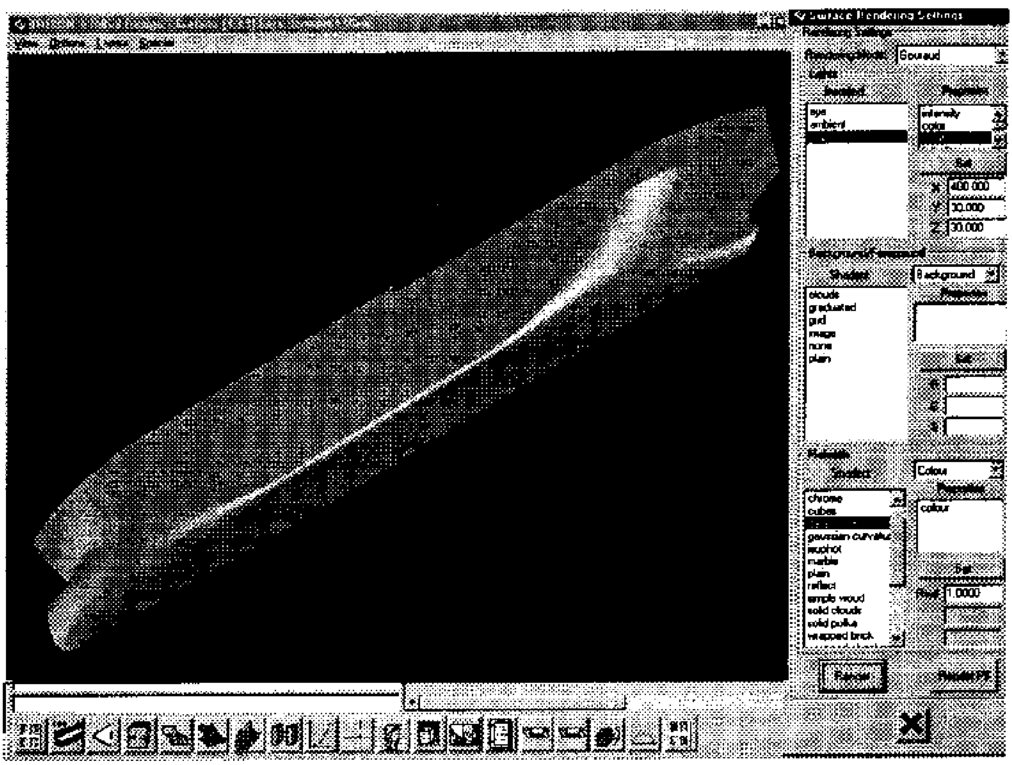


Figure 3 Hull form after import in Tribon Initial Design

Once the data has been transferred to the model basin or CFD analysis package the protocol allows for the ability transfer any modifications back to the shipyard. The transfer of information to support CFD is being carried out by the EU funded CALYPSO project, who are also using the same EMSA protocol.

Hull Design Approval

When the preliminary design has been completed, the design is sent to the classification society for approval against their rules and regulations. This protocol allows for the approval of both the hull cross section scantlings and the whole ship by direct calculations such as finite element analysis.

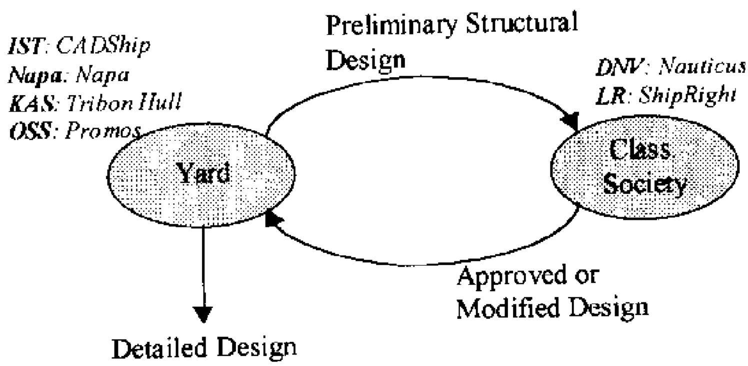


Figure 4. Hull Design Approval scenario

The majority of the business case is based on the AP218 but also includes parts of AP215 to cover the exchange of compartmentation information. The current implementations are listed in table X below.

Table 2. Hull Design Approval Implementations

Company	Role	Software	Platform / Operating system	Capability
KCS	Software vendor	Tribon Hull	Sun/Solaris	Import/export
Napa	Software vendor	Napa	PC/NT	Import/export
LR	Classification society	ShipRight[IS] LR-STRAND(FEA)	PC/NT	Import
DNV	Classification society	Nauticus	PC/NT	Import/export
OSS	Shipyard	PROMOS	SGI/IRIX	Export
Kvaerner as	User Shipyard	Tribon Hull	Sun/Solaris	Import/Export

The translators developed above have meant that the Classification Societies have been able to import the hull cross section into their respective systems and can perform rule based and direct calculations on it. Where implemented approval information can then be exported and re-imported into the originating system, where the design can be amended if the design has not passed the classification society rules.

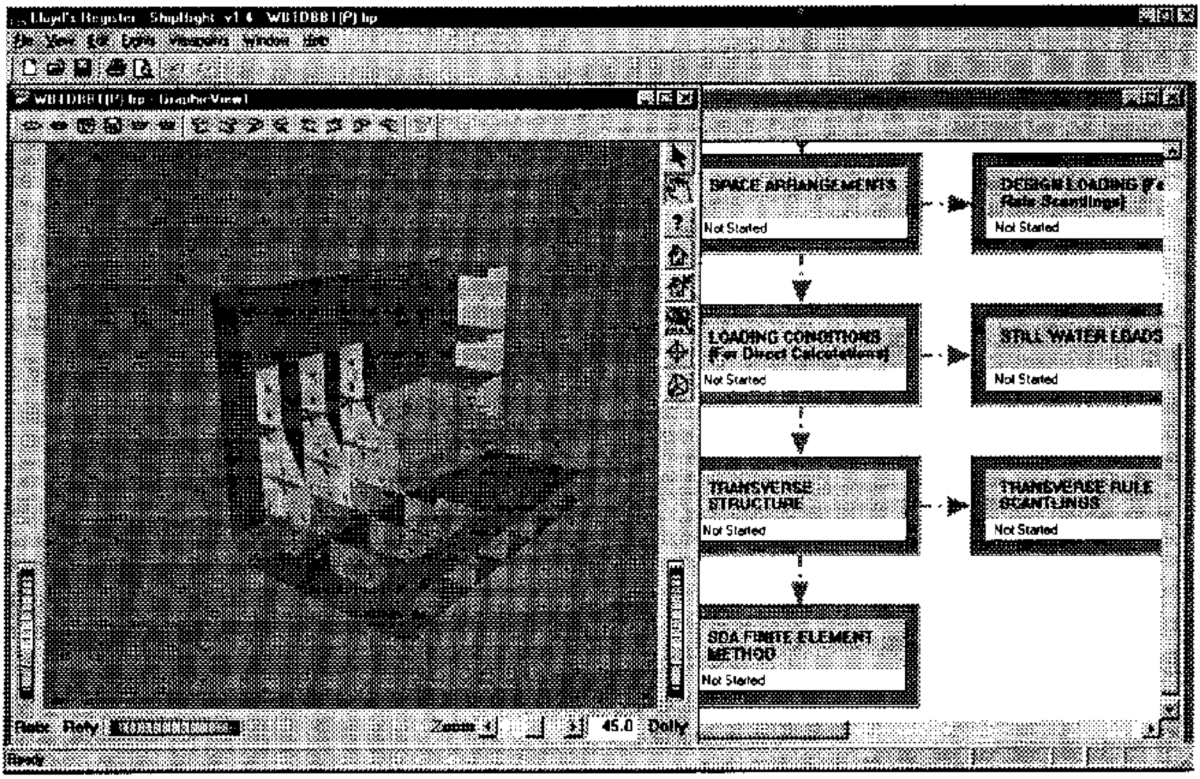


Figure 5. Hull Structures Imported into ShipRight[IS] for FEM Analysis

The SEASPRITE project has signed co-operation agreements with several projects that aim to implement STEP. The Korean Ship STEP project is working in several areas to develop and implement STEP. In this they are using the SEASPRITE models to develop a STEP data exchange capability between Korean shipyards and Korean Register. This therefore widens the scope for usage from Europe to a global capability, allowing the European classification societies to work closer with the Korean yards.

Design and Production Subcontracting

The European shipbuilding industry involves more and more participants in the lifecycle of a ship. The reason for this is that European shipyards cannot compete with the costs of developing simple ships such as bulk carriers against their Far Eastern rivals, and so the European yards are focusing on more complex ship types such as passenger craft. The development of more specialised ships requires more outsourcing of design and production to specialist engineering companies. These companies will be geographically disparate requiring implementation of virtual enterprises.

At the level of the individual yard the efficient sharing and distribution of product model information is handled by Product Data Management (PDM) systems. These systems support concurrent engineering practices by allowing engineers access to up to date versions of designs, routing information using workflow tools, and protecting against unauthorised changes to data. However, between collaborating partners, communications are still performed in an ad hoc manner, details of ship designs are often sent to incompatible systems, which then require manual adjustment of the data. Version control is very basic and often unstructured, and concurrent engineering involving multiple firms is hard to achieve.

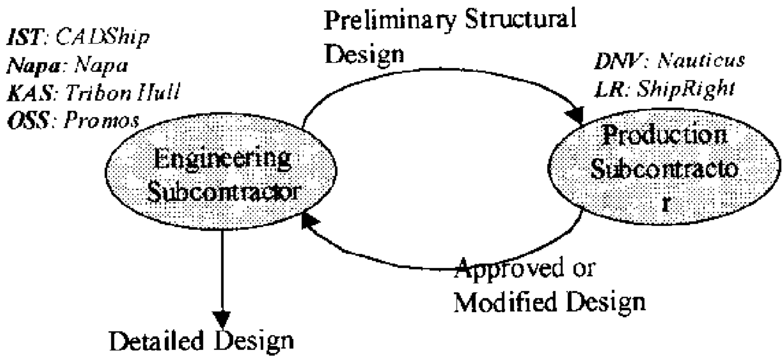


Figure 6. Engineering and Production Subcontracting scenario

SEASPRITE has aimed to overcome some of these problems by developing a model which supports data exchange between partners involved in the design and production of the ship. The model contains the capability to exchange the ship structural details as well as versioning and change control information. In this way the cost of changes to the design can be minimised by the rapid and controlled exchange of information.

Table 3. Engineering and Production Subcontracting Implementations

Company	Role	Software	Platform / Operating system	Capability
KCS	Software vendor	Tribon Hull	Sun/Solaris	Import/export
Napa	Software vendor	Napa	PC/NT	Import/export
OSS	Shipyard	PROMOS	SGI/IRIX	Export
Kvaerner as	User Shipyard	Tribon Hull	Sun/Solaris	Import/Export

Both the shipyard in this effort a part of a group of yards, which regularly subcontract work within their respective groups. For example the superstructure and hatch covers are regularly subcontracted out by the prime yard building the hull.

Handover of As-built definition

Many activities during the operation and maintenance of the vessel depend on information that is created during the design. In most cases this information is re-keyed into systems. The business case for handover of the as-built definition aims to allow the reuse of information through out the life of the ship. Scenarios that have been identified during the SEASPRITE project are the configuration of Hull Condition Monitoring Systems for the recording and display of survey data, and the configuration of stability analysis packages for use during Emergency Response situations. The project has implemented the exchange of as-built condition to configure LR's Hull Condition Monitoring system (ShipRight HCM). This can then be used for planning surveys, recording condition data and producing condition reports.

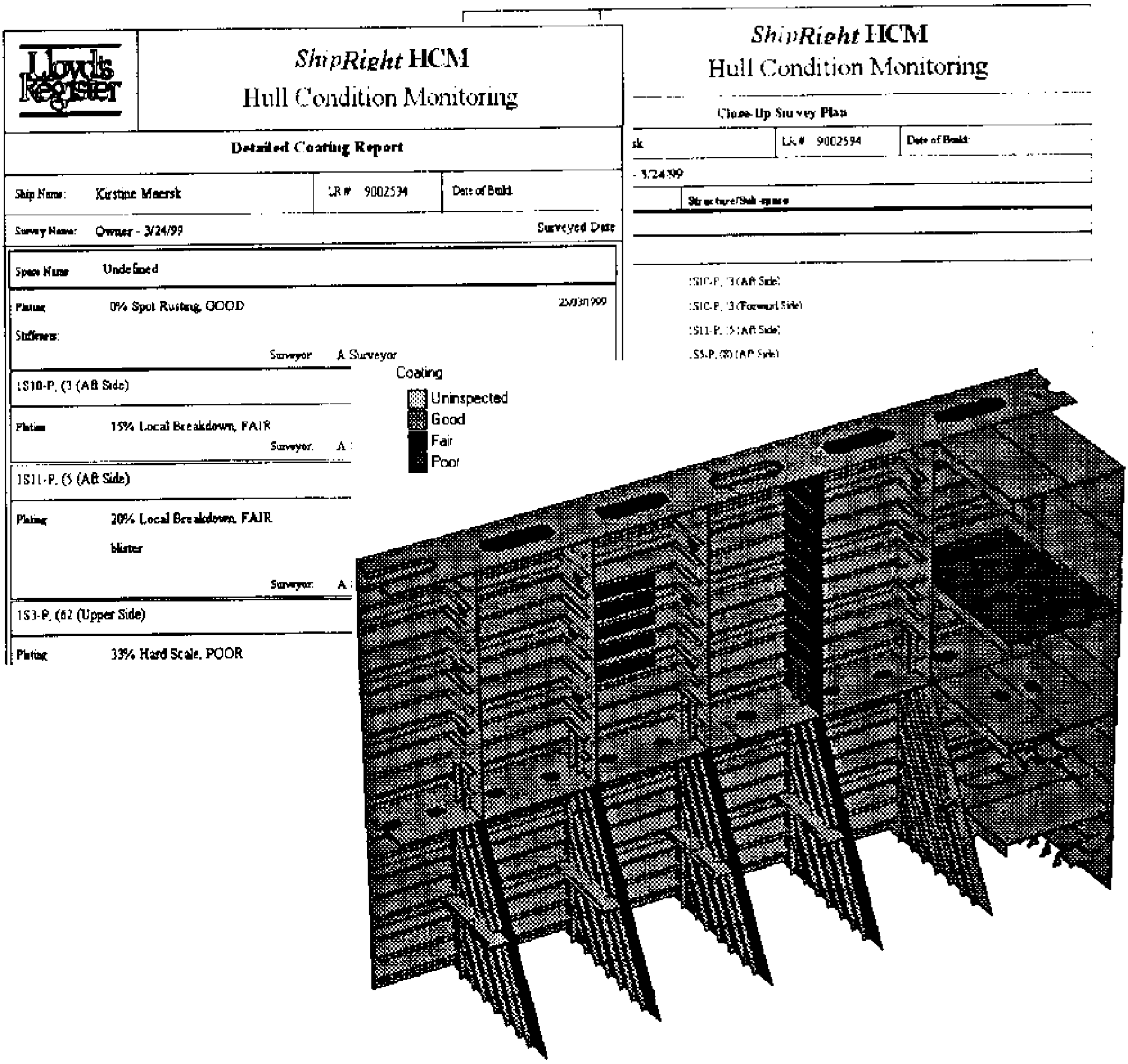


Figure 7. As-built hull data in ShipRight HCM with Survey Planning and Hull Condition Reports

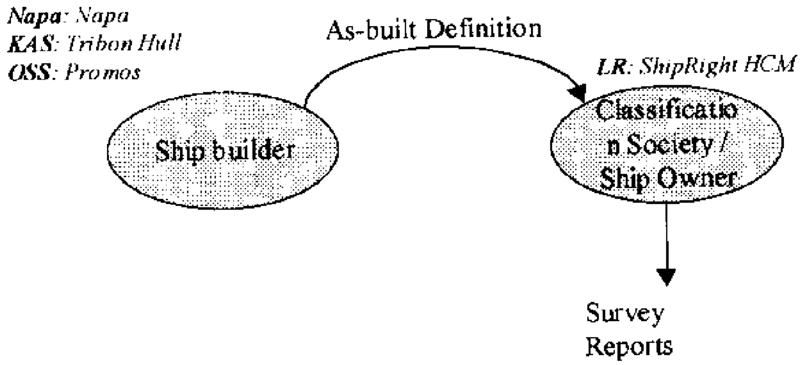


Figure 8. As-built hull data flows

The implementation within SEASPRITE has successfully showed the transfer of the as-built definition of the ship. This has been used to configure the HCM system and produce survey planning reports, and survey reports when condition data has been recorded against it.

Table 4. As-built definition implementations

Company	**Role**	**Software**	**Platform / Operating system**	**Capability**
KCS	Software vendor	Tribon Hull	Sun/Solaris	Import/export
Napa	Software vendor	Napa	PC/NT	Import/export
OSS	Shipyard	PROMOS	SGI/IRIX	Export
LR	Classification Society	ShipRight Hull Condition Monitoring	PC/NT	Import

STEP Exploitation

The above STEP development and implementation work has all been carried out in EU and industry funded research projects, and has been in development for nearly 10 years [1]. In order to exploit this large amount of work the companies involved are now preparing to implement STEP for commercial use. This is based on the AP subsets that are being published by EMSA as EMSA Protocols. This exploitation effort is not limited to Europe, there are implementations being developed in Korea and the USA which are also being based on these EMSA protocols.

The translators which have been developed as part of the SEASPRITE project are now undergoing business case testing, and will be completed by the end of the project in June 1999. At which time they will have been demonstrated at several relevant exhibitions and conferences. This will help to publicise that STEP is now ready for commercial exploitation within the maritime industry.

Data sharing

Standardised data exchange, however important, will not solve the problems of associative working. SEASPRITE has addressed this by facilitating the exchange of files between partners. This has resulted in the DEM (Data Exchange and Management Architecture).

DEM

The DEM is a CORBA based system that allows the controlled exchange of packets of information.

It allows version control, project management, data publication and data retrieval. All data storage and retrieval mechanisms are invisible to the user, so that to all partners in a project it appears as if information is local. The system is described in more technical detail in [3]

The user interface to the DEM architecture is the information browser that allows users to access information that has been released to a project. This information browser has been implemented as a standalone Java application and a Java applet, allowing the DEM to be accessed by participants without the need for lengthy installation of specialist software.

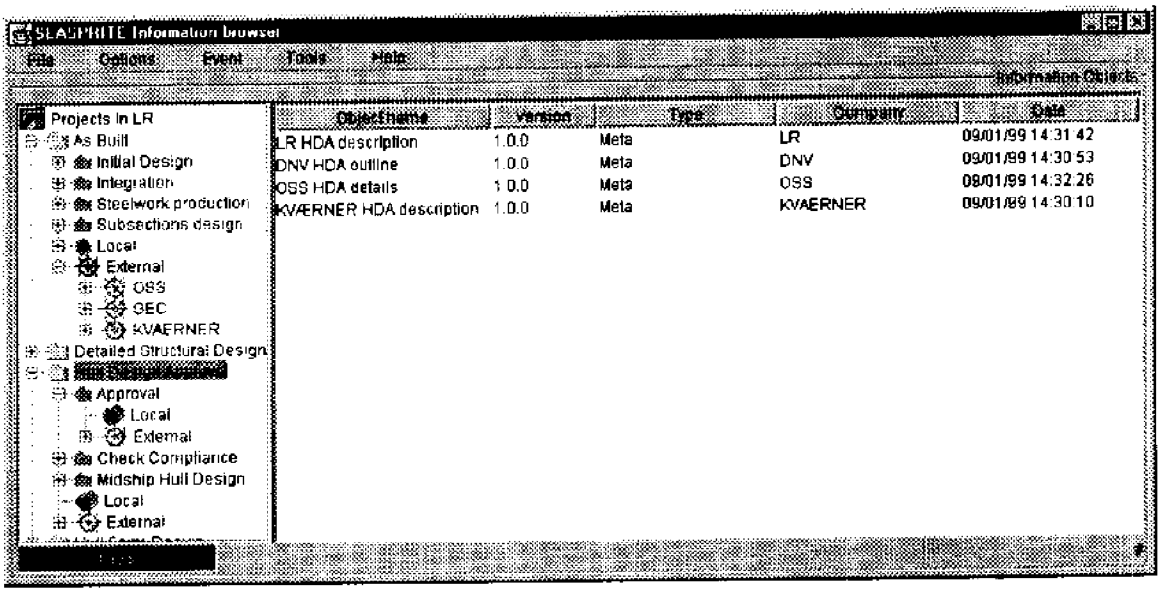


Figure 9 – The SEASPRITE Information Browser

The DEM, although managing the access and transport of project related data, it only does so according to the 'packets' of information as they are released. At the simplest level the packet would be a file, but also could be information resulting from a database query.

There needs to be active management of the information at the lowest level as it is exchanged between the partners involved ship design and build processes. In other words within the packets currently managed by the DEM. To this end the SEANET project was started to build on the work already carried out in SEASPRITE and develop a more advanced product data management system.

The SEANET Architecture and tools

SEANET is creating an open and extensible architecture by exploiting existing and developing new IT tools to enable efficient management, distribution and data exchange between companies in the ship building industry. The architecture provides a set of interconnected PDM systems, which utilise workflow as the means of controlling the lifecycle of the data within an organisation, and how it interconnects with other organisations in the industry.

The first deliverable if the First Architecture Prototype (FAP) which will be tested and evaluated against the main business cases. The main software components of this SEANET architecture are:

- PDM Server, a STEP-based tool able to support data vaulting, access permissions management, multiple navigation methods, data versioning and workflow support.
- Conversion Server, a software tool able to translate a source data model mapping a certain base schema into a target data model mapping a desired view schema.
- SEANET Web Server (SWS), a Web-based tool able to centralise the management of world-wide spread web-services (concerning data sharing and exchange) and to sequence them to proper compound services.

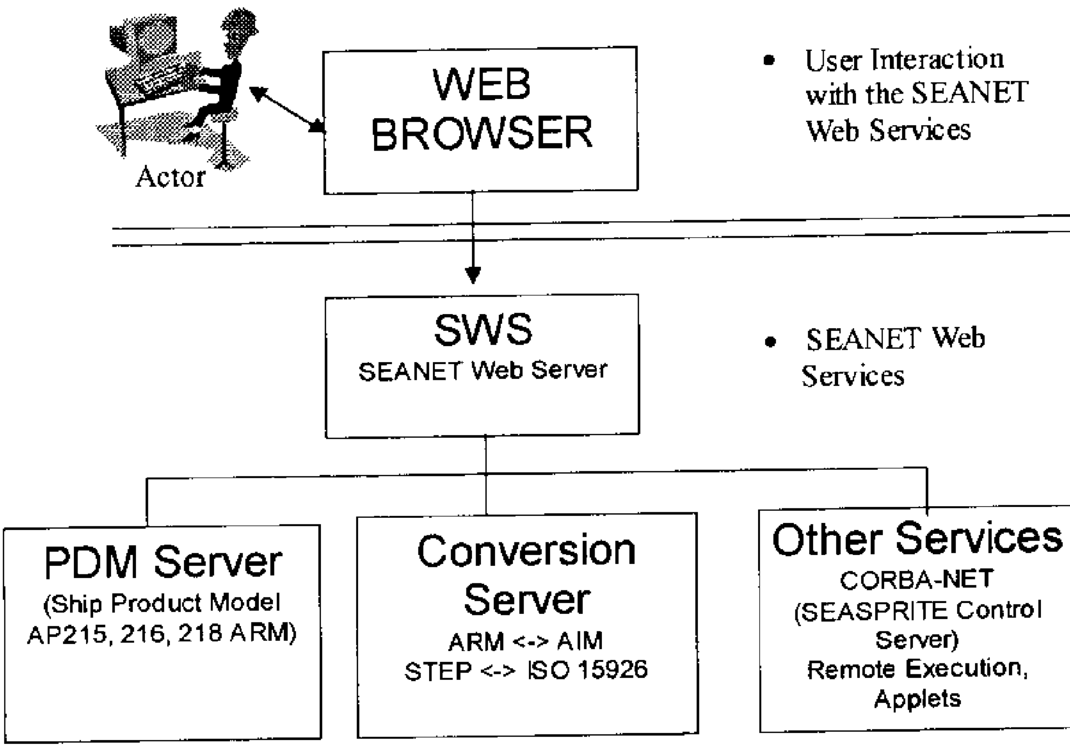


Figure 10. SEANET Architecture Modules

Such tools, each supporting only one step of the business processes identified in SEANET, need to be interlinked and interfaced, in order to be used in real-life applications.

The role to act as an interconnecting channel is played in the SEANET architecture by the SWS, which will be in charge of applications proper activation and data transfer via the INTERNET highway. Moreover, the FAP architecture aims at loosely interfacing the CORBA-based SEASPRITE DEM architecture.

PDM Server

The PDM capability is based on Quillion Systems PETS and offers a wide choice of development tools and the ability to access existing data, applications and services with a minimal amount of development work.

The PDM system will give the members of a product development team the opportunity to work with an integrated product and project data model, containing all product and project documentation independent of which system it was created by. The ability to navigate through the data by many different routes ensures that the correct information can be easily and quickly found. It also enables a faster and earlier release of information. These features will contribute to a faster product development cycle and thereby a reduced time-to-market.

A better quality of documentation is achieved when all parties work with one valid version of the product data. A document must be checked out by a user and is then automatically locked for other users. In this way the work with the correct revision/version of a document is ensured. A parallel development of a document is also prevented. An improvement in document quality should lead indirectly to improved product quality, even though the PDM systems themselves do not protect from design errors.

The PDM system can be used for an electronic distribution of documents or data. The advantage of this technique is that information is distributed to the proper people in an efficient way. The normal delay of the internal postal service is omitted, and documents are made available for instances down stream as soon as the document has been released.

Workflow support capability will complement the PDM system by ensuring that work is distributed in a timely and efficient manner. The workflow support tools will allow routing of tasks based on roles, sequencing of chains of tasks and monitoring and controlling the ongoing/executing business process with respect to status for the whole process or its individual subtasks.

Conversion Server

An innovative feature of the system will be to create systems capable of generating dynamic views of the STEP data stored in the architecture using EXPRESS-X as the mapping language between different EXPRESS schemas. This will provide the users with an interface from the ARM level SEANET environment to external schema based models as well as interoperability with official STEP shipbuilding standards on AIM level.

In analysing the different user views and the needs for mappings the following levels of view are identified:

1. **Application** views. These views represent the information requirements of the end-users within the scope of the SEANET information framework. It should be noted that the models are on Application Reference Model (ARM) level and are presented according to the Building Block methodology used by the Shipbuilding information standardisation community. The Ship Product Model (SPM) used in the SEANET context corresponds to the superset of relevant existing APs on ARM level, i.e. AP218/ARM, plus AP215/216/ARM as required.

2. **Interoperability** views for ensuring conformance and compatibility with other international standards and information models from closely related application domains. Candidate interoperability views include:

   - AP203, which is already an International Standard and therefore has considerable commercial implementation support.

- The Oil and Gas centred Standard ISO15926. This sister standard to STEP is being sponsored by the POSC/CAESAR consortium, which includes companies common to SEASPRITE and SEANET.

- AP215, 216 and 218 AIMs. The current implementations are based on the EMSA Protocols. When the shipbuilding APs become Standard, it will be necessary to conform to the standard by exchanging AIM based data. SEANET will contribute to an AIM based capability and will also help to validate the AIM based models in advance.

The information views of interest are illustrated by Fig. 1. (Note: the diagram does not present any system/implementation architecture, it is an illustration of the information views).

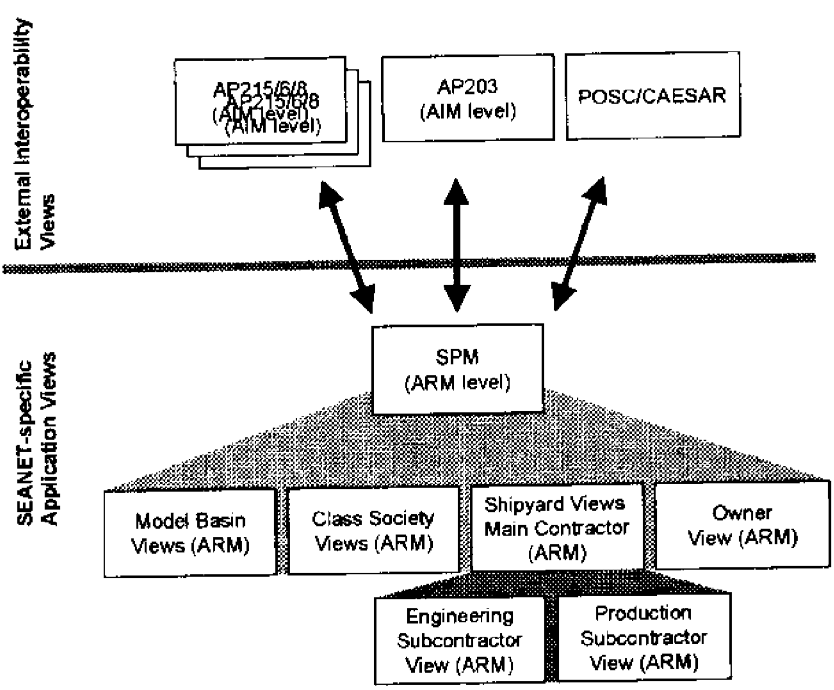


Figure 11. Candidate Application and Interoperability Views

The application views specified above will all be covered within the project by the current set of SEASPRITE translators; there is therefore no need to duplicate this functionality by separate mapping methods. Although this may be relevant for future implementations where there is no STEP translation capability.

The project is concentrating its efforts on the SPM ARM/AIM interoperability views at this time. This will give the partners an AIM based exchange capability and will serve to validate the AIM models as they are developed.

SEANET Web Server

Another innovative aspect of the SEANET system is the possibility given to an authorised thin client (browser) everywhere located in the net to register, book and execute web-services provided by the maritime society companies. This is possible thanks to the SEANET Web Server (SWS), a centralised manager able to support both elementary, atomic services (like PDM data navigation, selection, importing and exporting, STEP-based conversions, applets down-loading, file transfer) and sequences of them, forming the web-support to the SEANET business cases identified.

In particular, the SWS tool in the SEANET prototype shall provide the clients of the architecture with extensions to create a clear graphical user interface for the PDM and workflow components. The client will be kept minimal and ported to all the platforms used by the partners.

In the final SEANET system, the SWS shall at last represent an integration tool, able to harmonise different web-services under a common three tier architecture (client-SWS-dataserver) and to support the way for interconnecting workflows running at different sites on different PDM systems.

The feasibility of using XML, as an alternative method of data exchange, will also be investigated

Security

PETS provides a simple set of mechanisms to provide complete Access Control of all the information stored in the system. Access to the data is determined by the ownership of the data, combined with the group permissions of the logged on user. The permissions that can be set are: full access, read only, no access.

Using this structure it is easy to ensure that even if users from other organisations are allowed to access the PDM system, they can be denied access to most of the information unless it has been explicitly made available. This would be achieved by changing the ownership of the data, that is to be made available, to a group to which external users have read permission. Because the access permissions are defined on Groups (to which Users belong) then the addition of new Groups and Users is a straightforward task as they immediately inherit the access permissions of the Groups to which they are added. The Client-Server protocol transfers its data using128 bit encryption so that data on the network is always secure.

It is intended that the interface of final architecture with the web will configure firewalls and incorporate encryption which will allow partners to feel safe that the machines themselves cannot be compromised.

The exchange of sensitive data over the Internet requires that a high level of security be provided, both for user authentication/identification and for the data flow itself.

Available technologies supporting secure data exchange are:

1. At the network level, VPN (Virtual Private Network) support (by means of encryption routers or using ad-hoc software packages) can be implemented for server to server communication.
2. At the transport level, HTTPS can be used in client server or server to server communication, eventually adopting user identification schemas based on **client certificates** to ensure safe client authentication.
3. At the application level cryptography combined with data compression can be used in server to server or client to server communication.

All the above issues have been only sketched in designing the SEANET Web Server (SWS) Prototype and will be more deeply investigated for final SWS System.

Conclusion

The SEASPRITE project has advanced the development of the shipbuilding STEP APs 216 and 218 to the ISO Committee Draft stage. On top of this it has produced translators targeted at specific business cases so that the STEP technology can be exploited in the short term. In order that the data models remain stable the business case specifications are being published by EMSA as industry standard EMSA Protocols.

The STEP Ship Product Model, has been built to support data sharing as well as data exchange so SEANET is building on the work of SEANET by exploring the area of distributed STEP based PDM.. The tools developed will provide data sharing based on STEP as well as views, based on the emerging EXPRESS-X standard, to other complementary standards and formats.

These projects together are providing data exchange solutions for today's working and providing validation for the concepts and technologies required for future distributed electronic working in ship design and production.

Acknowledgements

Thanks are due to all the participants of the SEANET and SEASPRITE Projects. These are:

SEASPRITE: British Maritime Technology Ltd (BMT), Det Norske Veritas (DNV), Instituto Superior Tecnico (IST), Kvaerner as (KAS), Kockums Computer Systems (KCS), MARIN, Napa, Odense Steel Shipyard (OSS), SINTEF, Marconi Marine (Vickers) (VSEL).

SEANET: British Maritime Technology Ltd (BMT), Kockums Computer Systems (KCS), TXT, EuroSTEPSys Oy (ES), Odense Steel Shipyard (OSS), Quillion systems (QS).

Particular thanks are due to the contributors to the project deliverables, portions of which have been reproduced here.

References

1. Turner, Tim J. and :Langbecker, Uwe (1999), "The Shipbuilding Application Protocols and the role of the European Marine STEP Association", ICCAS 99
2. Sergio Gusmeroli, D310: "First Architecture Prototype", SEANET Deliverable, 5 March 1999
3. Kendall, J. and Hasund, K., Product Data Integration and Exchange for Ship Design and Operation, ICCAS 97
4. Healey, A, SEANET and Virtual Enterprise Support Architectures in the Maritime Industry, ICCAS 97

AN IMPLEMENTATION OF STEP TRANSLATORS FOR SHIPBUILDING PROTOCOLS BASED ON AN OBJECT-ORIENTED PRODUCT DATA MODEL APPROACH

Joyce Howell and Cathy Sabatini, Intergraph Corporation, Huntsville, AL

Robin Penley, Pete Lazo, and Rob Lisle, Newport News Shipbuilding, Newport News, VA

Abstract

Intergraph Corporation and Newport News Shipbuilding (NNS) have created a prototypical set of data translators for shipbuilding applications as part of the U.S. DARPA MARITECH sponsored MariSTEP project. Now completing its third and final year, MariSTEP is composed of a consortium of U.S. shipbuilders and CAD (Computer Aided Design) software vendors. The primary objective of the project is to create a prototype of a neutral data exchange mechanism using ISO 10303 - the Standard for the Exchange of Product Model Data (STEP). This mechanism, realized in the form of a translator, provides a capability to transfer data between product models at the U.S. shipyards. STEP translator implementations are based on an "object-flavored" information modeling language called EXPRESS, which became an ISO standard in 1994, ISO 10303-11:1994. STEP, through EXPRESS, enables some of the positive aspects of object-oriented design, namely the use of inheritance and modularization.

This paper, based on the authors' experiences within the MariSTEP project, describes how an object-oriented approach serves as a framework for addressing the implementation requirements of STEP translators for transferring product model data.

Introduction

Industry constantly seeks ways to improve productivity - to produce a better product at a lower cost within a reasonable time. The ability to exchange product model data provides a means to that end. It supports the business case scenarios for exchanges between the design agent, supplier, shipyard, classification society, and model basin. The exchanged data includes requirements specifications, preliminary and detailed design information, approvals, and manufacturing data, as well as the "as-built" information required for maintenance. Data exchange allows collaborative design by which a user can have access to the most current product model information at any given point in time. Note that the definition of data exchange is twofold: first, it must define a data model for the exchange, and second, it must define an interface mechanism between the requestor and the supplier of that data.

STEP has emerged as a leading mechanism of data exchange between CAD systems. Developed by the International Organization for Standardization (ISO), STEP provides industry-specific data models called Application Protocols (APs) which define schemas using the EXPRESS language (Part 11). Also defined within STEP are interface mechanisms for accessing that data via a text file or by Standard Data Access Interface (SDAI) using C++, C, FORTRAN, IDL, and JAVA.

Therefore, STEP meets the definition for data exchange; the data model is defined by the APs, and the access is defined by standard text file manipulation and SDAI.

The ability of STEP to exchange product model data that covers the entire product life cycle, including both geometric and non-geometric data, has given STEP an edge over its predecessor, the Initial Graphics Exchange Specification (IGES) format. IGES has been an accepted standard for exchanging geometric data for many years. STEP now supports not only the geometry constructs within IGES, but also has added the ability to exchange solids. Most importantly, with STEP, product model data is represented. In IGES, an object is identified by its geometry, i.e., a line or cylinder; in STEP, it would be defined in its application context. For example, within a STEP Ship Piping (AP217, see Reference 2) file, the object might be a 2-inch diameter copper nickel pipe with a pressure rating of 150 psi that is part of the fire control system. All this information (and more) would be captured within a STEP file.

EXPRESS, the data modeling language of STEP, readily supports many object-oriented concepts, making it well suited to an object-oriented CAD design.

Computer Related Terms

It is important to define some concepts here that will be referenced later in this paper.

A *business object* is an entity that exists within the domain of a specific business environment. Some examples of shipbuilding-related business objects are ship, pipe, pump, antenna, hull, and deck.

Product model data is a collection of graphical and non-graphical information that describes a business object over its entire life cycle.

An *object-oriented implementation* is an implementation that utilizes business objects to describe the product model data information.

Object-oriented language is a computer-interpretable language, such as C++, that supports the concepts of modularity, inheritance, data encapsulation, and defined behaviors.

A *file-based* CAD system stores its data and relationships in a sequential file format. Due to file size limitations and ease of access, the data is typically stored and configuration managed as multiple files. A *database-based* CAD system stores its data as information within a single database.

A *geometry-based* CAD system is dependent on the existence of a geometric representation of the business object in order to store the business object in the CAD system. This business object may or may not be stored as a separate entity, depending upon the capabilities of the CAD system, but its properties are commonly defined as attributes assigned to the geometric entity. An *object-based* CAD system can store a business object as a separate entity before any geometry is created. The business object's geometric and non-geometric properties can subsequently be assigned to the business object, as the information becomes available. The crucial difference here between the two types of systems is that the geometry is not essential to the creation of the business object in the object-based CAD system. Figure 1 illustrates the conceptual difference between geometry-based and object-based CAD systems data.

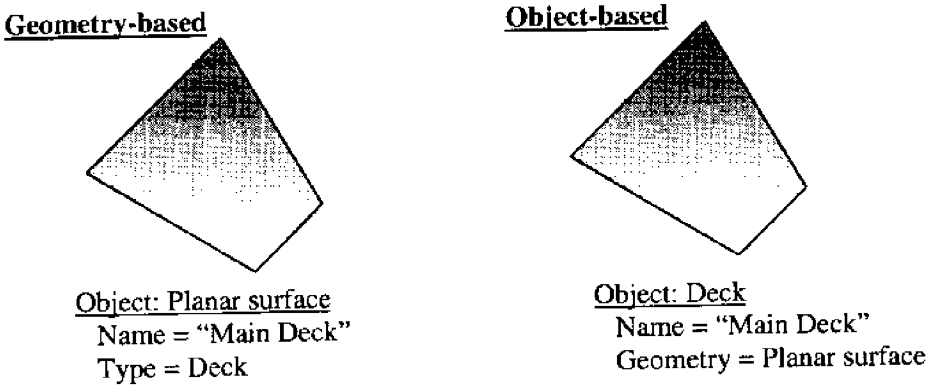


Figure 1 – Geometry-based Versus Object-based

Explicit geometry is geometry that is explicitly defined by geometrical elements such as curves and surfaces. *Associative geometry* is geometry that is defined topologically relative to other existing geometry. For example, in Figure 2 below, the Oily Bilge Tank compartment volume is defined by its topological boundaries: the surfaces of the main deck, the sides of the hull form, and two bulkheads. The original tank geometry is shown as solid lines. By modifying the z value for the deck, and the x value of the forward bulkhead, the volume was automatically recomputed as shown by dashed lines. *Derived geometry* is geometry that is calculated by the system based on the relationship between the objects themselves.

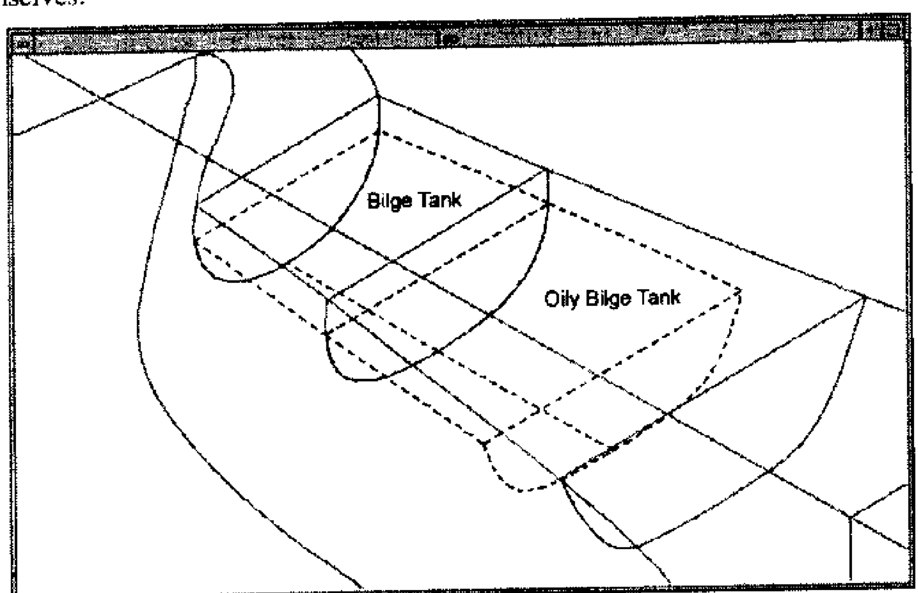


Figure 2 – Associative Geometry Example

A *legacy system* is any existing "last generation" system. These are typically file-based, geometry-based CAD systems. The business object data is typically stored as text attributes assigned to the geometric entity. The relationships between the objects are typically stored in a relational database.

Implementing STEP in a Shipbuilding Environment

Prototype STEP Shipbuilding Translator Implementation Projects

Members of the SEASPRITE program (a European maritime STEP development effort) and the U.S. NSRP "Convert NIDDESC APs to ISO APs" project have defined an initial set of shipbuilding APs to support international product model exchange requirements. These APs are developed and maintained by consensus with the ISO TC 184/SC4-T23-shipbuilding committee. There have been several pilot programs that have implemented prototype STEP translators for the shipbuilding environment. Under the European MARITIME and SEASPRITE STEP implementation projects, a modular approach to defining data models for shipbuilding applications was devised. This modular data modeling approach involves the creation of mini-schemas, each having a few EXPRESS entities, which define a specific concept or unit of functionality. These mini-schemas are called "Building Blocks." The shipbuilding APs are built from a collection of Building Blocks, some specific to the AP and some shared by multiple APs. The building block approach supports the object-oriented design requirement of modularity. The use of the ISO building blocks has been adopted by the MariSTEP project. MariSTEP, a U.S. DARPA-funded program, represents a consortium of U.S. shipbuilders and CAD software vendors. The MariSTEP team has successfully implemented STEP translators for the ARM (Application Reference Model) of four shipbuilding ISO standards. The standards consist of data definitions for ship arrangements (AP 215, see Reference 3), ship moulded forms (AP 216, see Reference 4), ship piping (AP 217, see Reference 2), and ship structure (AP 218, see Reference 5).

Implementor's Agreements and Improvement of the Shipbuilding APs Prior to Standardization

A primary goal of the first prototype implementation projects in both the U.S. and Europe has been to validate the content of the APs Application Reference Models which document the requirements for exchange of product model data for the particular ship design discipline. These early prototype implementations have been used to test the validity of the ARM requirements using realistic ship design data and to assess the ability of the various, dissimilar CAD systems to support the requirements that are progressing through the international standardization process. While the STEP standard is being developed as the next-generation capability for sharing of data, it was expected, and proved to be true, that in most cases the legacy CAD systems in use at the various shipyards could not completely fulfill the requirements for data exchange as documented in the Shipbuilding APs. In order to assure useful exchanges using the legacy infrastructure in spite of this limitation, both SEASPRITE and MariSTEP found it necessary to define and document implementor's agreements. For SEASPRITE, these agreements were documented in the Implementation Agreements Log. For MariSTEP, the agreements were documented in the System Requirements Document (SRD).

The MariSTEP SRD documented such agreements as which entities were required for exchange, attribute definition interpretations, required attributes that could be ignored, and attributes that were optional in the schema but were required for exchange. In many cases, the SRD documented informal agreements to allow successful exchanges between CAD systems with very different underlying architectures; in others, the SRD served to document proposed changes to the ISO schemas. In general, ISO changes were proposed if the need for the agreement fell into one of the following categories: 1) the APs ARM was found insufficient to support the requirements for a production environment exchange; 2) modifications to the APs data model would facilitate future implementations; 3) text definitions for the AP entities and attributes were unclear or subject to multiple interpretations. Each change proposed to the ISO schemas was presented at meetings of the ISO STEP Shipbuilding Committee for resolution by those parties involved in development of the appropriate shipbuilding AP.

The continuing goal of the various implementation projects is to feed back all lessons learned into the standards development process to improve the quality of the standard while it is most cost-effective to do so, before final international standardization. A parallel goal is to improve the CAD systems in use at the U.S. shipyards to fully support the content of the STEP shipbuilding APs. In this way, eventually, Implementor's Agreements and modifications to the requirements as documented within the AP data models will no longer be needed.

A Sample MariSTEP Translator Implementation

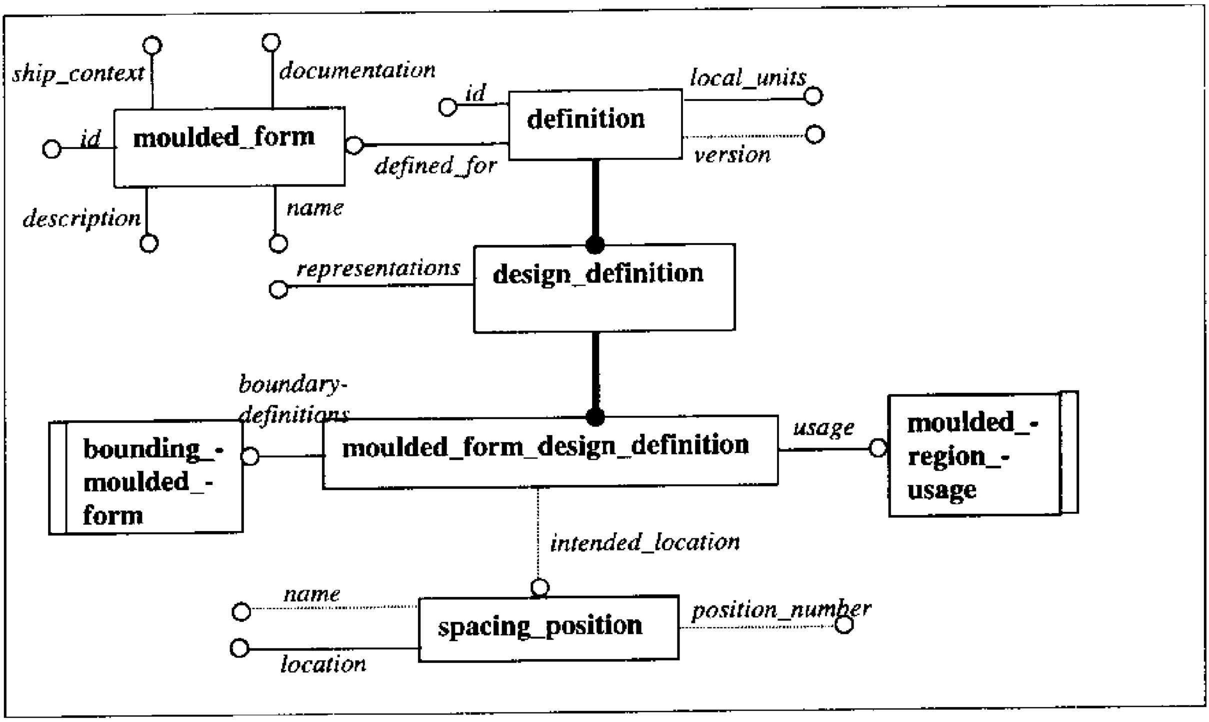
To illustrate a concrete example of the modularity and inheritance found in the STEP data models, a sample of the EXPRESS data definition for a ship moulded form is shown here. The ship moulded form application protocol; ISO 10303-216, (a.k.a. AP 216) specifies the data necessary to define the hull form and interior decks and bulkheads of a ship. The data definition of these moulded forms is captured in the ARM using the EXPRESS language. The MariSTEP project implemented translators for the ARM of AP 216, defining a subset of the data model for exchange.

In the MariSTEP implementation of ship moulded forms, a deck (business object) on a ship is modeled as a **moulded_form**. Attributes specific to the design lifecycle stage of the deck are captured within the **moulded_form_design_definition**. The EXPRESS entity definition shown below reflects the **moulded_form** entity as well as its related **moulded_form_design_definition.**

```
ENTITY moulded_form
 SUBTYPE OF (item);
END_ENTITY;

ENTITY moulded_form_design_definition
 SUBTYPE OF (design_definition);
  usage                 : moulded_region_usage;
  intended_location     : OPTIONAL spacing_position;
  boundary_definitions  : OPTIONAL LIST OF bounding_moulded_form;
END_ENTITY;

TYPE moulded_region_usage = ENUMERATION OF
    (blending_surface, bottom, deck, hull_appendage, ...);
END_TYPE;

TYPE bounding_moulded_form = SELECT (moulded_form_design_definition,
                                     external_moulded_form_design_definition);
END_TYPE;

ENTITY spacing_position
  SUPERTYPE OF (ONEOF (longitudinal_position, transverse_position, vertical_position));
    position_number  : OPTIONAL INTEGER;
    name             : OPTIONAL STRING;
    location         : length_measure;
END_ENTITY;
```

The **moulded_form_design_definition** is a specialized type of **design_definition** that in turn is a specialized type of **definition**. Through the attribute inheritance mechanism of EXPRESS, the **moulded_form_design_definition** also inherits the attributes of the **design_definition and definition** entities.

The STEP data model may also be represented graphically as shown below using EXPRESS-G notation. Each box represents an object with its attributes. These attributes can represent a relationship to another object. The thick lines represent a subtype relationship (specialization).



The **moulded_form**'s specific data includes properties of the deck, such as the *name*, *description*, associated *documentation*, and *ship_context*. The *documentation* attribute identifies specifications or related drawings that control the design of the deck. The *ship_context* attribute identifies the ship entity where the **moulded_form** is located; this could be a particular hull within a fleet of ships for which this deck is applicable.

The **moulded_form_design_definition** entity is linked to its related **moulded_form** entity through the *defined_for* attribute that it inherits because it is a subtype of the **definition** entity. The **moulded_form_design_definition** also inherits the *id*, *version*, and *local_units* attributes from **definition** and the *representations* attribute from **design_definition**. The *id* represents a *global_id*, which is a globally unique identifier. The *version* identifies the version of the instantiated object. The *id* combined with the *version* defines a unique, immutable object in the product model database. The *local_units* attribute allows the instantiated object to be defined in a local system of units, in addition to the global units defined for the ship.

The **design_definition**'s specific data is the *representations* of the instantiated object, or in this case, the deck. The *representations*, as implemented in the MariSTEP project, describe the explicit geometry and geometric topology of the deck, e.g., the faces, surfaces, edges, trim curves, vertices, and points that define its shape.

The **moulded_form_design_definition**'s specific data includes *usage, intended_location*, and *boundary_definitions*. The *usage* is an enumeration of possible functions that a **moulded_form_design_definition** might possess, such as *deck, transverse_bulkhead, hull_surface*, and *blending_surface*. The *intended_location* attribute specifies a relationship of the deck to a **spacing_position.** A **spacing_position** is a reference plane defined along one of the ship's major axes, X, Y, or Z. A ship is defined in a CAD system by a global X, Y, and Z axis. The ship's baseline is defined at zero along the Z axis. The deck's vertical height above baseline can be located by specifying an offset from a reference plane, or **spacing_position**, defined along the Z axis. The deck's shape is topologically defined by its *boundary_definitions*. For example, a flat deck, (one with no camber or shear) is initially defined as an infinite plane along the X and Y axes. But in reality, the deck's length and width are bounded by the hull form and interior bulkheads. The intersection of the bounding surfaces with the deck's surface define the explicit shape of the deck as seen in Figure 3 below.

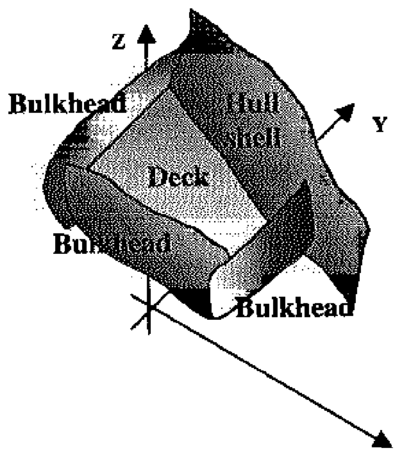


Figure 3 – Topological definition of a deck

MariSTEP Implementation for a Legacy CAD System

The MariSTEP translators were implemented in a variety of shipbuilding environments. Most of the implementations involved legacy CAD systems. So, not surprisingly, most of the reported problems with the implementations were related to importing inconsistent geometrical and topological data. Inconsistent, meaning that the geometrical and topological representation exported from the sending system differed from what the receiving system was expecting. The geometrical and topological data that was exported was all valid STEP data but inconsistent with the receiving system's modeling conventions. Also, geometric tolerance and data precision played an important role in converting the data. Without the geometry, the translators built on top of geometry-based CAD systems failed to transfer any of the relevant product model data. For the translation to be successful, importing the geometry rather than the relationships between the business objects was the most critical operation for these systems.

MariSTEP Implementation for an Object-based CAD System

Conversely, for the MariSTEP translators that were developed on top of object-based CAD systems, import of geometrical data did not pose a major obstacle. As long as the native CAD system could support the business object being imported, the main issue was where to store the imported object's attributes and relationships to other objects. To an object-based CAD system, geometry is just another attribute of the business object. If there were problems with importing the geometric data, it was not a major concern. As long as the relationships were established, the geometry could be derived by the receiving CAD system. For example, when importing AP 216 moulded form data in an object-based CAD system, the moulded form's shape was implicitly defined by its relationships to other moulded forms. If the relationships were not specified in the exchange, then the fallback was to import the explicit geometry defined in the exchange. This mechanism allows for relationships to be established at a later date so that the geometric representation can be regenerated based on the related objects.

A MariSTEP Approach to Support Both Object-based And Geometry-based CAD Systems

One of the MariSTEP schemas, AP 218 ship structures, attempted to bridge the gap between the geometry-based and object-based core issues by establishing a mechanism to support explicit and implicit geometrical data. In the EXPRESS fragment below, taken from the AP 218 schema, a SELECT type was used to allow for specification of plate part boundaries as either explicit geometric curves or as relationships between objects, such as the plate's relationship with a seam or the boundary of the system to which the plate belongs.

```
ENTITY plate_design_definition
  SUBTYPE OF (structural_part_design_definition);
    SELF\Definition.defined_for  : SET [1:?] OF plate;
    thickness                    : positive_length_measure;
    material_offset              : REAL;
    border                       : LIST [1:?] OF plate_boundary;
    moulded_surface              : OPTIONAL Any_surface;
END_ENTITY;

TYPE plate_boundary = SELECT(panel_system_boundary,
                             plate_boundary_relationship,
                             bounded_curve);
END_TYPE;
```

This approach proves problematic to both the geometry-based and the object-based CAD system. The legacy CAD system can easily import and export the explicit geometry, e.g., the plate's border if it is a **bounded_curve**. But, for the case of the plate's border being represented by relationships, the legacy CAD system must somehow extract or derive the geometry from the relationship since it is the geometry that is stored in the database. The relationship data may or may not be maintained by defining text attributes on the geometry, referred to as "attributed geometry."

With more advanced geometry-based CAD systems, the relationship data may play a role in deriving a new, revised geometric representation of the object. This feature is sometimes referred to as "associative geometry." In Figure 4 below, the curve geometry underlying the seam is also one of the curves defining the border of plates 1 and 2. The application knows that if the underlying curve changes, the border of the plates change also.

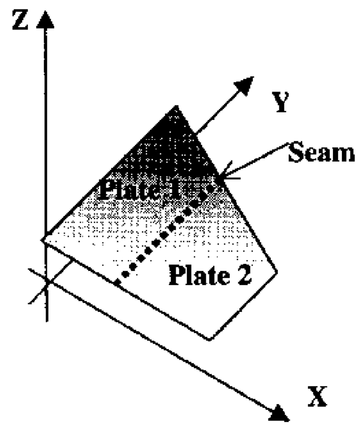


Figure 4 – Deck Plate with Seam

For an object-based system, the relationship information is the key. If the seam that represents the border of the plate is relocated, the relationship between the plate and seam is still maintained in the database. The application retains the relocated seam and can regenerate the plate geometry on an "as needed" basis. However, the explicit geometry poses a dilemma for the object-based system. On export, the object-based system can easily derive the geometry from the object relationship data. The derived geometry can then be extracted into the data exchange format. But on import, the receiving system must now store explicit geometric data that may conflict with the system generated geometry obtained from the relationships stored on the object. For example, the receiving system imports a plate with a seam border and derives the plate's geometry. If the derived geometry differs from the explicit imported geometry, what should the receiving system do? Since an object-based CAD system does not store the geometry because it can be derived, the explicit geometry is discarded and the relationship data, which can be maintained, is persisted. However, if the relationship data is missing, or invalid, the object-based system must somehow persist the geometry since that is the only graphical representation of the object. This approach is fundamentally inconsistent with object-based architecture.

Problems with STEP

Even though the STEP data model is specified in EXPRESS, a computer-interpretable language that can be used to generate translators, the data definition alone is insufficient for the development of a successful translator. There is a significant amount of engineering knowledge embedded within the text of the STEP AP documents which must also be implemented. Collaboration is required between both software development staff and domain experts who can interpret the intention of the data model. The MariSTEP consortium was an example of such a collaboration of shipbuilders who are familiar with the domain of shipbuilding data models and CAD vendors who are relying on the shipbuilding experts to relay the semantics of the shipbuilding data model.

Due to shortcomings of the EXPRESS language, it is possible to circumvent the intention of the data model. For example, it is allowable in EXPRESS to exchange an empty string for a *name* attribute's value, even though *name* is a required attribute in the data model. STEP also allows the specification of optional attributes, meaning that the receiving system cannot depend on those attributes being populated.

Within the MariSTEP project, it was decided that optional attribute did not require storage by the receiving system. However, this issue was raised to the ISO committee because many felt that the receiving system should not throw away data that may prove valuable to another system. Storing of this data may require CAD system enhancements. And it is difficult to say generically that a specific optional attribute, for example *description*, is irrelevant in all cases; it may have a semantic meaning -- considered optional for one moulded_form entity but relevant for another moulded_form entity. Also, many STEP data models used optional attributes as a mechanism to store life cycle phase dependent data. For instance, in the functional phase of ship design, a compartment on a ship may be specified to occupy a certain volume, but the decks and bulkheads that form its boundaries may not be defined until a later design phase. For this reason, the boundary data for the compartment is made optional in the STEP EXPRESS definition of the compartment. If an exchange of data occurs during the functional design phase, the compartment will have no boundaries, but at the later design phase, the boundary attribute will be populated with data.

Future of STEP

The future of STEP as a universally accepted neutral data exchange medium will be determined in time. But regardless of that result, the benefits of STEP must be recognized. STEP does not solve all the problems but has contributed to major progress in data exchange for the U.S. shipbuilding industry, even considering the MariSTEP early prototype implementation issues. Most importantly, STEP provides a mechanism for defining and reaching international consensus on application-specific data models. These data models may be implemented using various other technologies working with STEP data. With the advent of the Standard Data Access Interface (SDAI), access to these application specific data models will enable consistent data definitions in heterogeneous systems and provide the basis for a shared data environment, both internally and externally to the enterprise. A modification to the STEP architecture that is currently being refined is applying a modular approach to defining application-specific data in smaller modules. This process will provide a simpler and faster implementation of units of functionality within a specific application domain, rather than implementing an entire application protocol. This modular approach to STEP will hopefully provide a more readily accepted standard for the definition of product model data and will make its implementation faster and cheaper.

Conclusion

STEP through EXPRESS, enables some of the positive aspects of object-oriented design, namely inheritance and modularity. Though legacy CAD systems can make use of STEP data models, they are still geometry-based and, being file-based, access to the relationship data proves cumbersome. The union of STEP and an object-based system provide the best solution to utilizing the full functionality of the STEP product data definitions. Currently, Intergraph Corporation and Newport News Shipbuilding are jointly developing a next generation CAD system under the COMPASS program. New hardware and software are being customized for the shipbuilding design and manufacturing environment. The software is object-based and will support the STEP shipbuilding application protocols. Together NNS and Intergraph will continue to pursue the development of STEP as an enabling collaborative technology for the shipbuilding enterprise.

References

1. ISO 10303-11:1994, Industrial automation systems and integration - Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference manual.
2. ISO 10303-217, Industrial automation systems and integration - Product data representation and exchange - Part 217: Application protocol: Ship piping.
3. ISO 10303-215, Industrial automation systems and integration - Product data representation and exchange - Part 215: Application protocol: Ship arrangement.
4. ISO 10303-216, Industrial automation systems and integration - Product data representation and exchange - Part 216: Application protocol: Ship moulded forms.
5. ISO 10303-218, Industrial automation systems and integration - Product data representation and exchange - Part 218: Application protocol: Ship structure.

THE SHIPBUILDING APPLICATION PROTOCOLS AND THE ROLE OF THE EUROPEAN MARINE STEP ASSOCIATION (EMSA)

Tim Turner and John Kendall, Lloyd's Register of Shipping
100 Leadenhall Street, London, EC3A 3BP, UK

Introduction

The emerging ISO 10303 standard[1] (Standards for the Exchange of Product Model Data - commonly known as STEP), represents a significant opportunity for the maritime industry to improve its overall efficiency and world-wide competitiveness by facilitating engineering data exchange, flexible computer integrated design and manufacturing, co-operative engineering and effective life-cycle data management.

However, rather than developing a single, Ship Product Model in a single data model, the ISO Ship Team (ISO TC184/SC4/WG3/T23), have chosen to develop a suite of Application Protocols [11][12][13][14] which together make up the Ship Product Model. From these, a core, generic and extensible Ship Common Model[10] can be used to describe (and predict) the behaviour of the Ship APs. This approach has it's own advantages and disadvantages which are reflected in the paper presented.

At the European level, the maritime industry is already making a significant contribution to the development of the ISO STEP standard and the Marine Product Model by means of a number of national, European and industry funded projects. These projects, however, have a finite lifespan and may well terminate before their valuable input can be formally incorporated into the ISO standard or can be fully released by industry. For this purpose an association of organisations from the maritime sector has been formed which is known as the European Marine STEP Association (EMSA).

EMSA uses a methodology (complimentary to that of the ISO mechanisms) called the Building Block (BB) approach[1] to modularise the development of Application Protocols, which has been adopted by the Shipbuilding Team within the ISO committee and is being promoted in other groups within STEP. These are held in a central repository for which a number of tools have been developed that help with their management and use.

EMSA promotes the use of & development of STEP in the Marine Industry through a number of avenues. One of these is through a set of Business Cases where bottlenecks in the industry focus attention on areas that could gain significant benefit from the use of STEP technology. These form useful conformance classes (or subsets) of the Ship Product Model which can be developed from the same building blocks as those being used for the ISO models. This paper will describe this initiative and the need for industry to start developing translators for even these small business cases.

[1] Initially developed under the NEUTRABAS[16] & later the MARITIME[17] projects.

ISO Shipbuilding AP Requirements

The design of the Ship Product Model (SPM) being developed within the ISO Ship Team currently distinguishes between a number of systems and disciplines as shown in Figure 1.

Due to the complexity, the scope of different systems and disciplines, and the potential amount of data likely to be exchanged with the SPM, these requirements are being developed independently as a suite of APs. Those currently identified with an AP number are being developed in the first phase of development.

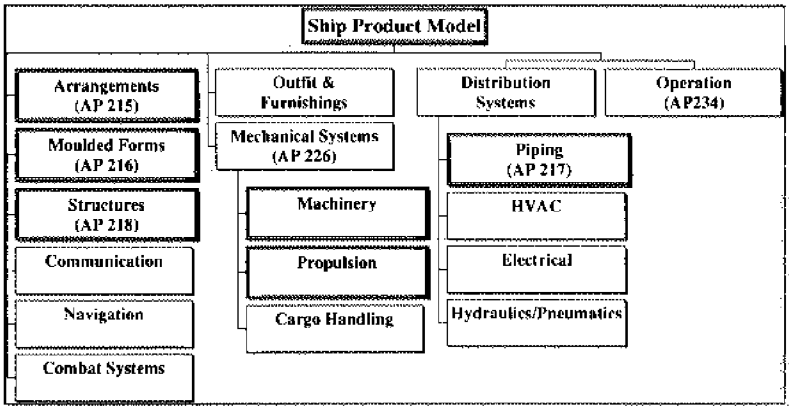


Figure 1: Current Requirements for the Ship Product Model

Simplified Application Protocol Contents

Simply put, the major items of an AP document consist of an Application Activity Model (AAM), the Application Reference Model (ARM) and lastly an Application Interpreted Model (AIM).

The AAM describes activities and information flows using diagrams and textual definitions. STEP provides a notation for this called IDEF0, a decomposable diagramming mechanism for the description of inputs, modifiers, controls and outputs for each activity. Ultimately, the inputs and outputs identified by the AAM are further defined within the ARM.

The ARM describes the information requirements (data) used in the domain delimited by the AAM, through a number of "clauses" (sections) in the document. The document structure (see Figure 2) is the same for all APs, with the most detailed definitions being provided in section 4 consisting of statements regarding Units of Functionality (UoF) in clause 4.1, application objects in clause 4.2, and any assertions of those objects in clause 4.3. STEP provides a modelling language called EXPRESS (ISO 10303-11) which describes the data in terms of entities and attributes to represent the domain. EXPRESS is a formal object-oriented language allowing sophisticated models to be built and represented either textually or graphically using EXPRESS-G.

Figure 2: ISO AP Document Structure

Type of element		*Element*
Preliminary		*Title page* *Contents, ...* *Foreword* *Introduction*
Normative	*General*	*Title* *1.Scope* *2.Normative Reference*
	Technical	*3.Definitions, Symbols and abbreviations* *4.InformationRequirements (ARM)* *5. Application Interpreted Model (AIM)*
Supplementary		*Informative Annexes (AAM)* *Footnotes* *Index*

The AIM, which describes the same data, but interpreted in terms of the Integrated Resources (IR's) - a generic, domain independent set of constructs, is the last form of the data and represents the final standardised version. The process of going from the ARM to the AIM is called interpretation. The

AIM is documented as section 5 of an AP. The AIM is also described using EXPRESS, but the terminology of the domain is replaced by that of the generic resources and a set of complex rules, which preserve the semantics of the ARM. However, the terminology of the ARM is maintained through a mapping table (clause 5.1) which shows how the requirements from the domain have been mapped to the generic constructs of the IR's (clause 5.2). The most important and difficult task is the selection of the particular IR construct as the target of the mapping. To do this requires an extremely thorough and broad knowledge of the IRs and how they have been used in the development of other AP AIMs. The purpose of this last step is to enable the use of a common set of constructs across industry (rather than developing single of un-interoperable standards).

Simplified ISO Process

The AP development process at ISO is constantly undergoing change in reflection to new technology and ideas. However, in recent years the *basic* ISO process has been as shown in Figure 3 below.

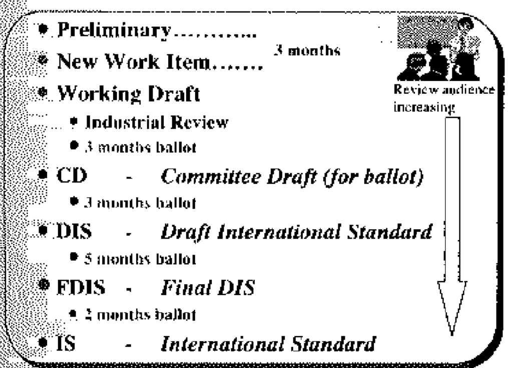
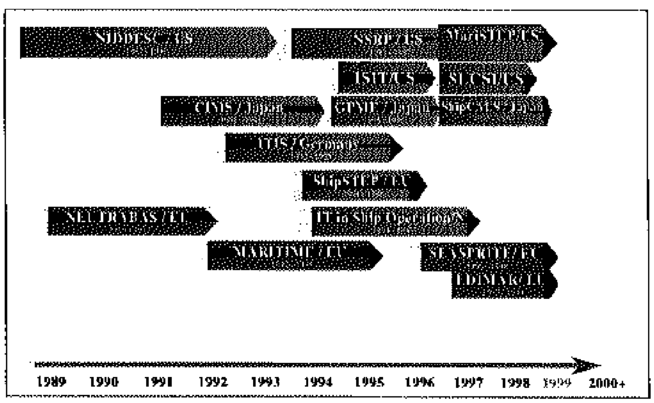


Figure 3: Typical ISO Stages/Process

As Figure 3 alone shows, AP development is an expensive, time consuming process. Typically, at each review stage (only major ones shown), the audience for review is widened, from the initial, internal committees to STEP, to the whole of industry in general. Of equal importance is the fact that the content of the AP increases up to the CD stage. However, even without carrying out any work, the process itself takes a minimum of nearly 1.5 years. Hence most single APs can take between 3 and 5 years to develop, depending upon industry review. For developing a set of inter-dependant suite of AP's, as in shipbuilding, it has taken somewhat longer. One reason for this has been the need to ensure interoperability and consistency between them at the ARM level before proceeding to the AIM level. It has also been evident that, unlike shipbuilding, other domains such as automotive or aerospace have been very customer driven providing much of the demand for such standards.

Figure 4: Supporting Projects over time



New personnel tend to arrive when supporting national projects are initiated, but then leave when they have been completed. This injection and then depletion of expertise over time has had a knock-on effect on the progress at the ISO level, sometimes resulting in old problems being revisited, sometimes with different conclusions and impact for the models being developed. Figure 4 shows some of the initiatives of the last ten years in shipbuilding, which have contributed to the ISO process.

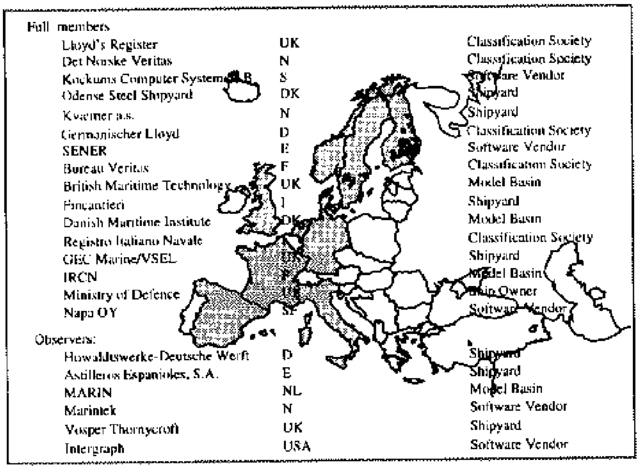
The European Marine STEP Association

The European Maritime STEP Association (EMSA) was formed partly in response to this problem and to co-ordinate the on going the European input into the development and implementation of the STEP standard. Additionally, EMSA was created to help industry to maximise the return on the investment so far and to ensure that the maximum benefits are obtained in future. For this purpose an association of organisations from the maritime sector has been formed which is known as the European Marine STEP Association (EMSA).

The Association

In 1998 EMSA represented some 22 companies in the Marine industry, including 5 classification societies, 6 shipyards, 3 software vendors, 4 model basins, 1 ship owner & 1 research institute.

Figure 5: Members and Countries



Through it's co-ordinating relationship to STEP on one hand, and as a forum for co-ordinating European STEP based projects, EMSA allows it's members to collaborate with others and develop early solutions to the business needs of the market and provide feedback through EMSA to it's members and the STEP community.

EMSA promotes the use of and development of STEP in the Marine Industry, through the co-chairing of the ISO Technical Committee on Shipbuilding (ISO TC184/SC4/WG3/T23), the establishment of relationships with similar organisations (e.g. ProSTEP - automotive STEP initiative) and a number of (independently funded) research projects in Industry, training and EMSA's own business cases (to be described).

The EMSA Approach

EMSA uses a methodology (complimentary to that of the ISO mechanisms) called the Building Block (BB) approach to modularise the development of Application Reference Models (ARMs), which has been adopted by the Shipbuilding Group within the ISO committee and is being promoted in other groups within STEP. Each Building Block is made up of three schemas; an import schema (to pull in entities from other Building Blocks), a model schema (using those entities imported and defining new ones), and lastly, an export schema (to describe those parts of the model visible to other Building Blocks). Each BB, EXPRESS entity, type, where rule, and function etc., are documented as part of the development of the BB.

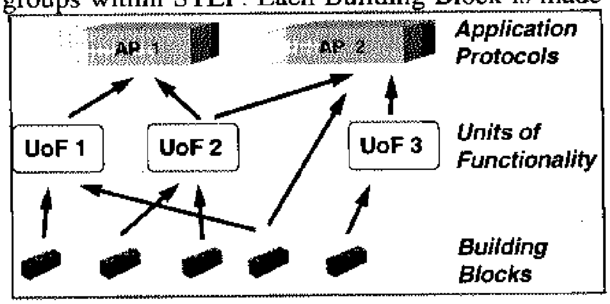


Figure 6: EMSA Building Block Approach

This serves both to inform other modellers about the scope, purpose and use of the BB, and can be used to document the application objects within the ARM when creating the AP.
Rules and guidance notes in the form of a "Cookbook" have been developed for the usage of EXPRESS within an AP to enable consistency of both the model and modelling style which provide examples and strategies for alternative mechanisms for the representation of the data requirements.
As the AP ARMs are built upon a number of Units of Functionality (UoF's), and that each UoF may reference one or more Building Blocks, the APs can be built up from a collection of Building Blocks provided within a "shortform". This shortform schema enables the AP to be built up from Building Blocks into what is known as the longform schema. All references are resolved in this final schema during compilation. All Building Blocks are fully documented in terms of providing textual definitions of the entities, attributes, types and functions etc., which means that much of the effort to document an AP can be automated via a number of scripts. This modular approach enables the different shipbuilding APs to be developed from a single source of information which reduces redundancy and maintenance of the model and helps to make the documentation process more manageable.

EMSA also manages and controls access to the BB server, an electronic email service where all Building Blocks are kept under revision control, automatically checking the EXPRESS syntax, posting updates received from BB owners/editors, or issues on the model to members, the AP developers and the EMSA web site[2]. The same is true of the shortforms and these can be used to retrieve Building Blocks according to a particular shortform that makes up a specific AP ARM. Latest developments also allow the automatic generation of longforms, subsequent resolution of references, checking of semantics and the posting of such updates to members, AP developers and the EMSA web site.

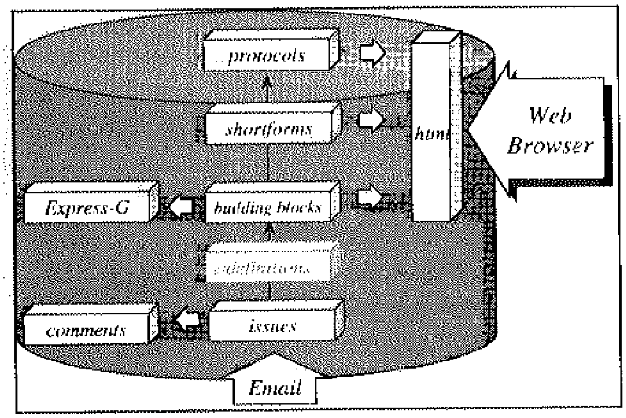


Figure 7: EMSA Server

The Ship Common Model

In order to develop a suite of APs it was soon realised that in order to maintain consistency and interoperability across the APs, the team would need to define a mechanism to help identify those areas of overlap between the different models. This resulted in the development of a "Ship Common Model" (SCM), which has served to provide a backbone for the development of the AP ARMs. In order to validate the semantics of the models being developed through the APs, it was decided to also implement a series of prototype ARM-based translators. This would also provide further opportunities for the verification of the approach, the SCM and to release early examples of the potential of STEP to the shipbuilding industry

The SCM identifies four types of information; the SCM Framework, the Product Structure, a set of Utilities and lastly a number of Support Resources.

---

[2] http://www.emsa-bbs.org/

- Framework

The SCM Framework acts as a generic, core model used by all of the shipbuilding APs. It provides the basic (abstract) entities and structural relationships between items, definitions and representations. Here we allow an item to have several definitions to provide for lifecycle differences in the model. For example, an item (such as a compartment) might have a definition such as a compartment design definition, a functional definition, a production definition, or assembly definition etc., for use during the design & build phases, and perhaps an operational definition, a maintenance definition etc., for use during other lifecycle phases.

Each definition may then have a form of representation, which describes the shape, or geometry of the item defined. Items may then be collected together in item structures via an item relationship restricting the attributes to consist of either another item or other item structures (sub-structures). Item relationships allow items to be associated together and because the product structure views are both item structures and items, these different views can be related.

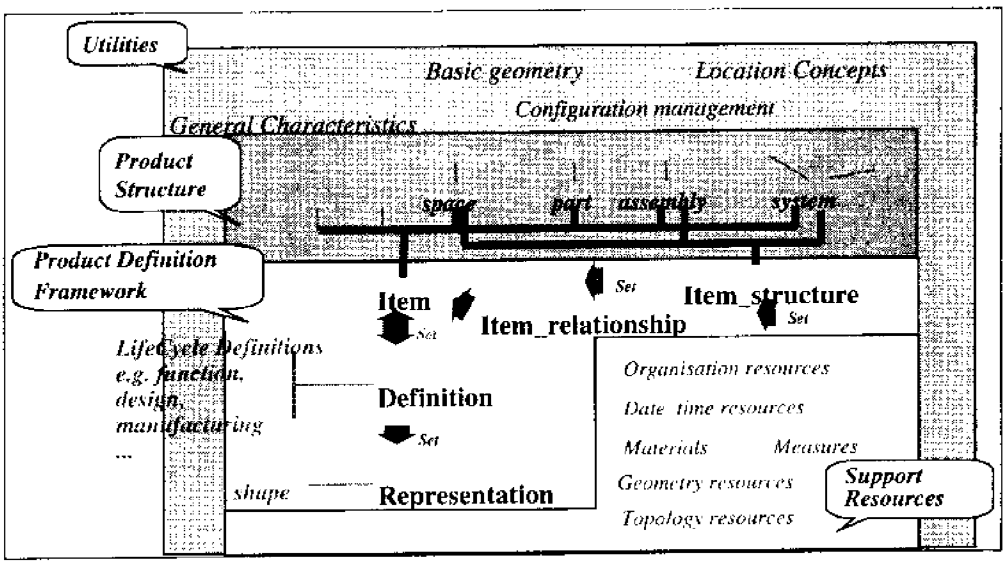


Figure 8: The Ship Common Model

- Product Structures

The SCM Product Structures are built upon the Framework and provide a set of generic views that represent a structuring of the product by (for example) system, space, and assembly or by connectivity. Any one or more of these can be adapted by any protocol for it's own use. New protocols should use or update those generic models rather than providing an isolated structuring technique. In this sense the Product Structure mechanisms are actually domain independent until specialised by the AP under consideration. However, where a view is missing or not catered for, the model should be updated to take account of the requirements in a generic manner similar to those already established.

- Utilities

The SCM Utilities provide a set of support services that most APs might need. As the name suggests they are not mandatory for use in each AP as this depends more on the nature of each AP. They are characteristically, unlike the previous parts, not sub-typed or specialised for use in other APs. In general they are used "as is". Therefore, those APs wishing to use employ configuration management systems or other utilities such as location concepts, general characteristics etc., must use these parts of the SCM.

- Support Resources

The Support Resources underpin the Framework and the Utilities. They consist of the STEP Integrated Resources also known as the Part40's, and the Application Interpreted Constructs (AICs). These provide the basic definitions of the constructs used in all STEP models.

Of course the SCM can be treated just like any other AP and the Building Blocks for this can be configured accordingly. Figure 9 shows a planning model of how these Building Blocks can be configured for this purpose. Where more than one Building Block is referenced, a shadow effect has been drawn. The use of such a "meta model" has its advantages and disadvantages for both the ARM level and Application Interpreted Model (AIM) levels.

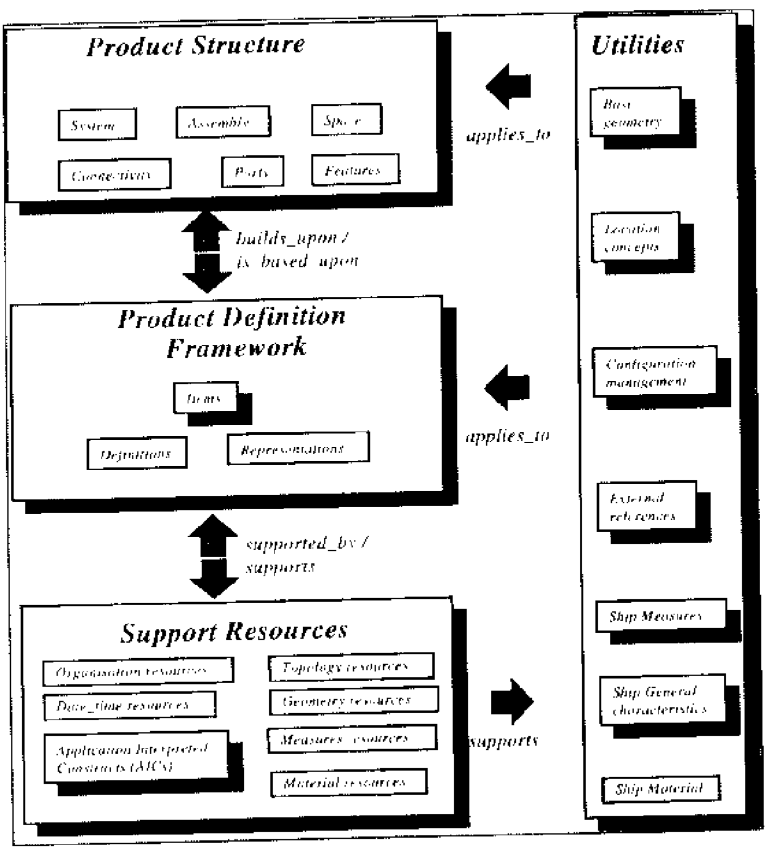


Figure 9: SCM Planning Model

SCM and ARM Level Use

The primary use for the SCM has been at the ARM level and in particular to help co-ordinate the different uses of the Building Blocks across the different APs. Apart from the main drivers discussed above, the SCM also helps to overcome different modelling styles and reduces the need to develop independent ways of organising each model. This might be especially true in cases where there is more than one way to model the domain. By providing a flexible backbone, with ready built viewpoints on the domain, the modeller is free to examine how best to make use of the constructs given, rather than develop their own mechanism to represent the view of the domain. This mechanism is something frequently only discovered towards the end of the modelling effort, when the strategy of the model is mature and a consistent approach to the model's organisation is required. Hence the modeller can instead, concentrate on the issues at hand and specific to the domain.

characteristically, unlike the previous parts, not sub-typed or specialised for use in other APs. In general they are used "as is". Therefore, those APs wishing to use employ configuration management systems or other utilities such as location concepts, general characteristics etc., must use these parts of the SCM.

- Support Resources

The Support Resources underpin the Framework and the Utilities. They consist of the STEP Integrated Resources also known as the Part40's, and the Application Interpreted Constructs (AICs). These provide the basic definitions of the constructs used in all STEP models.

Of course the SCM can be treated just like any other AP and the Building Blocks for this can be configured accordingly. Figure 9 shows a planning model of how these Building Blocks can be configured for this purpose. Where more than one Building Block is referenced, a shadow effect has been drawn. The use of such a "meta model" has its advantages and disadvantages for both the ARM level and Application Interpreted Model (AIM) levels.

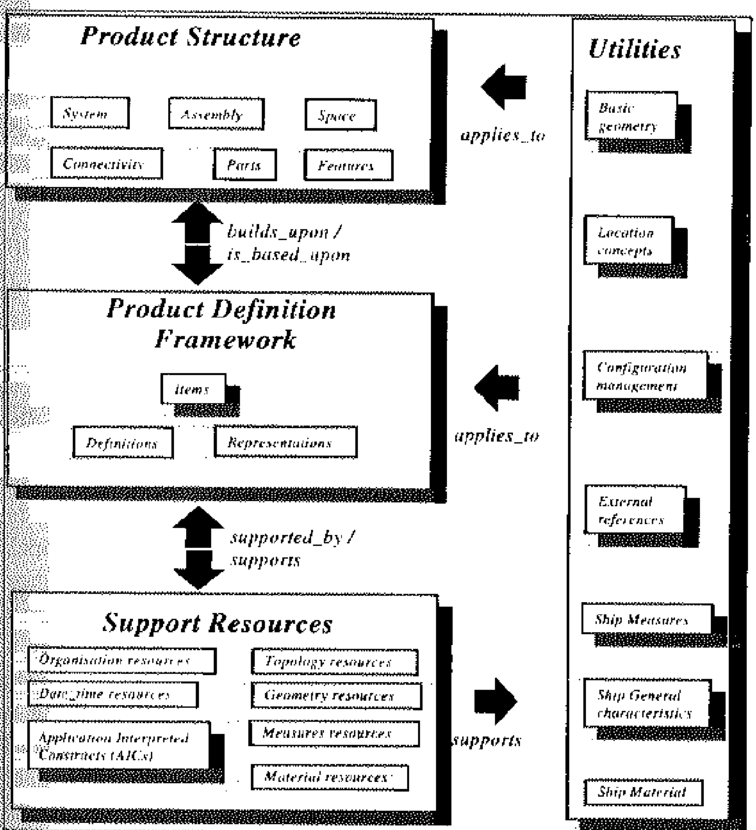


Figure 9: SCM Planning Model

SCM and ARM Level Use

The primary use for the SCM has been at the ARM level and in particular to help co-ordinate the different uses of the Building Blocks across the different APs. Apart from the main drivers discussed above, the SCM also helps to overcome different modelling styles and reduces the need to develop independent ways of organising each model. This might be especially true in cases where there is more than one way to model the domain. By providing a flexible backbone, with ready built viewpoints on the domain, the modeller is free to examine how best to make use of the constructs given, rather than develop their own mechanism to represent the view of the domain. This mechanism is something frequently only discovered towards the end of the modelling effort, when the strategy of the model is mature and a consistent approach to the model's organisation is required. Hence the modeller can instead, concentrate on the issues at hand and specific to the domain.

Such principles are not new, but it can take time before those benefits, which seem expensive to develop for a single AP, are then seen to be re-used and provide additional benefits for all. For example, when considering[3] several separate models, it is often useful to know how the model is structured and what underlying principles have been used. Where each model has a different structure there is less in common and therefore, more difficult to follow. Where the structure is similar, there is more in common and therefore, allows the user to trace and/or follow patterns in the model. Such consistency of underlying structures is an important factor in the development of a suite of APs providing reliability, and to a degree, a level of predictability in the model. At the level of this common structural model (or "meta-model"), it can be easier to identify mechanisms for the integration of such models. One feature of using meta-models is the ability to change the entire model, globally, by only a small modification in the meta-model.

From the perspective of implementing translators based on the ARM, many of the benefits found above are also of importance to those developing the interfaces required between proprietary systems and the neutral format of the STEP model. In particular, generic functions for navigating, accessing and processing the data in a STEP file or database can be developed with specific functions for each AP where they cover unique aspects of the domain or where the viewpoint is different. By re-using such functions and algorithms, this allows the developer to concentrate on ensuring that the mapping between the proprietary and STEP models are correct, perhaps refining the algorithms as required.

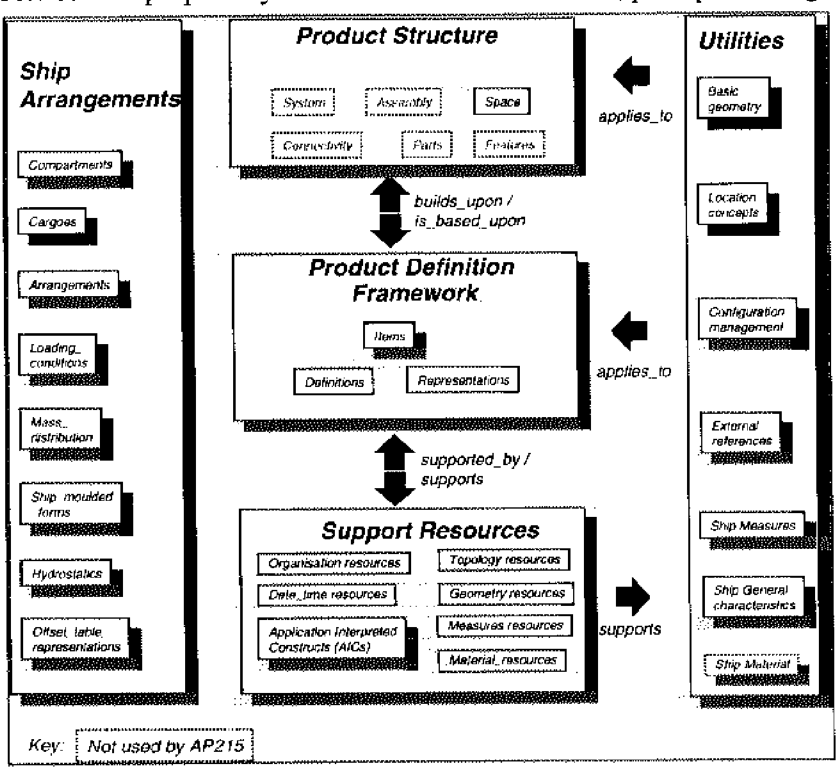


Figure 10: SCM Planning Model for use with AP215

The SCM provides a mechanism for setting each AP into context through the specialisation of the planning model. This powerful tool allows the AP to identify which parts of the SCM are being re-used for that specific AP and which are not. It enables the user to instantly see how the AP is related to others by assessing which parts are held in common between them and those that are not. In the same way it also shows areas that are specific to the AP and not in the SCM. It can also be used to assess (subjectively) the level of conformance to the SCM. For example, the planning model of the SCM shown below has been specialised to show the possible re-use of the SCM by the AP for Ship Arrangements (AP215). From this figure it is possible to infer that AP215 uses the

[3] Especially when reviewing models of significant size

entire Framework, one viewpoint of the Product Structure suite, most Utilities and Supporting resources. However, it also tells us that a number of new components have been added, such as the Compartments, Cargoes and Arrangements UoF's, for example, shown on the left-hand side.

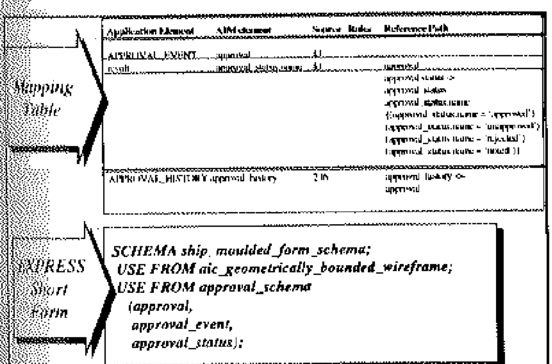
Due to the distributed modelling environment managed by EMSA, a change to one of the Building Blocks within the SCM means that it only need be changed once, all other APs which use that part will need to only re-import the latest brick to be conformant again.

SCM and AIM Level Use

The documentation of the AIM consists of Clause 5.1 (Mapping Table) and Clause 5.2 (AIM Shortform). In addition to this, the Interpretation Team generates an Interpretation Report describing the methods, assumptions and agreements used to carry out the interpretation. The AIM Mapping table shows how each UoF & application object of the ARM maps to one or more AIM constructs. It is often the case that whilst carrying out the interpretation process, minor inconsistencies in the ARM are discovered and decisions on the interpretation are dependant on how these are resolved. Ultimately, this results in an ARM issue, which contains the proposal of the AIM development team. Therefore, the ARM is never complete until the AIM is finalised.

Mappings are described through mapping paths that target one or more entities of the Integrated Resources. These can then also be constrained by Mapping Rules to restrict the conditions under which that mapping is to be used.

Figure 11: AIM Mapping Table & Shortform



The entities used from the Integrated Resources are contained in a number of EXPRESS schemas known collectively in STEP as "the Part 40's". These schemas are selected based upon the entities required as identified within the mapping table. Hence an AIM EXPRESS Short-listing can be developed for use within Clause 5.2. By resolving the Short form listing, a complete EXPRESS model (longform) can be generated.

However, the Mapping Rules from the document also need to be written in EXPRESS as either local or global rules to maintain the semantics. Occasionally, there is no individual IR entity to map to which provides the correct attributes. In such cases the mapping may branch in a number of directions identifying more than one target where the use of one provides the context on another. The complexity of the mappings can easily become un-manageable when dealing with large models. For example, early results for AP216 (Ship Moulded Forms) are indicating that there will be over 400 pages of tables (just for the AIM Mappings) and this AP is the smallest of the Shipbuilding APs that are active.

Interoperability of the final model with other translators must be considered when developing the AIM since the first instinct is to follow the same method of abstraction of the ARM. In the Ship AP ARMs the abstraction method is largely through the use of subtype/supertype constructs to capture the abstraction, and this was used in the initial mappings to the IRs. However, it became apparent that the IRs make more use of entity to entity associations and association entities rather than subtyping and introducing a new subtype of an IR entity specifically for an AP. In fact doing this will immediately make that part of the AP un-interoperable if this would be the only AP implementing it. Interoperability is always broken as soon as the basic constructs being used across the different AIMs are changed in one of the translators. Hence an agreement was made to re-map the APs without the introduction of subtypes. This was also in the light of other APs, such as the AP214 Team, (Automotive Design) who explicitly removed all AIM subtypes where possible to be more interoperable with other APs.

Other mapping agreements have been made between the shipbuilding AP teams regarding issues around; AIM subtypes for Management Resources and PLIB[4] Services; mapping of data type semantics, unnecessary attributes in target entities, mapping of ARM attributes that do not have direct AIM equivalents, and a range of general agreements around the mapping of the Ship Common Model. For example, as the SCM Framework is organised in a hierarchy by item/definition/representation, this hierarchy should be mapped to the Integrated Resources into a similar manner (see Figure 12) since this was possible without the introduction of new subtypes. Other mapping agreements are shown graphically below in Figure 13.

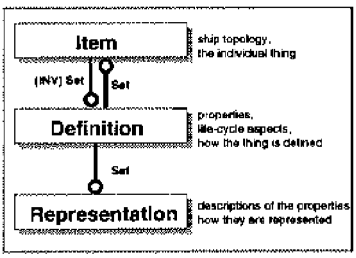


Figure 12 SCM Framework

For example;
- Items map as follows:
- ship maps to product which is the only product in all ship APs
- parts, systems, assembly and moulded_form map to product_definition
- features map to shape_aspect
- item_structures should map to group

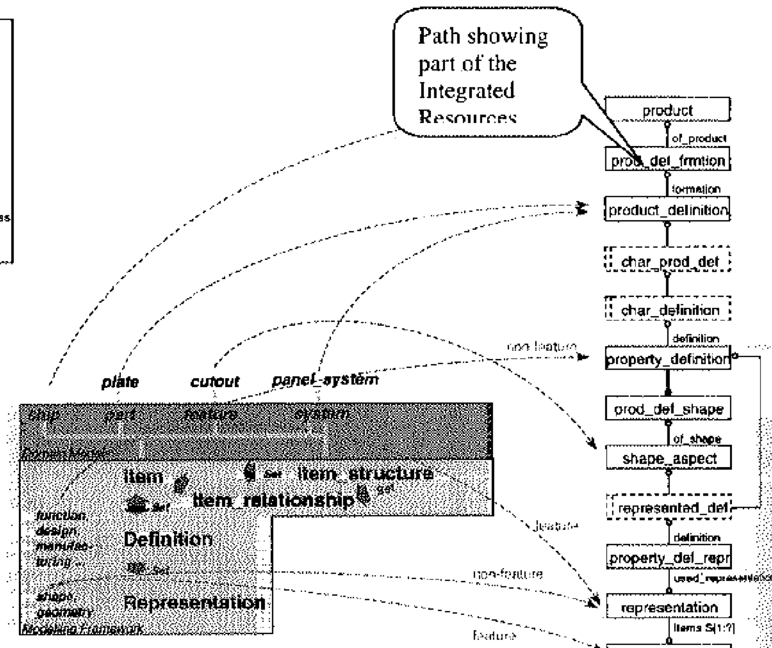


Figure 13 Early Mappings of the SCM

[4] Parts Library ISO 15584

- Subtypes of both, item and item_structure (like ship_moulded_form or system) shall map to a complex instance of a mapping destination of item (product_definition in this case) and group
- Definitions should map as follows:

- to product_definition if the defined object maps to product
- to property_definition if the defined object maps to product_definition
- to representation if the defined object maps to shape_aspect

- Representation should map to representation
- Representation_item should map to representation_item

Such agreements are required because there are many ways to traverse the IRs from the initial entry point on the graph to the destination or target entity. Of course some of the ARM entities within the SCM are abstract supertypes which may have many different subtypes which may or may not introduce new attributes at lower levels. Such subtypes might map to completely different IR entities thus making mapping at the meta-model level more complex. Traditionally, these supertype mapping would be reproduced from that of the subtypes. However, this makes the tables larger and more complex, where a simple referencing in the tables to such subtypes should be acceptable rather than providing redundant information.

In general, the SCM has provided a source of stability between the different ARMs being interpreted such that a common mapping strategy can be devised. Parts of the strategy are very restrictive (necessary to allow interoperability), and others not so. Most take into account the type of entity, whether it is abstract or not, the presence of attributes, context to the product (the ship) and previous mappings.

The further use of the SCM within AIM implementations is still beyond the scope of this paper since none of the Ship AP AIMs are complete enough at present. However, when looking at the work that is currently under way it is possible to see large similarities between the AIM of AP216 and AP218. These similarities are in terms of the underlying mappings of the models, but also many of the local and global constraints that accompany the mappings, will be the same. This means that many of the functions and code (which will use the rules as a specification) will be able to be re-used from one translator to the other. This will ensure interoperability between the Ship APs. Lastly, because the Ship APs have not introduced any Ship AP - specific subtypes in the AIMs, it means that most other translators who have used the same Integrated Resources of STEP will be able to recognise the data from the translators built to the specification of the Ship AP AIMs.

One final comment on the use of AIM-based translators is that since ARM-based translators are currently being developed and tested, it means that we shall have the ability to test these translators back to back, or at least to compare the results to previous exchanges of data.

EMSA Business Cases

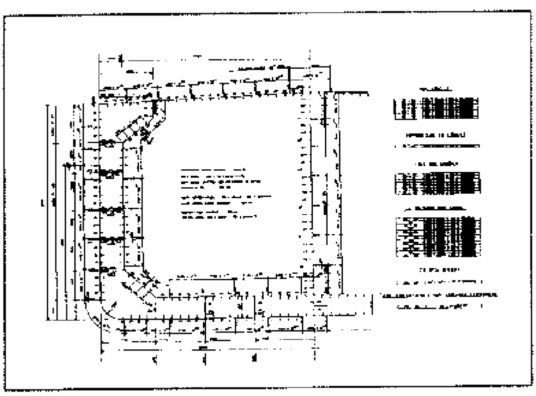
Business cases are identified by members of EMSA where the exchange of data can be shown to provide a commercial benefit and aim to develop pilot (ARM-based) translators based upon these using the methodology described below. The scope of the exchange must be large enough to provide a significant cost saving if implementation is to be carried out, but also small enough that it can be developed rapidly. Ultimately, EMSA lobbies the respective Application Protocols (APs) by raising issues against the AP at the ISO level, or to accommodate these requirements as the basis of conformance classes on the AP. These business cases are also known as EMSA Protocols. The business cases, having identified their data requirements, can review the Building Blocks and map those requirements to the Building Blocks. This provides both a tabular documentation of the requirements and forms an input for the shortform required in order to generate the EMSA Protocol.

EMSA has a number of business cases that are currently active. One centres upon the interaction of a shipyard with a classification society during the design approval phase of the lifecycle, whilst the second involves a shipyard or design office and a model basin, again during the design period.

The Hull Cross Section Business Case

Hull cross-section information is commonly exchanged between shipyards - or other engineering offices - and classification societies during the design phase of the ship life cycle. The exchanges are for the purpose of plan approval of the shipyard design. A hull cross-section is the collection of all those parts of the ship structure at a specified longitudinal position that are relevant for longitudinal strength. For the analysis the ship is viewed as a beam, with a length of approximately the length between perpendiculars and with cross sectional properties that vary over length.

Currently the required data for approval are mostly communicated using technical drawings. The drawings are output from ship design and construction systems. At the classification societies offices the drawings are either checked manually against the rulebooks, or they are entered into an approval system. Based on this input the classification society produces an approval report that highlights deficiencies of the ship design if this is necessary. The deficiencies may be noted down by red-marking the drawing. In case of computer-aided approvals an automatically produced report returns detailed results to the shipyard. Already for the near future it is envisaged to use digital data exchange among the ship construction systems and the analysis packages of the classification societies. The implementations based on this business case specification will contribute to the new way of working. They will imply that electronic copies of three dimensional ship designs are sent to classification societies via e-mail or similar means. Nowadays hull cross section approval is still based on two-dimensional data, with all the limitations this means for secure analysis.



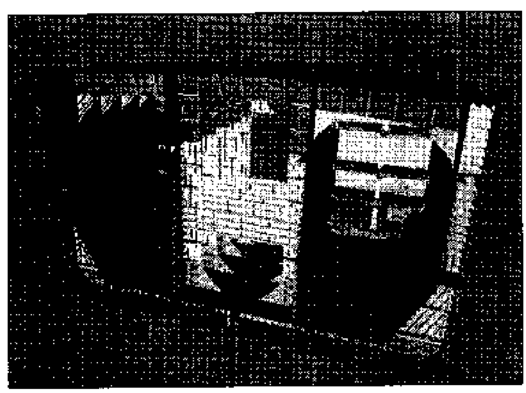


Figure 14: Typical 2-D Mid-Ship Section Plan Figure 15: Some 3-D Hull Cross Section elements

Without major human interaction the ship construction data will be read into a ship structural analysis and approval package. Deficiencies of the design will be recognised and corresponding notes be added to the received model. The validated and marked digital model will be returned. It will be imported into the construction system of the design office and changed to satisfy the classification society comments.

This future scenario leaves the roles of the players in this business case untouched; design and analysis are strictly separated. Such activities today typically take in the order of six weeks, but with this scenario it is possible to foresee that at least a 50% reduction in this lead is possible. However, with the new data exchange technology the process can be made even more efficient. The analysis tools can be given to the designers so that they can check their draft designs immediately; no cross-company communication would be required. The classification societies would finally be handed a ship design that has already been verified and which only needs a final approval. In summary, the shipyard or design office may in the future do the verification by analysis themselves. This, however, requires a seamless communication between ship construction and structure approval systems, as aimed for by this business case specification. This business case is seen as the basis of a conformance class for AP218 Ship Structures.

Towing Tank Business Case

The other business case revolves around the interaction of a shipyard and model basin during the generation and testing of the hull form. This business case is seen as the basis of a conformance class for AP216.

The moulded form of the ship is sent to the model basin from the yard for a series of hydrodynamic tests. The hull shape is imported into the model basin's computer aided modelling system in order to produce the physical model. After the tests the model basin uses the test results to suggest a modified hull form that is sent to the yard. In this iterative process, the information may need to be several exchanges made between the shipyard and model basin.

A model basin typically handles one hundred ship geometries per year. This geometry handling implies the transfer of information from paper drawings or a variety of digital forms, to a format readable by the model basin's CAD systems. Each transition requires three to five man-days work for translation and verification of the data unless the CAD systems used by the model basin and shipyard are the same and if the information is paper based additional time is required for postal transaction.

A similar situation exists in the application of Computational Flow Dynamics (CFD). All CFD tools

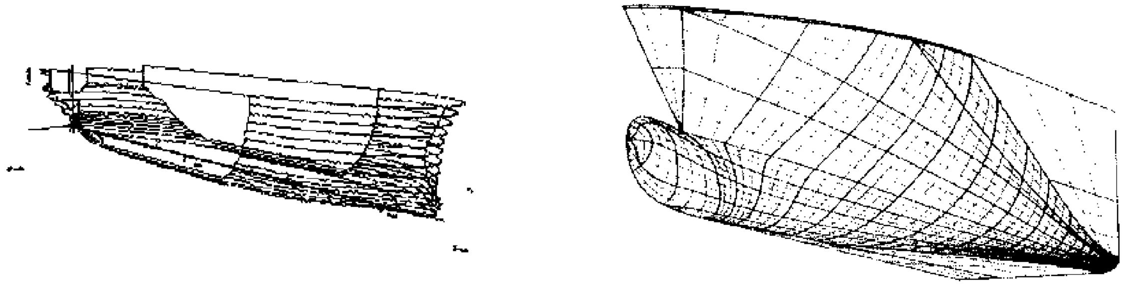


Figure 16 Examples Moulded Form using Traditional Lines and Surfaces

used require information about the hull shape to be translated to a proprietary format for further processing. When the shipyard receives the modified hull form it is usually transferred into two systems. It is translated to the native CAD format used by the yard for further design and eventually building of the ship. It is also archived in its current format for future use.

With the introduction of STEP, each of the companies (the shipyard and the model basin) uses translators from their CAD system's format to the STEP file format. Such translation requires around quarter of an hour. The electronic transfer of the file will typically take quarter of an hour using standard systems. The shipyard can also use 'hull moulded form' STEP models to help integrate the different phases of the ship design, build and through life support.

EMSA has a number of other Business Cases being developed for the following types of data;

- Design Data - A yard plays the role of the Main Contractor (MC) who subcontracts out a part of the ship's design. The Engineering SubContractor (ESC) then in parallel to the MC, produces a detailed design. It is possible to update modifications made by the MC into the system of the ESC, and in the reverse situation, the MC has to add modifications to the data from the ESC into his system. Collating all the design data, the MC sends a complete & detailed design to a Production SubContractor (PSC). Reality shows that there is no straightforward flow of data and several iterations are needed to complete the detailed design and to send necessary modifications to the production yards. So it is necessary to use also use tools for configuration management of the data.
- As-built data - When the ship is completed the as-built ship product data will be transferred to the ship owner and the classification society. This hull definition data can then form the basis for systems used for monitoring the condition of the hull and recording the results of surveys. The ability to exchange the as-built hull data automatically between the computer systems involved will reduce by several weeks the time taken for the classification society to configure these systems according to the specification of each different ship.

Other suggestions have been made but with only finite resources, only those above have been defined fully.

Business Case Implementations

EMSA supports implementation projects such as SEASPRITE[2] and CALYPSO[5] to help develop and implement the business cases that have been identified. SEASPRITE is a 3-year project, involved in the implementation of translators using the EMSA Business Cases described above. These were reported in EMSA's Newsletters[3], showing successful results of the prototype translators dating from January 1998 to May 1998. Work has continued on these translators and is also being reported at ICCAS [4][6][7] elsewhere this year. It also possible that these translators will be demonstrated at other events during the year[5]. The CALYPSO project has also been using the Towing Tank business case to exchange hullform data between shipyards and model basins for use with Computational Fluid Dynamic (CFD) analysis.

A similar initiative has also been active in the USA under the MariSTEP[8] project, who have been developing similar schemas for the US shipbuilding industry. These initiatives are broadly complementary and will hopefully be brought together under mutual co-operation at the level of EMSA and NIDDESC[6].

[5] For example, PDT-Days Europe, Stavanger, Norway April 15/16 1999. ISO TC184/SC4/WG3 T23 Meeting, Lillehammer, Norway June 1999 or NorShipping, Stavanger, Norway June 1999.

[6] Navy Industry Digital Data Exchange Committee, equivalent organisation to EMSA in the USA

EMSA is now investigating the mechanisms to help verify to which level the translators produced according to the EMSA Protocols actually conform to the specifications. This is seen as a necessary step towards acceptance from the industry that has now started to actively request for software conformant to STEP. Those EMSA Protocol schemas that act as conformance classes on actual Ship APs will of course become obsolete when those APs are finally produced, but such translators will not become redundant for some time.

Immediate Future

By putting ARM-based translators into use as soon as possible, the impacts by which the business processes themselves will need to change will begin to become apparent with the associated the business benefits. When the AIM translators in the future replace these, most customers will only be aware of an update to the software being used. ARM or AIM usage will become completely transparent. Hence the development and use of ARM-based translators will not be wasted when the AP AIMs become standardised, for two main reasons. Firstly, the former translators will provide a mechanism for validation of the AIM-based systems ensuring that the semantics of the ARM have been maintained in the AIM. Secondly, by using mapping languages such as EXPRESS-X, a further (pluggable) processor would be able to take the ARM-based instance data from any existing system and to translate those into AIM-based instance data (and vice-versa for importing data). This will provide an upwardly compatible add-on to make ARM-based translators conformant. A project to research this idea has already been set up (SEANET[18]).

EMSA is also interested in ensuring that the Building Block methodology is upwardly compatible to such new initiatives as the Modules Approach. This could involve a mechanism for merging of Building Blocks into Modules at the level of UoFs. At this level, the AIM specifications could then be added since this is the level at which the interpretation process is carried out. Hence a new "UoF short-listing" could be designed to accompany the existing one for Building Blocks.

Conclusions

The advantages of the Building Block approach for the development of the Ship AP ARMs has been shown. The approach has also been shown to help AIM development. Groups[7] within ISO have now been building upon this and other approaches to define a new "Modules" approach, which further confirms this as the type of approach required in AP development in the future.
EMSA provides a significant European input to the development of the ISO APs through the Business cases that focus attention on industry bottlenecks. It also provides support for the continued use of the Building Block methodology, the EMSA Server and Ship Common Model. EMSA will produce a set of guidelines on how translators for the EMSA Protocols will be verified such that customers can be satisfied that the underlying technology has been proven. Industry needs working solutions now. The ISO process is a necessary but voluntary and expensive one, which requires consistent and sustained effort before the final fruits will become available. Until this time, ARM based translators will provide the drivers for today. Modellers and vendors should seize the opportunity to stimulate the market now.

[7] ISO TC184/SC4/WG10

References

[1] ISO 10303-1:1994, Industrial automation systems and integration. Product data representation and exchange Part 1: Overview and fundamental principles.

[2] ESPRIT 22390: SEASPRITE - Software Architectures for Ship Product Data Integration and Exchange, Available from Internet: http://www.irtpdd.demon.co.uk/seasprite.html

[3] EMSA News: Vol5 N0.1 January 1998 and Vol5 N0.2 May 1998, edited by T.Turner. Lloyd's Register of Shipping, London, UK. Available from Internet: http://www.oss.dk/emsa/e-news/emsanews.htm

[4] Kendall, JMP "Ship Product data Interchange to Support Ship Design Processes". 10th International Conference on Computer Applications in Shipbuilding (ICCAS 99), June, 1999,Boston, USA

[5] Brite/EuRam BE95-1721 Computational Fluid Dynamics in the Ship Design Process. Available from Internet: http://www.oss.dk/resdev/internationalprojects/calypso.htm

[6] Catley, Don "Prototype STEP Data Exchanges in Ship Initial Design and the Provision of an Applications Programmer Interface to 'TRIBON'", 10th International Conference on Computer Applications in Shipbuilding (ICCAS 99), June, 1999,Boston, USA

[7] Grau, M and Koch, T "Applying STEP Technology to Shipbuilding", 10th International Conference on Computer Applications in Shipbuilding (ICCAS 99), June, 1999,Boston, USA

[8] Howell,J, Penley, R, Lisle, R, and Lazo, P "An Implementation of STEP Translators for Shipbuilding Protocols Based on an Object-Oriented Product Data Model Approach", 10th International Conference on Computer Applications in Shipbuilding (ICCAS 99), June 1999, Boston, USA

[9] Turner, T & Mays, J, "Minutes from the ISO TC 184/SC 4/WG 3/T 23 (SHIPBUILDING TEAM) STEP Meeting, Bad Aibling, June 1998"

[10] Turner, T "AP Development guidelines for Shipbuilding", November 21st. 1997, ISO TC184/SC4/WG3/N701

[11] ISO 10303-215, Industrial automation systems and integration. Product data representation and exchange Part 215: Application Protocol: Ship Arrangements.

[12] ISO 10303-216, Industrial automation systems and integration. Product data representation and exchange Part 216: Application Protocol: Ship Moulded Forms.

[13] ISO 10303-217, Industrial automation systems and integration. Product data representation and exchange Part 217: Application Protocol: Ship Piping Systems.

[14] ISO 10303-218, Industrial automation systems and integration. Product data representation and exchange Part 218: Application Protocol: Ship Structures.

[15] ISO/WD 10303-218, Industrial automation systems and integration -- Product data representation and exchange – Part 218: Application protocol – Ship structures – Interpretation report. ISO TC184/SC4/WG3 T23, 1998-05-18

[16] ESPRIT 2010: NEUTRABAS - Neutral Product Definition database for Large Multifunctional Systems.

[17] ESPRIT 6041: MARITIME - Management and Reuse of Information over Time.

[18] A.R. Healey "Virtual Enterprise Support Architectures in the Maritime Industry" 9th International Conference on Computer Applications in Shipbuilding (ICCAS 97), October 13-17, 1997,Yokohama, Japan

NEUTRAL FORMAT DATA EXCHANGES BETWEEN SHIP PRODUCT MODELS AND ANALYSIS INTERFACES

Alan K. Crawford and Greg Harrington, Newport News Shipbuilding, Newport News, VA
Robert M. Ames and Richard T. Van Eseltine, NSWC, Carderock Division, West Bethesda, MD

Abstract

During the past decade, significant effort has been expended on the development of neutral format exchanges of CAD and, more recently, Product Model data. These efforts include IGES, PDES, NIDDESC, and the more recent STEP initiatives. These efforts have been driven by the need to exchange data within or between companies and government agencies. There have been many successful projects which depended on the exchange of data using these neutral formats, but most have concentrated on the exchange of data between similar systems, primarily CAD and drafting systems. This paper addresses the need to access data from emerging Product Model systems to feed legacy analysis systems. A data exchange was developed and tested between a ship product model and several naval architecture analysis systems. The transfer included, among other attributes, the molded form definition, compartment definition, and connectivity information. To support this effort, a server was developed to provide the ability to create, query, and retrieve the important geometry and relationships for ship molded forms and compartmentation for a given analysis task or set of tasks. An interface was created to load the server from a Newport News Shipbuilding workstation VIVID® product model using a modified STEP format file. On the other side of the server, interfaces were developed for several standard analysis programs. This paper will provide an overview of this project including the geometry server, the interface to the product model, and the interfaces to the analysis programs. It will discuss the project philosophy, the attempt to follow the existing STEP standard and to use existing software technology, and noted capabilities and deficiencies in the currently available standards and technology.

Disclaimer

Introduction

Two years ago, the Naval Sea Systems Command initiated a task to study data exchange between design and analysis data. The concept was developed by a group of Navy personnel as a benchmark to measure progress toward establishing a collaborative modeling and simulation-based engineering environment within the NAVSEA community; and, to provide recommendations to the CAD and standards development communities as to the requirements associated with the future viability of the NAVSEA collaborative environment.

The focus of this initiative was to design and implement software necessary to take geometric product data and perform analysis on it. The test of this capability was demonstrated using Newport News VIVID® product model geometry and PME equipment data with the analysis performed using the Ship Hull Characteristics Program (SHCP), and the Ship Vulnerability Model (SVM). Software tools and services were designed to provide a means to communicate product data to the analyst in the form that they require in order to perform their functions. The approach was not to perform these services in a way specific to the VIVID® geometry or to any specific analysis, but rather in a generic fashion applicable to many types of analysis and multiple sources of product model geometry.

Because of the wide variety of geometry models and the lack of "intelligent" topology in most CAD systems necessary for complex design, it was determined that a new geometric topology was critical to success in translation to many discipline-specific analysis applications. The use of standards such as STEP to translate product data from product model geometry to a common geometry server was pursued as the desired approach; but liberty was given where necessary to allow extension of the standard where necessary to support this new topology and its new geometric entities.

The requirement for intelligent geometry is particularly evident when modeling and analyzing complex parts such as naval combatants. The resultant effort by the Naval Surface Warfare Center, Carderock Division (NSWCCD) to provide this intelligent geometry as a service to applications was one of the major outcomes of this effort. The implementation of this capability, called Geometry Object Structure (GOBS), was a Common Object Request Broker Architecture (CORBA) service that provides the required CAD-to-CAE linkage. Implementing GOBS as a CORBA service was determined to be more beneficial than a flat neutral file because it provides methods as well as data with the geometric objects. This allows applications to use the service to interrogate the geometry and/or construct additional data entities like section curves without requiring each application to possess an equivalent geometry engine. Each analysis application was also wrapped as a CORBA client/server object.

The demonstration of this environment with associated test applications and product model data is shown in Figure 1.

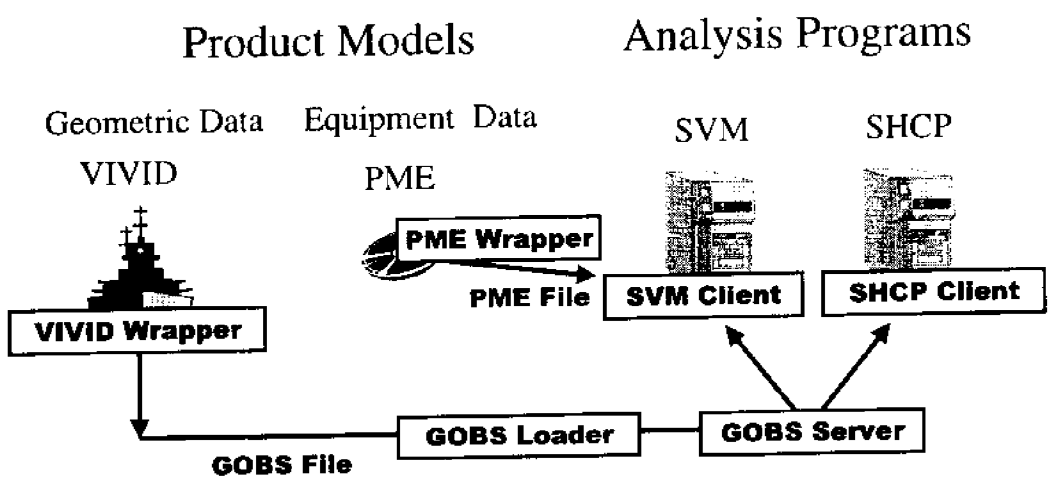


Figure 1 – Demonstration Environment

Product Model Technology

There is no single accepted definition of a "Product Model" in industry or even within the shipbuilding community. It has been used to describe geometry based Computer Aided Design (CAD) models and solid topological models. The Newport News Shipbuilding view of a product model goes beyond these definitions to describe a product model as an all-encompassing model of a ship project. A ship product model takes advantage of the available computer tools for describing the geometry of a ship, but it also includes all attributes necessary for the design, production, and operation of the ship.

The product model for a product as complex as a ship is not likely to be defined by a single computer system. It will likely require the use of a series of systems that work within their own domain and communicate with other systems through public interfaces. A simplified view of some of these domains is shown in Figure 2. The division between domains is arbitrary. Some may choose to combine the production planning into the design and geometry definition model. Others may split the design and geometry definition model into modules for design, detailing, and manufacturing.

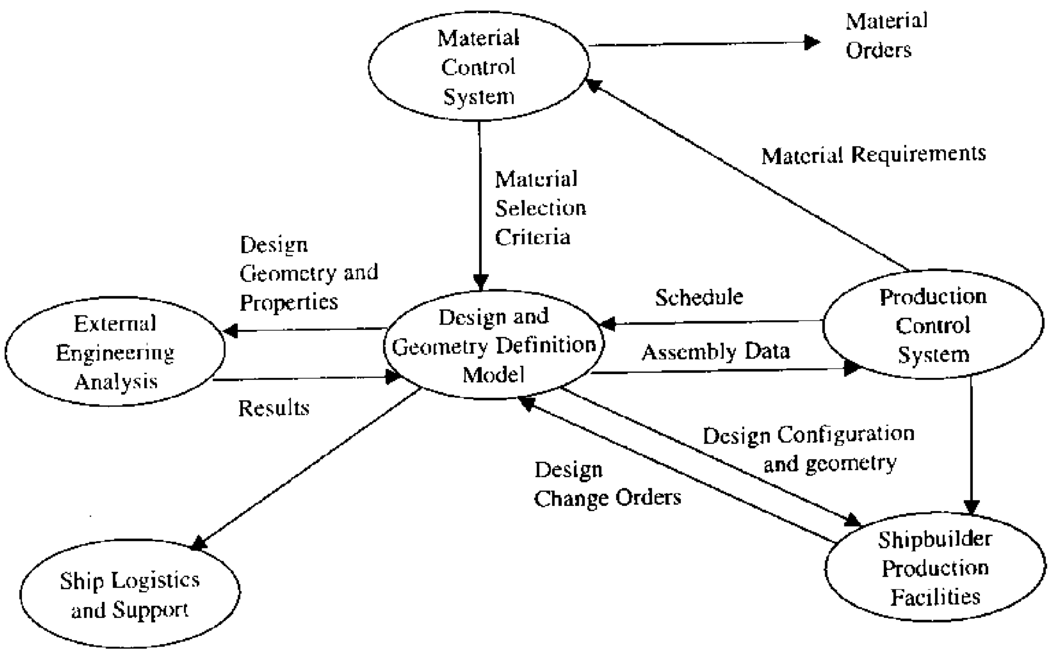


Figure 2 – Demonstration Environment

Newport News Shipbuilding has chosen to concentrate product model development on the design and geometry definition module. The key to this model is to capture the information needed at each step in the process to allow each step in the process to build upon the outputs of the previous step. The model presents each user with a view of the model suitable for a specific task. In an ideal product model, each decision will be made once, as early as practical, and propagated through the remaining steps in design and construction. Changes to the design are accomplished by changing the decisions at each step, with the results propagating through to the geometry.

The configuration of the design and geometry model defines an organization, a topology, and a set of representations for the data. The organization allows the data to be accessed by logical groupings or views. The topology defines the logical and physical relationships between the model elements. The representations are generated as a result of the topology. The focus of this project was structural design and this focus will be used to simplify the definition of the above concepts.

The primary organizational elements are systems, spaces, and assemblies. Systems define the functional elements for the ship. This includes plate systems and profile systems. Plate systems are decks, bulkheads, hull shells, and general plate systems. Spaces define the spatial arrangement of the ship. They can be either compartments, which define physical spaces within the ship bounded generally by plate systems, or zones, which define logical spaces within the ship bounded by arbitrary boundaries or by plate systems. Assemblies define the construction sequence for collections of parts. An assembly can be defined either bottom-up from a set of parts or top down as all parts contained within a geometric volume.

The topology of the model is defined by connectivity and bounding relationships between elements. These two concepts are integrally related, in that a bounded object is logically connected to the bounding object. However, it is possible to have connected elements that are not involved in a bounding relationship. This topology is crucial to the generation of the final geometry from the base geometry and decisions defined at each step in the process.

Representations are views of the geometry for a ship. Each representation may contain a combination of independent geometry not affected by the topology and derived geometry that is dependent on the topology and related geometry. The supported representations are design, detail, and manufacturing. These three views are shown in Figure 3 for a sample bulkhead.

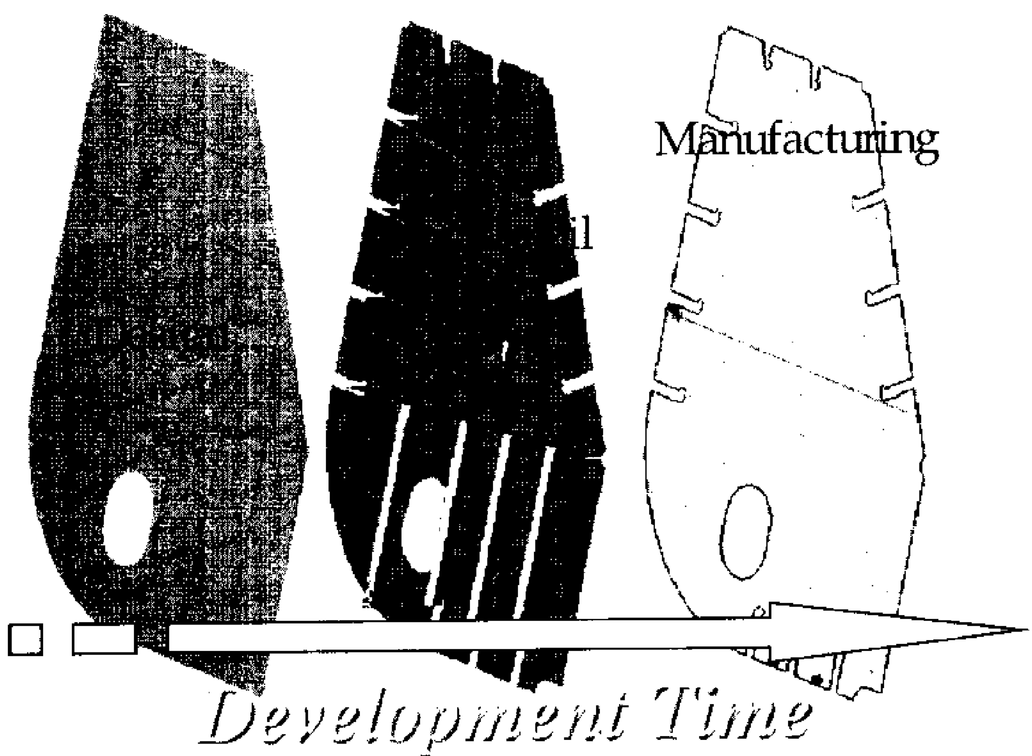


Figure 3 – Three Representations of a Bulkhead

Consider the example of a deck system. Initially the deck is a concept that a deck is needed to serve some purpose. The exact height for the deck may not be known, but the deck can be defined as bounded by two frames and the hull shell. When the deck surface and the surfaces of the frames and hull shell have been established, the deck will be trimmed by this bounding geometry. A planar deck surface can be defined by a single value for the deck height. The design representation of the deck is the trimmed face defined by the deck surface and its boundaries. The deck can be assigned attributes, such as thickness and tightness, that will drive the generation of part details on the system, but these details will not be shown in the design representation. The deck can also be split by seams to indicate changes in the system attributes for different portions of the deck.

When parts are generated on the deck, they automatically inherit the properties of the parent deck, including surface, boundaries, seams, thickness, and tightness. These attributes are used to generate the detailed representation of the plate part. Unlike the design representation, this detailed representation shows the effects of thickness and other details. For example, it is a solid that shows its true thickness, the edges have been trimmed to account for the thickness of bounding plates, and cutouts are present on the edges in way of stiffeners and seams. Ideally, this representation can be generated automatically from the system and topology definition, but it may require some alterations at the part level.

The part contains attributes, generally driven by rules, that define the manufacturing requirements for the part. These attributes can include such items as added material for assembly, weld bevel details, and weld shrinkage factors. With this information, it is possible to develop the manufacturing view of the part from the detailed view. This view depends on the specific manufacturing plant capabilities, but it is generally a curve based geometry that details the edges and marking lines on one or both faces of the plate. For curved plates, it includes the flat pattern for cutting the plate and may also include templates required for forming the plate.

The space model of the ship defines the boundaries of the spaces via relationships to the bounding plates and arbitrary geometry. It also defines the attributes of the space, such as tightness. The connectivity between spaces is also defined, along with connection attributes that define accessibility between spaces.

The assembly model of the ship is a hierarchical taxonomy where the ship is divided into assemblies and parts. Each of these assemblies is also composed of assemblies and parts down to the lowest level assembly. An assembly can also contain a topological definition that defines a region within the ship. In this case, the list of parts and assemblies is derived.

Geometry Object Structure (GOBS)

The GOBS technology allows CAD system geometry and attributes to be presented to engineering modelers and analysts in a form that allows for convenient discretization according to the requirements of their models. The GOBS model purports that geometric product model data is defined and represented as 'views' of geometric objects. The word "view" is in quotes because it is actually an object that appears as geometry. This is not to say that GOBS does not allow geometric objects to represent geometric product model data only that another more powerful approach is available. This is contrary to most CAD representations where the geometry defines the view and the object simultaneously. In addition, GOBS contains connection entities that define common boundaries between objects like the intersection at a deck edge and the hull. The boundary of the deck knows where it is located on the hull and visa versa.

A discussion of some of the GOBS objects follows. It is important to understand that due to space constraints not all GOBS objects are discussed nor are the various methods available to applications as a CORBA service.

To understand one aspect of this new geometry topology, an example of three compartments within a ship, depicted in Figure 4, is illustrated. This three compartment case, while simplistic in appearance, actually poses a number of challenges to product modeling. Consider the "knowledge" that must exist at transverse bulkhead 2, (Trans-2). This bulkhead plays a number of roles one of which is the boundary of three compartments.

The bulkhead is connected to the hull, port and starboard, the longitudinal bulkhead, and both decks above and below. In addition, there are locations on this bulkhead that may be of interest to analysts such as the corner points at intersections with other surfaces (longitudinal, hull, deck, etc.). This bulkhead also plays a role as a boundary, or Face, of each individual compartment. These boundaries can be described as "views" of the bulkhead as seen by each compartment and unique to each compartment. Consider also that the object Trans-2 may play a role, or roles, in many other "views" such as a watertight bulkhead bounding a zone on the ship.

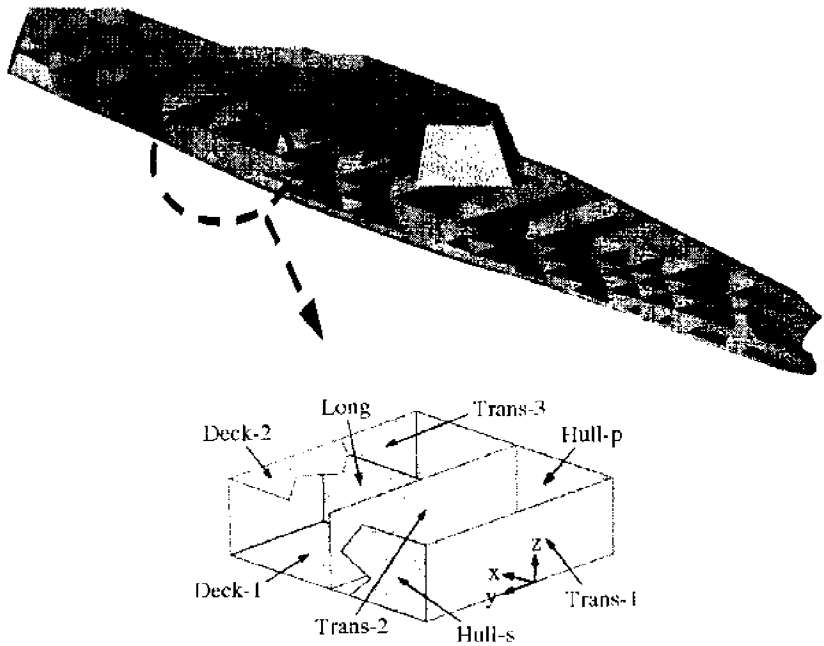


Figure 4 - Three Compartment Test Case

Product Model Views

In GOBS "views" of product model data are actually objects that compose existing geometry into unique physical objects. Similarly there are views that associate physical objects into like groupings. Views that create physical objects from geometry elements are called *Topological Views*. Views that associate *Topological Views* into common groups are called *Common Views*.

<u>*Topological Views*</u>

The term *Topological View* is foreign to most familiar with geometric modeling. It's best to think of them as traditional surfaces, trimmed surfaces, and Brep solids, with additional capability. The construction of *Topological Views* allows for member shape objects, like surfaces and solids, used in the creation of a *Topological View*, to also play a role as geometric members in others *Topological Views*.

<u>*Common Views*</u>

Common Views do not have any spatial constraints. Unlike *Topological Views*, they are simply a logical grouping of *Topological Views*. *Common Views* can also have other *Common Views* as members. *Common Views* are the primary vehicle by which domain analyst or designers will view or interrogate the product model. One example of a *Common View* could be "Habitability Spaces on Deck 3". Another *Common View* called "Ship Habitability Spaces" could contain the *Common View* "Habitability Spaces on Deck 3" as a member. Similar uses of *Common Views* could include "Exterior Surfaces", "Compartments", "Machinery Spaces", or "Mast".

<u>*Shape Objects*</u>

Some distinctions should be made of the differences between GOBS shape objects and what can be considered typical geometric entities in applications that use and compose geometry such as CAD systems. In GOBS, geometry (*Topological Views*) is the association of shape and *Properties*. Current shape objects are *Surfaces*, trimmed surfaces (*Faces*), and manifold brep *Solids*.

One major difference in GOBS modeling is the representation of *Faces*. Currently CAD systems today consider a *Face* to be composed of a single *Surface* bounded by a single outer boundary and any number of inner boundaries. The typical CAD model does not allow the underlying *Surface* to be used in the construction of any other *Face*. It requires that a copy of that *Surface* be made. GOBS, on the other hand, allows for a single *Surface* to be used in the construction of any number of *Faces*; where the *Face* object contains reference to one *Surface*, one outer *EdgeLoop*, and any number of inner *EdgeLoops*. This concept is illustrated more clearly in Figure 5, where the deck on a ship is shown highlighting three *Faces* used as compartment boundaries. All three *Faces* share a common deck *Surface* and are defined by a selection of *Edges* that compose a bounded *EdgeLoop*.

Because *Surfaces*, *Faces*, and *Solids* are shape objects they have no *Properties*. In GOBS the *Topological View* class associates member shape objects with physical characteristics or *Properties* and can be thought of as a geometric component, or part. The *Topological View* has *Properties* of a physical or performance nature, where the underlying *Surface*, *Face*, or *Solid* object, is simply providing information on its shape. As *Topological Views* are composed, the grouping into *Common Views* is the next natural step.

In Figure 5, *Topological Views* of regions on a deck are illustrated. In this case they appear as "Comp 1 Deck", "Comp 2 Deck", and "Comp 3 Deck". Each *Topological View* use *Faces* ("FA1", "FA2", "FA3") to define the shape of a compartment (*Common Views* named "Compartment 1", "Compartment 2", "Compartment 3"). Similarly these *Common View* compartments are also shown as members of a single *Common View* defining a zone ("Zone 1"). In summary, this example demonstrates how each compartment derives its shape from *Faces*. These *Faces* are defined as simple boundaries (*EdgeLoops*) on single underlying geometric element which is a *Surface* (Deck 1) defining the entire deck shape. These compartments are then associated in a *Common View* to support design domain knowledge.

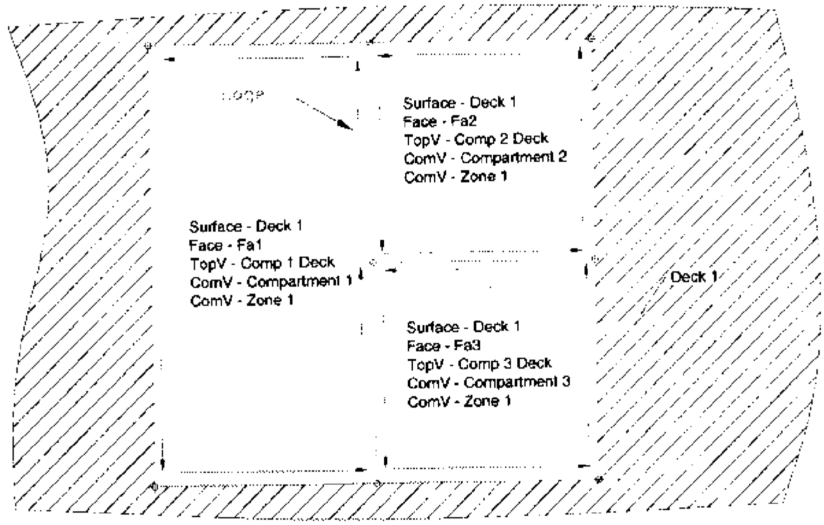


Figure 5 - Three Compartments on Deck 1

Again, GOBS takes the position that topology is a view of space not the space itself. An office room, ship's compartments, or other like space can be viewed as the collection of faces that make the walls, floor, and ceiling. To the occupant the wall of the room extends to the intersection of other walls, ceiling, and floor. The wall, however, may be defined as the space bounded by the outside walls of a building. Thus, the office room could be represented as a list of connected faces where the view of the wall is the region of the larger wall surface with local boundaries applied.

Another fundamental feature of GOBS is the CoEdge object. The CoEdge provides a unique role in the discretization of the geometry for analysis. Essentially, a CoEdge knows all edges located on each surface and declares them to be equivalent in 3 space, see Figure 6. It also contains an n-dimensional spline function, which maps the parameterization of each edge into a single function. This allows for the continuity of points along one surface to migrate to another surface without having to perform closest point approximations.

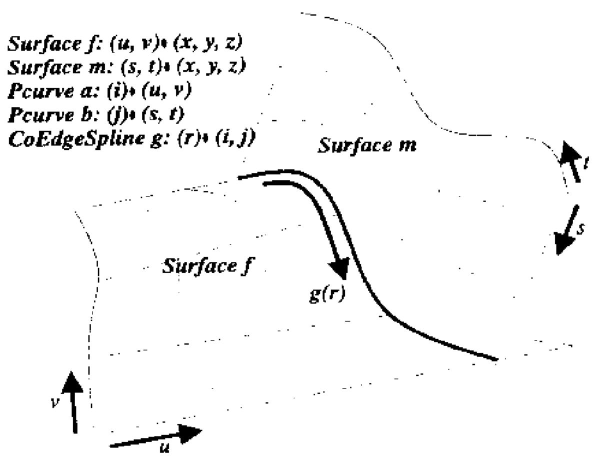


Figure 6 – CoEdge Spline Dependencies

With this topology the ability to traverse boundaries, both logically, explicitly, and with information on the relationship of surface parameter space affords many advantages. Clearly the ability to grid or mesh across trim surface boundaries with node continuity is the most obvious.

With the GOBS objects available as a CORBA service to legacy applications, the communication of a single product model geometry in multiple views provides an efficient and effective means for multidiscipline analysis.

VIVID® Product Model to GOBS Converter

There is a fundamental difference between the definition of a ship in GOBS and within a product model such as VIVID®. The GOBS definition is concerned with a ship definition of topology and geometry. The product model combines a functional approach with topology for its definition. Fortunately, the product model provides the information and relationships required to create a GOBS definition. This is an advantage not shared by typical CAD models.

In the product model, the topology and geometry are not directly associated to all dependent objects. For example, a part on a deck has no surface and a compartment has absolutely no geometry. The GOBS model requires a direct relationship between all objects and its defining geometry. The product model derives this relationship when it needs to access the geometry. Note that this does not imply inefficiency in the GOBS definition or storage standards. The geometry and topology are shared, not duplicated.

Figure 7 shows a much-simplified mapping of the objects in the VIVID® product model required to support the export to GOBS. Of these, the analysis tools to be served using GOBS are mostly concerned with plate systems and spaces. Structurally, a ship can be defined by collection of plate systems. Although the intersection of these plate systems forms enclosed areas, these have no meaning without the definition of a *Space* object. Plate systems are used primarily for structural analysis. Spaces can support any number of other forms of analysis – flooding, HVAC, cargo capacity, and accessibility, to name just a few.

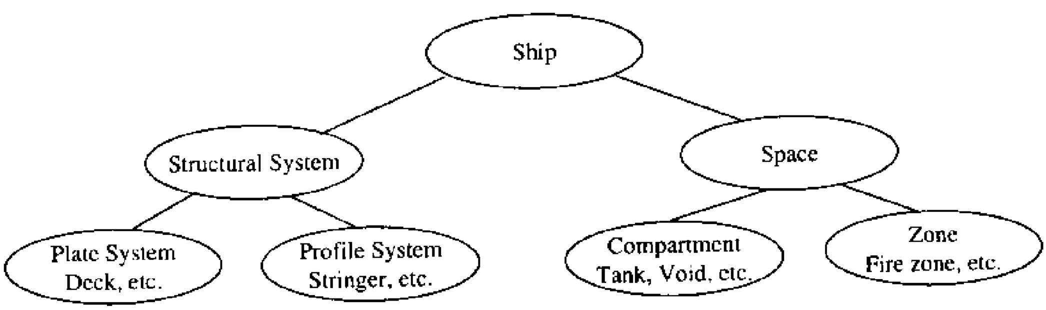


Figure 7 – VIVID® Elements Required to Support GOBS

To illustrate the conversion method from VIVID® to GOBS, Figure 8 shows the process for two compartments. To export the entire ship, this would be expanded to include all plate systems and spaces. First a sheet body is created to represent each plate system forming the boundary of the compartment. The bodies are then united, forming internal edges at the intersections between the sheets. Typically, only a portion of a large plate system forms the boundary for a compartment, leaving unwanted geometry at this stage of the model. The faces with free edges on the united body are removed to leave only the faces on the perimeter of the space.

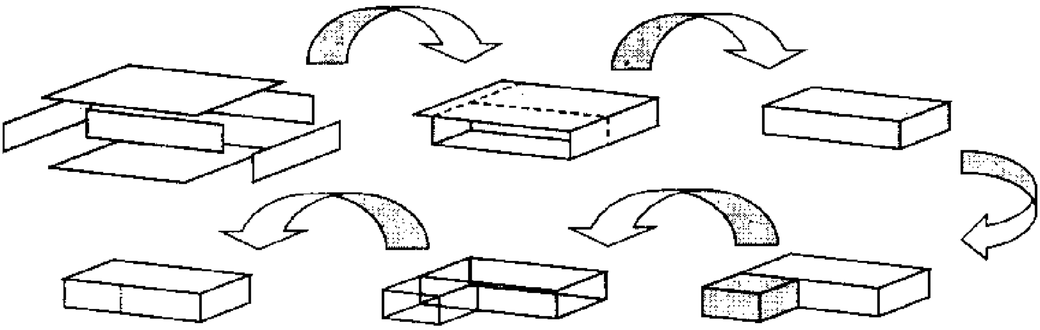


Figure 8 – VIVID® to GOBS Topology Creation

At this point we have the data necessary to topologically describe one compartment. However, we want the various systems and spaces to reference the combined topology for the entire ship. We repeat the first three steps for all other systems (one additional compartment is shown) and join copies of each to form a master model. Each individual model is sliced with a wire frame of the master to arrive at the final topology. Notice that the example compartment shows an additional face in its definition caused by the intersection with the adjacent compartment. This additional face is not necessary to describe the first compartment, but is preferred, since the two compartments can be easily shown as adjacent through their common topology.

With the topology defined, the remaining task is to export the geometry and topology definitions used in VIVID® to GOBS format. The shipbuilding community is moving towards shipbuilding ISO STEP Standards, which will ensure formal and consistent exchanges along with predefined object types and their attributes. AP 216 (Molded Forms) and AP 215 (Ship Arrangements) are both applicable to this project. These standards were employed as a test of the applicability of the standards to our case. Some limitations with the existing standard were noted and required the development of a custom schema to handle certain elements. These deficiencies were passed to STEP development representatives for possible inclusion in a later draft, particularly in Part 42 Geometry. Table 1 offers a summary of the mapping between GOBS and STEP 42.

Table 1 – Entity Mapping GOBS vs. Step Part 42

GOBS Type	Equivalent STEP Part 42 Entity
Surface	Rational B-Spline Surface w/Knots
PCurve	Pcurve
PPoint	Point_On_Curve
Edge	Edge
Face Loop	FaceBound
Face	Face
Solid	Brep w/voids
Coedge and Copoint	N/A
Topological and Common View	N/A

One difficulty was encountered in the generation of the geometry for this project that was unrelated to the primary goals of the project. This represented the first time that hull surface data developed outside of Newport News Shipbuilding was used in VIVID®. Hull surface data for the test case was provided by NSWCCD. This data represented perfectly valid NURB surfaces, but it contained internal degeneracies and other mathematical definitions that the VIVID® modeler could not handle.

VIVID® uses a commercially available modeler toolkit that requires simpler surfaces than those provided for this test. The surfaces used for VIVID® production projects are developed specifically to avoid these problems by defining the hull shell with a large number of simple hull patch surfaces. This is a costly process that was not performed for the purposes of this test. This did not prevent the production of output for the test, but it did limit which areas of the ship could be produced faithfully. The areas that could not be modeled were limited and did not impair the overall success of the project. However, this is an area for further work; either in the area of improving the modeling tools or in making it easier to split a hull surface into simple patches without changing the shape of the surface or introducing gaps between the patches.

Legacy Analysis Tools

Legacy analysis tools play an important role in the design of Naval vehicles. These tools have been verified and validated through years of use. Many contracts specify which tools are to be used for different types of analysis. The analysts understand the limitations of these tools as well as their strengths. This includes the ability to select tools that will produce reliable and accurate solutions under specific situations. Legacy analysis tools represent huge investments of money and time. Thus, there has long been a need to access data from CAD systems for use by legacy analysis tools. With the emergence of product model systems, the need for legacy analysis tools to access the data from these systems also exists. Acquisition reform has intensified this need by its requirement for integrated product and process development.

This project involved two analysis programs that were to obtain their data from the Geometry Object Structure (GOBS) CORBA server. These analysis programs are the Ship Vulnerability Model (SVM) and the Ship Hydrostatics Characteristics Program (SHCP). For brevity, only SVM will be discussed. However, SHCP also involves the same issues that were addressed with SVM during the course of this project. This section of the paper will discuss how GOBS was used to obtain geometric data that was used to generate SVM ship structure data.

Product Model Definition Required By Ship Vulnerability Model (SVM)

Many levels of topological fidelity are feasible with GOBS. A robust level of fidelity for GOBS is needed to provide information necessary to construct input for legacy analysis tools. SVM needs information about the faces that are shared by compartments and whether faces have bulkheads behind them. Additionally, two major views of the ship geometry are needed to generate SVM ship structure data. These views are the ship decks and the ship compartments. The ship decks are needed to determine SVM deck levels and the ship compartments are needed to generate SVM compartments, SVM plates, and SVM points. A GOBS object with a robust level of fidelity provides this information.

GOBS CommonView objects were defined to provide the view information for SVM. The first major CommonView object represented all the decks of the ship. This CommonView object was composed of CommonView objects where each of these objects was a ship deck. Each CommonView object representing a deck was composed of TopologicalView objects. These TopologicalView objects described the geometry of the deck.

The second major CommonView represented a view of all compartments on the ship. This object was composed of CommonView objects where each of these objects was a ship compartment. Each of these CommonView objects was composed of TopologicalView objects that were the compartment faces. These TopologicalView objects also contained the material type for the face and the thickness of that material.

GOBS Interface Methods Used by Ship Vulnerability Model (SVM)

The GOBS server has many methods available for use by legacy analysis programs. These methods provide the means to extract topology and attribute information. These methods also allow for the evaluation of points on a surface and/or curve as well as other important functions needed by legacy analysis programs.

The interface to SVM uses many of the methods available from the GOBS server. For brevity, only a few will be discussed here. Methods are used to determine if surfaces and faces could be considered flat as well as if curves and curve segments could be considered straight. These methods are important in the algorithms used by the interface to SVM. Methods to extract the relationships between CoEdge objects and Edge objects are also used. These methods are important in understanding what faces are shared by compartments and what faces have bulkheads behind them. These methods allow the interface to insure that SVM plates were not duplicated.

Interface Implementation to Ship Vulnerability Model (SVM)

The Ship Description File (SDF) provides the input data necessary for SVM. BRIDGE is an existing program that edits an SDF file allowing data to be added, modified, deleted and displayed. SVM uses the SDF file created by BRIDGE and is run separately. The High-Level Architecture (HLA), which was developed by the Defense Modeling and Simulation Office (DMSO), defines a template for Simulation Object Models (SOM). A SOM was developed for the SDF. A CORBA server was then developed to implement the SDF SOM. Because of resource and time constraints, only the ship structure and vital component sections of the SOM were implemented. A SVM Builder CORBA client was developed to interact with the GOBS CORBA server and the SDF CORBA server. This client provided the vulnerability analyst with the capability to generate a SDF file for BRIDGE. Figure 9 depicts this implementation.

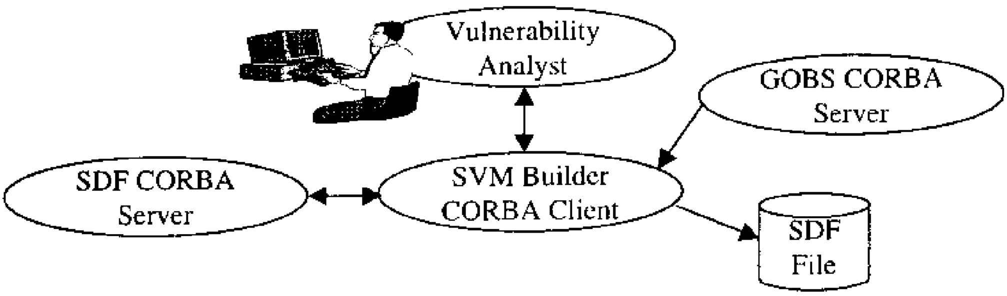


Figure 9 - Overall Structure for SVM Interface Implementation

The CORBA client aided the analyst in building the SVM points, SVM plates, and SVM compartments. This was generally accomplished by the following steps:

1. Retrieve CommonView object that contains all decks from the GOBS server.

2. Process this CommonView object to determine SVM deck levels.
3. Retrieve CommonView object that contains all compartments from the Gobs server.
4. Process compartments by constructing SVM plates for each face of each compartment.
5. Specify the SVM points that represent the corners of the each compartment.

The SDF server constructed SVM points, SVM plates, and SVM compartments given information from the client. When the SVM ship structure was completed, the client retrieved the SDF object from the SDF server and wrote it to the file specified by the vulnerability analyst. This file could then by processed by BRIDGE to add the additional information needed to execute SVM.

Conclusions

The primary objective of this project was to demonstrate the use of a neutral representation to provide data from a product model to legacy analysis programs used in ship design. In this case, an object server, rather than a simple flat file, provided the neutral model as a service to analysis. This allows the client programs to access the data for a ship as a single entity without having to create a single large transfer file from the product model. It also permits incremental update of the data without re-executing the entire transfer. Secondly, this effort tested the STEP standard as a means of loading this neutral representation from an external product model.

The GOBS server performed well as a neutral representation for the test cases attempted for this project. It provided the topology and geometry required to run SVM and SHCP. It should provide the data required for other analysis programs as well, since it was not tailored specifically to the above two programs but to general analysis program requirements. GOBS provides a set of objects that allow for the communication of the ship model as an intelligent, connected structure; along with methods to access and interrogate the geometry.

Although VIVID® was the only product model to actually load the GOBS server, the server architecture was developed independently from the VIVID® portion of the transfer and no changes were made to the format of this data to cater to the needs of the VIVID® product model. Any product model that can provide the geometry, topology, and required attributes could load this server.

This project demonstrated the use of STEP Part 42 entities. With a few exceptions, all data required by GOBS could be handled through Part 42 entities. STEP AP 216 (Molded Forms) and AP 215 (Ship Arrangements) were demonstrated as part of this project. As emerging standards, they were used to load some of the data into the VIVID® system and in its export.

The project also identified a few key areas for further development in data transfers between systems requiring this type of data. In several areas of the STEP schema definition, particularly in the topological entities, there are several ways to represent the same type of entity. The participants in this project had to agree to a specific definition in these areas. This is reminiscent of the "flavoring" required to transfer IGES data between unlike systems, but it was actually surprising how few of these problems were found in STEP. Further development of surface healing technologies and greater modeler tolerance for surface "anomalies" is required. Fortunately, this is an area that is being aggressively pursued.

APPLYING THE STEP SHIPBUILDING PROTOCOLS AS A BASIS FOR INTEGRATING EXISTING IN-HOUSE SHIP DESIGN APPLICATIONS

Stein Ove Erikstad, MARINTEK A/S
Dariusz E. Fathi, MARINTEK A/S
Trondheim, Norway

Abstract

In this paper we discuss how the STEP shipbuilding protocols can be used as a means to achieve a low-budget, product model oriented integration framework for a preliminary ship design system. Today, MARINTEK has several applications for the advanced analysis of the hydrodynamic performance of the ship, such as resistance, propulsion, seakeeping and manoeuvring. In the paper we describe how these applications can be brought together using a STEP-based product model for data integration, and a component-oriented, web-based "workbench" for the integration of presentation and user interaction.

Introduction

MARINTEK has a long tradition of developing advanced methods for the analysis of ship resistance, propulsion, manoeuvring and seakeeping behaviour. To make these methods available to the industry, they are typically packaged as software tools that can be used in the design process

Until now, these tools have been developed as standalone applications, each having an independent user interface, and an independent representation model of the relevant parts of the ship. Some of the programs have graphical user interfaces, while others are batch-oriented. Figure 1 shows an overview of some MARINTEK programs related to ship hydrodynamics/hydrostatics and their need for input data. A number of translators exist to allow for some degree of integration through file-based data exchange and to some extent, the ship hull geometry files are the same. However, this integration mechanism is not satisfactory – it is difficult to implement and maintain, it requires a lot of resources when used, and it doesn't offer the level of seamless integration that is required by the iterative process in the early stages of ship design.

As can be seen from Figure 1, many of the programs need very similar input, in particular related to the loading condition information and the ship hull geometry. Regarding the ship hull geometry, the hydrodynamic codes in the figure are either based on strip theory, which basically needs description of ship sections, or panel methods, which need a 3-dimensional description of the ship hull by means of flat panels.

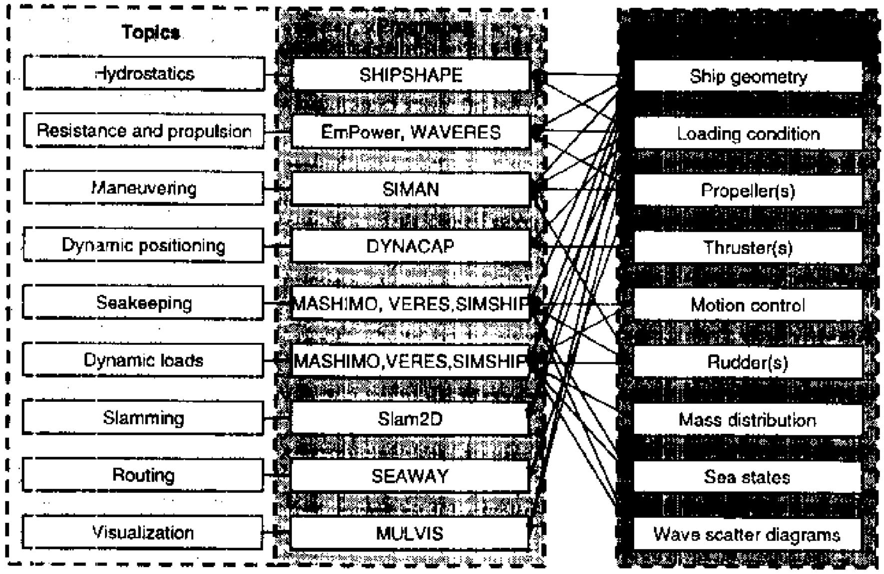


Figure 1: Some MARINTEK programs for analysis of ships and the required input data for the different applications [1].

The motivation for a stronger integration of these codes by means of a common representation model can be summarised into the following points:

- From the developer's point of view, developing new programs very often requires much of the same input as existing programs, and much of the programming effort is spent re-using old routines for reading geometry files as well as internal representation of the ship hull. Typical derived quantities, such as the ship's displacement, wetted surface etc. are also calculated in each program or given manually as input. Hence, there is clearly a benefit by means of programming time and error sources by accessing the input data through a common representation model and common program components. This gives the developer (which in our case are mainly hydrodynamicists) more time to concentrate on the more interesting parts of the computer code, i.e. solving the actual hydrodynamic problems rather than spending time on modelling.
- From the user's point of view, there is also a clear benefit of having a common representation of the ship: The data needs only to be given once, and time spent with conversion of data between different programs, as well as the possible error sources of keeping the different input files synchronised and up-to-date can be minimised.
- In addition to the clear benefits of having a common representation of the ship and other input data, there is also a clear benefit of re-using graphical user interfaces and giving them the same familiar look and feel. For the developer, this means that only the user interfaces needed for input which are not yet implemented in the common framework need to be programmed. In addition, there will be a defined framework for developing new components. For the user, this will mean less time spent learning to use new programs, as the user interface is essentially the same.

As a consequence, MARINTEK has undertaken an effort to develop a framework, called ShipX, which will bring both current and future analysis tools together into a single, integrated design environment

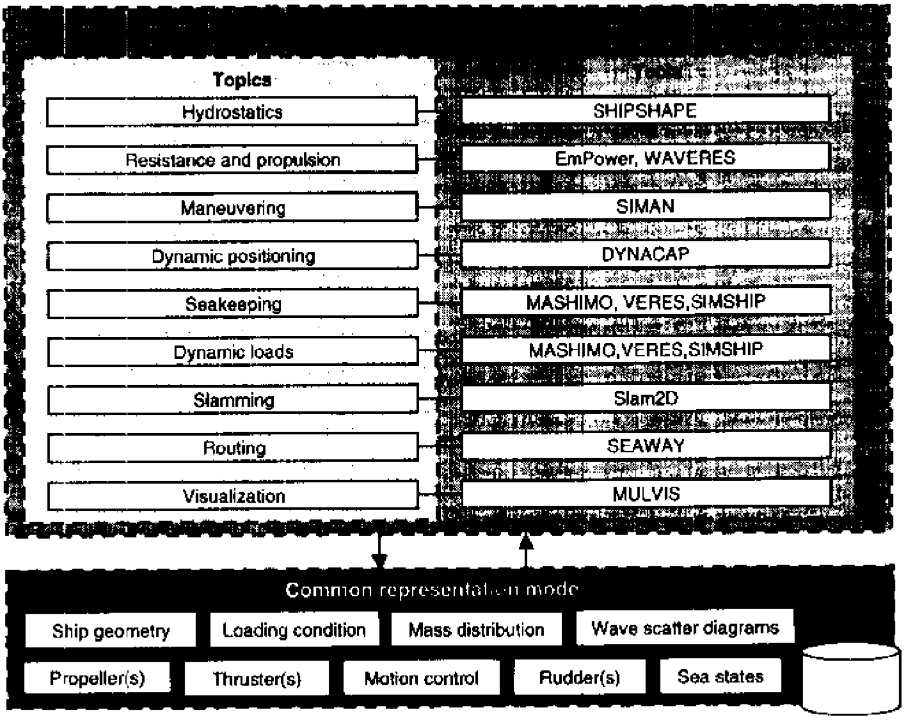


Figure 2: A possible way ahead?

What we wanted to achieve

In the development of the ShipX framework, we initially stated a set of design goals that should be guidelines in the development of the system architecture. These can be summarised as follows:

Simplicity

Due to limited availability of time and resources to develop an initial solution, the application framework to be implemented should be simple and cost effective. This ruled out the alternative of developing a complete, comprehensive in-house product model from scratch – an alternative that has become increasingly more common in large corporations today. Instead, we chose to go for a solution that to a large extent re-used existing technology and used off-the shelf software components.

Flexibility and Extensibility

In the first version, the analysis programs that we want to integrate are mainly related to the determination of the hydrodynamic characteristics of the ship. Thus, we have also limited the scope of the representation model to focus on the ship moulded form representation.

However, in later versions we want to have flexibility with respect to extending the functional scope, covering such areas as global strength analysis, preliminary steel weight estimation, cost estimation, and intact and damage stability. This requires that we at an early stage take into consideration both how the representation model can be extended, and how new analysis modules can be "plugged into" the existing framework.

Weak bindings to platform and programming language

Traditionally, most analysis software developed by MARINTEK has been – and still is – programmed in FORTRAN. Taking into consideration coding efficiency, execution speed, and in-house programming experience, FORTRAN is still a viable alternative. In later years, utilising other programming languages such as Visual Basic, Java, and C++ to create graphical user interfaces have been used more and more common in MARINTEK. In ShipX we need to take this into consideration, allowing components developed in different languages to co-exist within the same framework.

ShipX will primarily be developed for the WinNT platform. Still, it would be advantageous to have a high degree of platform neutrality, allowing the use of existing MARINTEK software packages developed for the Unix platform. This is most important for the server part of the framework.

Design Principles

Taking the above mentioned goals into account, we have applied the following design principles for ShipX:

Data integration using a product model

As previously mentioned, the existing applications we want to integrate into the ShipX framework are each having a separate representation of the hull form that is customised for the specific needs of the application. These different representations are to a large extent overlapping in scope. Still, in order to exchange data today in a typical iterative design process, translator software needs to be used extensively.

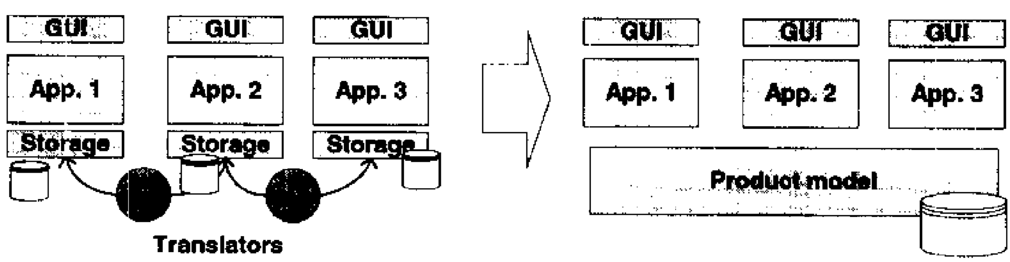


Figure 3: From separate representation and storage to data integration using a product model

In order to achieve the necessary level of integration, the different applications need to share common data through a single representation model and corresponding storage service. For this we want to use a product model that is based on the shipbuilding protocols in STEP, as illustrated in Figure 3. The relevant parts of the STEP model will be translated into a class library and made available to the client applications through the ShipX Information Services layer. For persistent storage, we use an object-oriented database (ObjectStore [2]) that integrates tightly with the other parts of the information service.

Presentation and control integration using components and web technology

For the end users to perceive the different applications in ShipX as a single, integrated design environment, the interfaces for user interaction and presentation must also be given a uniform look and feel. We want to achieve this by using *components* as the standard building block in the user interface. These components will be developed either by wrapping up existing applications as single components, or by dividing existing applications into pieces. In addition, we plan to develop a number of new components that will provide functionality common to the existing applications.

These components are brought together in a container application – or "workbench" - to form a complete and consistent user interface (Figure 4). For this we want to use a standard web browser, for the following reasons:

- The browser itself is free, widely distributed, multi-platform and in constant development
- The browser is capable of hosting a wide range of component technologies
- The combination of mark-up languages (HTML, DHTML, XML), a standardised programming model (Document Object Model – DOM) and scripting allows for cost effective development and configuration of the user interface, and enables the implementation of dynamic behaviour and work process support.
- For future versions of ShipX it opens up the possibility of distributing updates and extensions using the Internet.

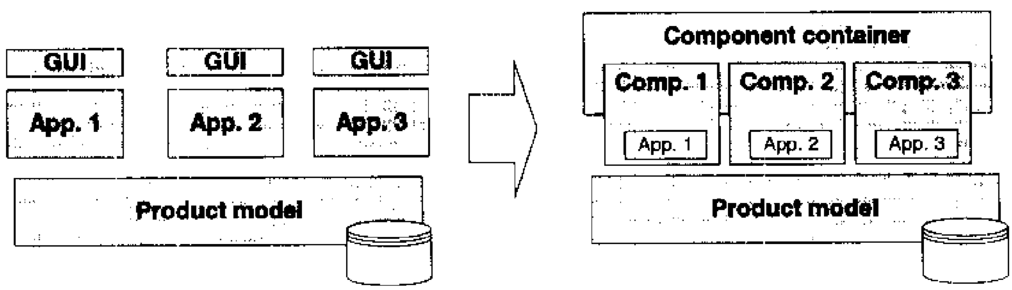


Figure 4: Presentation and control integration through components and web technology

Layered architecture

ShipX is divided into a set of clearly separated layers. Each layer has a well-defined set of responsibilities, which is exposed to adjacent layers via a limited number of interfaces, thus encapsulating the internal implementation. This layered architecture – which seems to have become a standard in information systems development – reduces the dependencies between the different parts of the system, thus supporting flexibility and extensibility. The main layers and corresponding responsibilities in ShipX are illustrated in Figure 5.

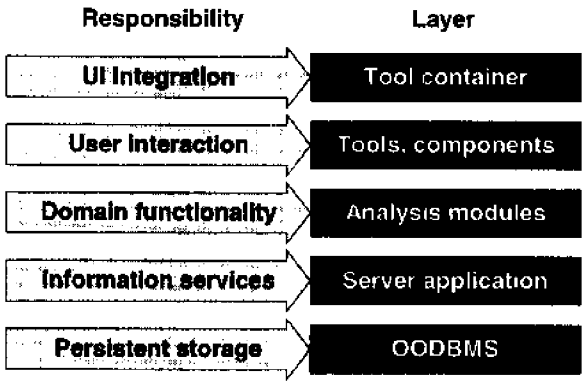


Figure 5: The different layers in ShipX and their responsibilities

Why Use the STEP Shipbuilding Protocols?

Our main motivation for using the STEP shipbuilding protocols as a basis for our representation model was the limit on available time and resources. The alternative solution of developing an in-house product model from scratch would in itself require more resources than we had available. Thus, to a certain degree we were forced to re-use an existing model.

Until now, the main focus of the marine industry has been on applying the STEP shipbuilding protocols for the *exchange* of information between different systems. However, as it is stated in the introduction to each of the protocols, they are suitable *"not only for neutral file exchange, but also as a basis for implementing and sharing product databases and archiving"* [3].

For us, it was the suitability for implementing a product database that was of interest, and whether this offered an appropriate common representation model for ShipX. Our experiences so far indicate that this was a sound decision, and we can summarise the main benefits in the following points:

- *The STEP models incorporate extensive domain knowledge and experience.* This comes as a result of the shipbuilding community spending substantial resources over many years developing these models, involving a large number of domain experts in the process. The models have from the very beginning been designed to be a common representation across the native, system-specific representation. This makes them suitable for being a basis for integration of existing applications in ShipX. To a certain degree they have also been tested in real-life scenarios through the EMSA business cases [4], with promising results.
- *The scope of the STEP models covers a large part of the shipbuilding domain.* This means that even though the initial focus in ShipX will be on the ship

moulded forms representation, we can extend the scope in later versions by integrating other applications within the ShipX framework.

- *The STEP models are consistent across the different application protocols.* This is achieved by having a common meta-model – the Ship Common Model – that serves as a basis for all ISO-STEP Ship Application Protocols. Consequently we can extend the ShipX representation model in later versions without having to do major restructuring of the first phase implementation.
- *A representation model based on STEP makes information exchange with other systems easier to accomplish.* Using the STEP protocols as the underlying representation model in ShipX, it means that STEP-based information exchange with other systems only implies a one-to-one mapping from ShipX's internal representation to a neutral STEP format.

Handling Complexity Using Model Facades

Based on the aforementioned arguments, our general impression is that the STEP Ship Application Protocols offer a suitable and cost-effective starting point for an in-house product model, due to qualities such as consistency, extensibility, standardisation and completeness. However, such qualities seem to have a corresponding cost.

Our initial assumption was that the client applications should have complete knowledge of the common representation model, and that they would retrieve the data required for the analysis tasks by traversing the model object by object, accessing the values of relevant fields along the way. However, this approach turned out to be less attractive when we started the integration of the client applications – from their point-of-view, the STEP-based representation model was too complex to provide an efficient interface to persistent data. This complexity can be characterised as follows:

- *Model fragmentation.* The STEP models consist of a large number of relatively small objects, each having a fairly limited number of fields.
- *Deep inheritance hierarchies.* In order to ensure a coherent modelling approach in the different shipbuilding protocols, each protocol is based on a common foundation consisting of a set of abstract, general modelling concepts in the Ship Common Model. From this framework, the more concrete entities in the different application protocols are derived.
- *Large size of model.* This characteristic comes partly as a consequence of the two others: Many small objects and many levels of abstraction tend to make the model large.

Thus, in order to gain access to the information necessary to perform a certain function, the client application will typically have to traverse a long chain of objects. This tends to make the implementation of client applications more difficult and error prone, and the large number of objects that has to be accessed turns into a performance overhead.

A solution to this is to use a model façade, as illustrated in Figure 6. The façade design pattern provides a unified, high-level interface to a subsystem, making it easier to use for clients. It also helps reducing the dependencies between the subsystem and its clients, promoting portability, flexibility and independence of the subsystem. In ShipX, the façade pattern implementation has the following advantages:

- It hides the complexity of the STEP-based representation model from the clients, offering instead a simpler interface that is specifically designed to serve the particular needs of the known client applications.
- It will contribute to correct usage and consistency of the model by encapsulating common functionality.
- It makes it possible to change and extend the representation model (say, due to new revisions of the STEP AP's) without affecting client applications. Only the façade implementation (on the server side) needs to be changed or extended.

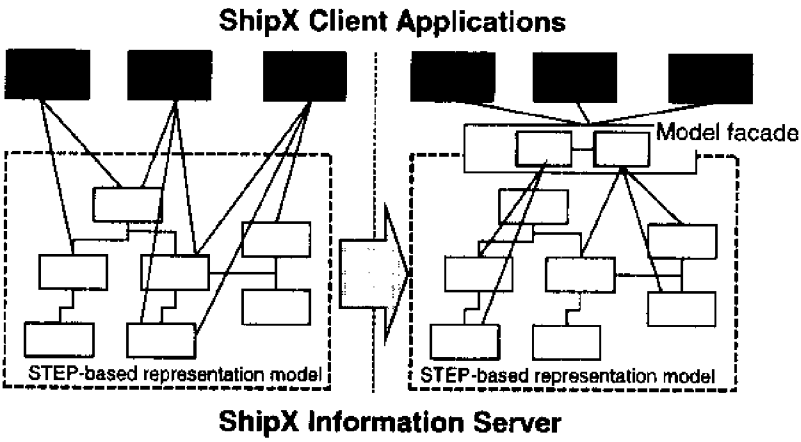


Figure 6: Applying a façade design pattern some of the complexity of the STEP-based representation model is hidden from the client applications

In addition to this, the model façade also serve a more strategically important purpose in ShipX. In the first version of ShipX, the information server will primarily serve "dumb" objects according to the common representation. Most domain behaviour (business rules and methods) will be contained within the client applications.

However, in the longer term it is our goal to be able to share more of the domain behaviour that is common between the different applications. This can be achieved by "migrating" this behaviour from the client layer to the server layer. For this purpose the façade model seems to be a more appropriate host than the STEP-based representation model, in terms of object granularity and abstraction level.

Implementation Considerations

An overview of the technologies chosen for the different layers in the ShipX implementation is illustrated in Figure 7.

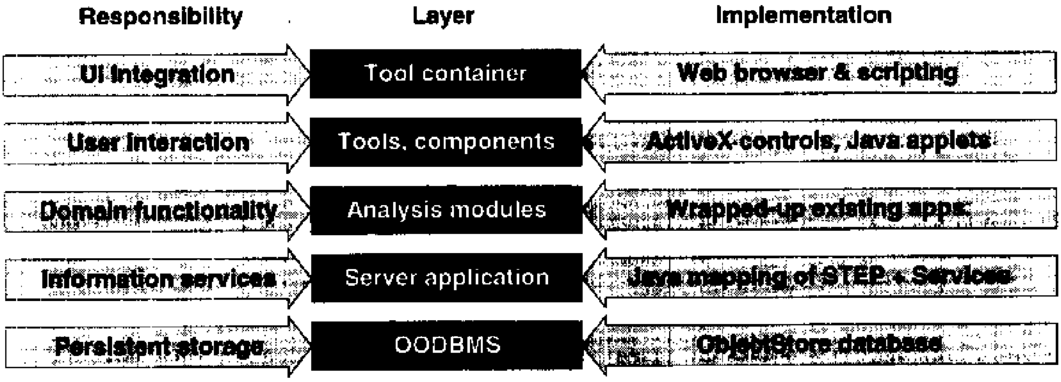


Figure 7: The main layers of the ShipX architecture, with corresponding responsibilities and implementation technology

Mapping the STEP model into a Java-based Information Server

The implementation of the representation model can broadly be summarised through the following steps:

- The relevant parts of the EXPRESS schemas were mapped one-to-one into a corresponding UML model using Rational Rose [5]. The different entities were made part of UML packages corresponding to the EMSA Building Blocks in order to maintain a perceivable span of control.
- From the UML model, a corresponding Java class library with default implementations of the basic model behaviour was produced, using the code generation facilities of Rational Rose.
- For persistent storage, we chose an object-oriented database, ObjectStore PSE Pro for Java. This database solution integrates the persistence mechanism seamlessly into the Information Server – any Java object that is created within a transaction and referenced by another persistent object will be automatically made persistent, without having to make any explicit arrangement for this object in the database up-front.
- In the development of the façade model, we initially captured the information requirements from the client applications through a set of UML use-cases. From these, a limited number of central domain objects were derived with corresponding methods. These façade objects will never be persisted themselves – they serve only as "wrapper" objects encapsulating a part of a complex underlying representation model.
- Further, a set of services was implemented as a thin layer above the ObjectStore database, encapsulating some of the more complex aspects of database management.

Our general impression is that implementing the STEP-based representation model has proved to be an achievable endeavour even within the limited resources. An alternative solution could have been to use a third party STEP toolkit that could make available a database solution directly based on the EXPRESS schemas of the relevant application protocols. However, while this is an obvious choice for a large-scale system, it is not compatible with our requirements for a lightweight solution that can be developed and distributed at a limited cost.

Closing Remarks

The key question we have tried to answer in this paper is whether the STEP shipbuilding protocols can be used as a means to achieve a low-budget, product model oriented integration framework for a preliminary ship design tool.

Our experiences so far indicate that the answer is "yes". We avoid the initial spending of large resources in developing a common representation model, relying instead on the assumption that the STEP models are sufficiently expressive, complete, consistent and effective. This approach makes it possible to maintain the focus on the development of tools and model behaviour related to marine technology and hydrodynamics.

However, we see the need to encapsulate some of the inherent complexity of these models, in order to make the client application's access to data more efficient. This is achieved through a model façade, which offers a level and scope of model interaction that is specifically designed for the known needs of the existing applications we wanted to integrate. It is here our main focus will be in the future development of ShipX, by the migration of domain knowledge from today's client applications.

Acknowledgements

This work has been conducted as a part of the strategic research project SKIPRO 2001 which is a five year research program which started in 1997 with the main focus on resistance and propulsion of ships. SKIPRO 2001 is founded through the Norwegian Research Council and the marine industry.

References

1. MARINTEK Web Pages, www.marintek.sintef.no
2. Object Design Inc., www.odi.com
3. ISO TC184/SC4/WG3/N563, Working Draft, "Product Data Representation and Exchange – Part 218: Ship Structures", Hamburg, 1996
4. European Marine STEP Association, www.oss.dk/emsa/
5. Rational Software Corp., www.rational.com
6. Gamma, Erich, Richard Helm, Ralph Johnson, John Vlissides (1994), "Design Patterns: Elements of Reusable Object-Oriented Software", Reading Massachusetts, Addison-Wesley

The VIRGINIA Class Data Transfer/Sharing Evolution

Gregory Morea
Engineering Supervisor
Electric Boat Corporation

Introduction

The late 1980's were a time of great optimism for the submarine construction community, particularly for Electric Boat Division (EB) of General Dynamics Corporation. The TRIDENT ballistic missile submarine construction program was in full swing with one ship per year being built solely by EB, SSN 688 attack submarines were being built at a total rate of four per year by both EB and its rival, Newport News Shipbuilding (NNS), and both shipyards were continuously busy with this work. In the planning and design stages was the Navy's newest attack submarine, the SEAWOLF. Planned to be the successor to the long running SSN 688 program, the SEAWOLF was to be a 30 ship class, designed jointly by both EB and NNS, incorporating the latest technologies in design, construction, and mission performance. It was projected that both EB and NNS would compete on construction contracts, assuring both a favorable price to the Navy and stable employment at both shipyards.

This vision was not to last into the 1990's, however, as world events dictated severe changes to the Navy's shipbuilding program. The collapse of the Soviet block lessened the Navy's need for an ambitious new submarine construction program. In response, the Navy cut the number of TRIDENTs needed from 24 to 18, a 25% reduction. More severely, in 1991, President Bush cancelled almost the entire SEAWOLF program; leaving EB with little more than the lead ship to finish and perhaps go out of business. Subsequent negotiations over the years have added two more SEAWOLFS to the construction program, but the challenge was on for EB to figure out how to maintain a submarine design and construction capability with a fraction of the business of previous years. It was in this environment that the New Attack Submarine (NSSN) was first conceived. If one wishes to examine the data transfers required for NSSN, however, we must first look at the work in this area done to support SEAWOLF.

SEAWOLF, The Basis for Electronic Data Transfer Established

When the Navy embarked on the design of the SEAWOLF, it put into place an arrangement never seen before in Navy shipbuilding, the idea of two shipyards concurrently carrying out the design, with the explicit understanding that construction could be carried out at either yard. Figure 1 shows the design split of the SEAWOLF. To improve efficiencies with the flow of data between the two design agents and the construction yard(s), the Navy funded and participated in an extensive digital data exchange development effort. The focus of the effort was to develop and implement methods to electronically transfer design and construction information, in selected areas, to minimize the amount of manual re-entry of data throughout the design/construction process. The selected areas were a) non-processable text (word processing), b) processable text (database), c) drawings, and d) product models. Working groups were formed with participation from the Navy, EB and NNS personnel, and over a multi-year period, each of these areas had processes developed and implemented to transfer data.

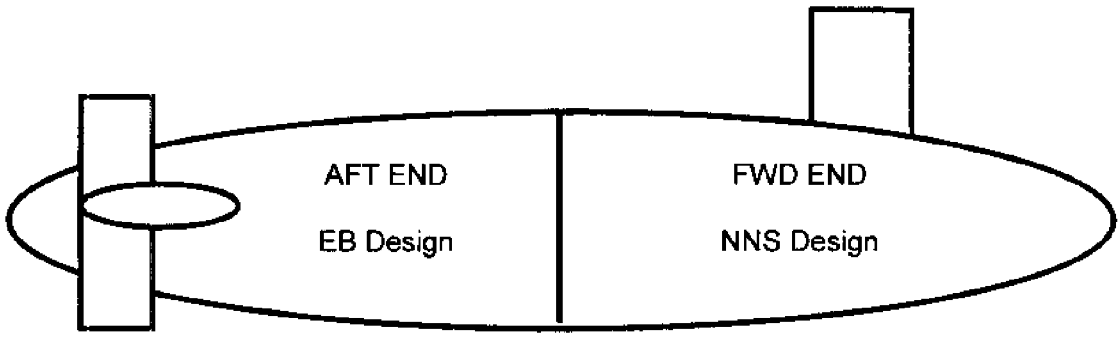


Figure 1, Design Split of the SEAWOLF

The working group concentrating on word processing largely focused on procedures to share documents between the Navy, EB and NNS. A list of acceptable word processors, e.g., Wordperfect, Word, Wang, was agreed to, and procedures for translating between formats was documented. There was very little technical development to be accomplished as commercially available products were utilized.

The database working group faced a more formidable challenge. To design and construct a submarine, numerous databases had to be maintained and transferred between the involved parties. Included were files such as the Engineering Parts List (EPL), Master Drawing Schedule (MDS), Controlled Pipe Joint List (CPJ), and the like. Each of these databases resided at either EB or NNS, and they were written in proprietary formats. To enable the transfer of these databases, the working group established a Data Element Dictionary (DED), which contained the layout of each record of each involved database which was used to format transfer files. The databases could then be set up and used in any manner at both shipyards, and the DED record layout would govern the transfers. This process is still in day to day production use.

The drawing working group was chartered with the task of developing a way to electronically transfer two-dimensional drawings between the Computervision (CV) system used by EB and the CADAM system used by NNS. The assumption behind this transfer was that shipyard specific information could be added by each yard in their native system and then publish the drawing for construction use. Figure 2 shows this process. The mechanism chosen for this transfer was the Initial Graphics Exchange Standard (IGES). IGES is a neutral file transfer mechanism designed specifically for the transfer of information between dissimilar CAD systems, and it is an ANSI National

Standard. During the development of this task, the working group participants became heavily involved in the development and maintenance of IGES itself, a relationship that continues today Although the working group developed a production capability for drawing exchange, this capability was seldom used in production due to the ultimate selection of a single shipbuilder and the extensive use of hard copy drawings by the shipbuilder.

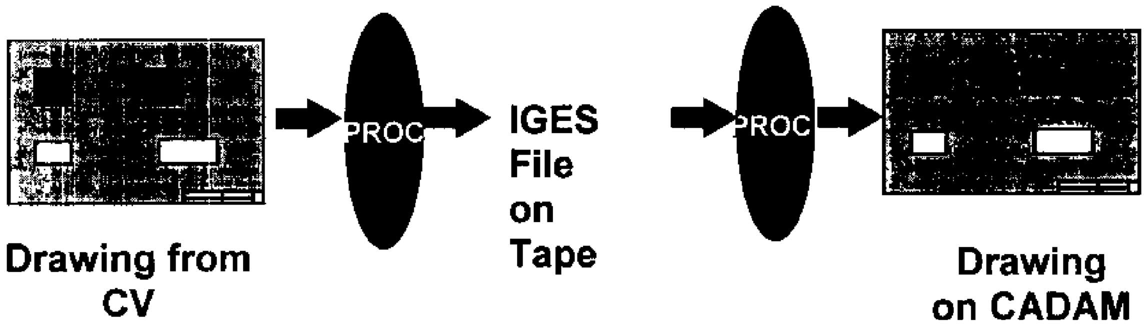


Figure 2, IGES Exchange Schematic

While the drawing exchange was not extensively used in production, the final area of working group development, product models, was and is. One must be careful here to realize that the use of the term "product model" in the context of SEAWOLF refers to three-dimensional models of pipe and ship's structure that would be used in manufacturing. The structural models are three-dimensional wire-frame models (Figure 3) of plates and shapes that the shipbuilder receives from the design agents to begin the plate lofting process. The models are broken down into two-dimensional templates that guide plate cutting operations. Again, IGES is used as the transfer mechanism, and the same CV and CADAM systems are involved.

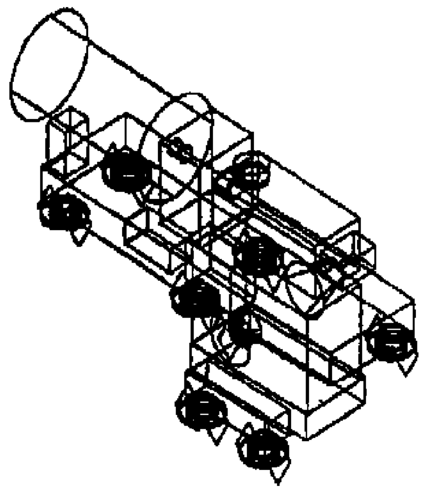


Figure 3, Typical Wire-Frame Model

In a similar manner, piping fabrication models are exchanged between EB and NNS. These models contain pipe centerline, pipe attribute, and component origin, orientation and part number information. This information is used by the shipbuilder to feed automatic pipe fabrication programs without manual re-entry of the data. IGES is used here as the transfer mechanism, but it is specifically tailored to pipe transfers. Thus, entities such as "pipe flow path" are found in IGES files to support this transfer, and the processors to generate the files were custom written for both the EB and NNS systems. The specific file format for this exchange was made available to industry as the "3D Piping IGES Application Protocol", available as a companion to the IGES standard.

The significant aspect to all of the efforts described above is that they were all designed and implemented to *electronically transfer* data from the design agent to shipbuilder to increase efficiency in the construction process. The Navy, EB and NNS were all keen to realize that these transfers could and did alter the manner in which a shipbuilder receives data from a design agent. In today's environment, many of these types of transfer are taken for granted, but they had their start in the SEAWOLF program.

SEAWOLF Lessons Carried to NSSN, the Beginnings of an Integrated Environment
As construction began on the SEAWOLF, the Navy and EB recognized the need to go further in improving efficiencies in communicating between design agent and shipbuilder. In fact, for EB it became a matter of survival. With the termination of the SEAWOLF program, EB management in the early 1990's realized that the entire process of designing and constructing submarines had to be overhauled with significant cost

driven out of the process for the Navy to continue to procure submarines, and thus EB stay in business. This led to the formation of a corporate-wide initiative to re-conceive the entire design/construction process within EB, PADP, or the Production Automated Design Process. A detailed description of PADP is the subject for another paper, but suffice it to say that one of the key areas examined was the *movement of data* within the organization.

One of the ways in which control was put over movement of data was the selection of one CAD environment for the majority of the company's design work. After studying several alternatives, EB concluded that CATIA would be its CAD system of choice for design work in the future. At this point, it is crucial to understand that the environment that was implemented at Electric Boat included a number of products under the umbrella "CATIA." To store and effectively manage the design data, the CATIA Data Manager (CDM) was selected. As the power of visualization was clearly understood, EB designed and constructed several rooms where large scale walkthroughs could be conducted, and links were custom developed between CATIA and the commercial visualization tool, IGRIP. Finally, links were established between CDM and the many material, standard parts, and related databases required to design and construct a submarine. Figure 4 shows this environment.

Of more importance than the actual choice for CAD system, EB realized that a new design/construction *process* had to be implemented to successfully design and build any successor to SEAWOLF. To accomplish this process implementation, EB evolved the PADP effort into a program designated as 90/10; 90% capability with 10% of the risk. The charter of this program was to develop a process that would be state of the market without incurring undo risk due to reliance on emerging technologies. Since the start of 90/10 in the early 1990s, this philosophy of achieving state of the market has allowed EB to create an environment where design can be accomplished more efficiently than was done for previous classes of submarines. One might ask, "why not state of the art?" Under the 90/10 philosophy, it is felt that the risks inherent with state of the art outweigh the benefits of early implementations of technologies. Figure 5 is a high-level view of this process.

With both a process and comprehensive array of design tools in place, EB was able to negotiate a unique arrangement with the Navy to both design and build the successor to SEAWOLF, the New Attack Submarine, or NSSN. Unlike all earlier submarine programs where separate contracts were awarded for design and construction, the NSSN was to be both designed and built under a single contract at Electric Boat. This would allow both design and construction efforts to maximize efficiencies at EB by tightly coupling. In this way, the Navy and EB felt that the best value could ultimately be delivered to the Navy.

This tight coupling of design and construction activities took place in many areas, and it utilized many new technologies brought to EB by the CATIA environment. Key to this coupling was the involvement of the shipbuilder much earlier in the design process.

ELECTRIC BOAT'S SYSTEM ARCHITECTURE OVERVIEW

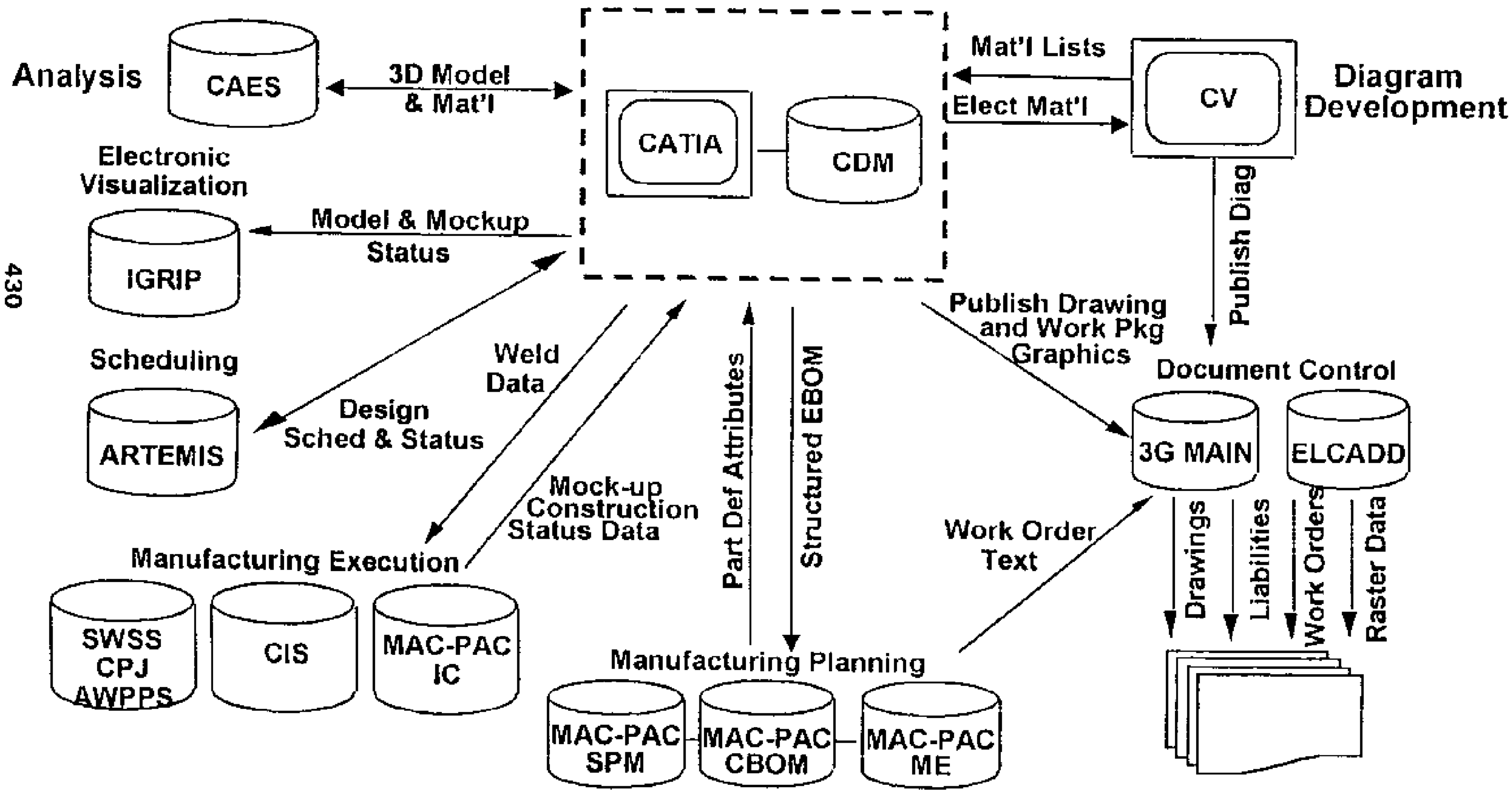


Figure 4

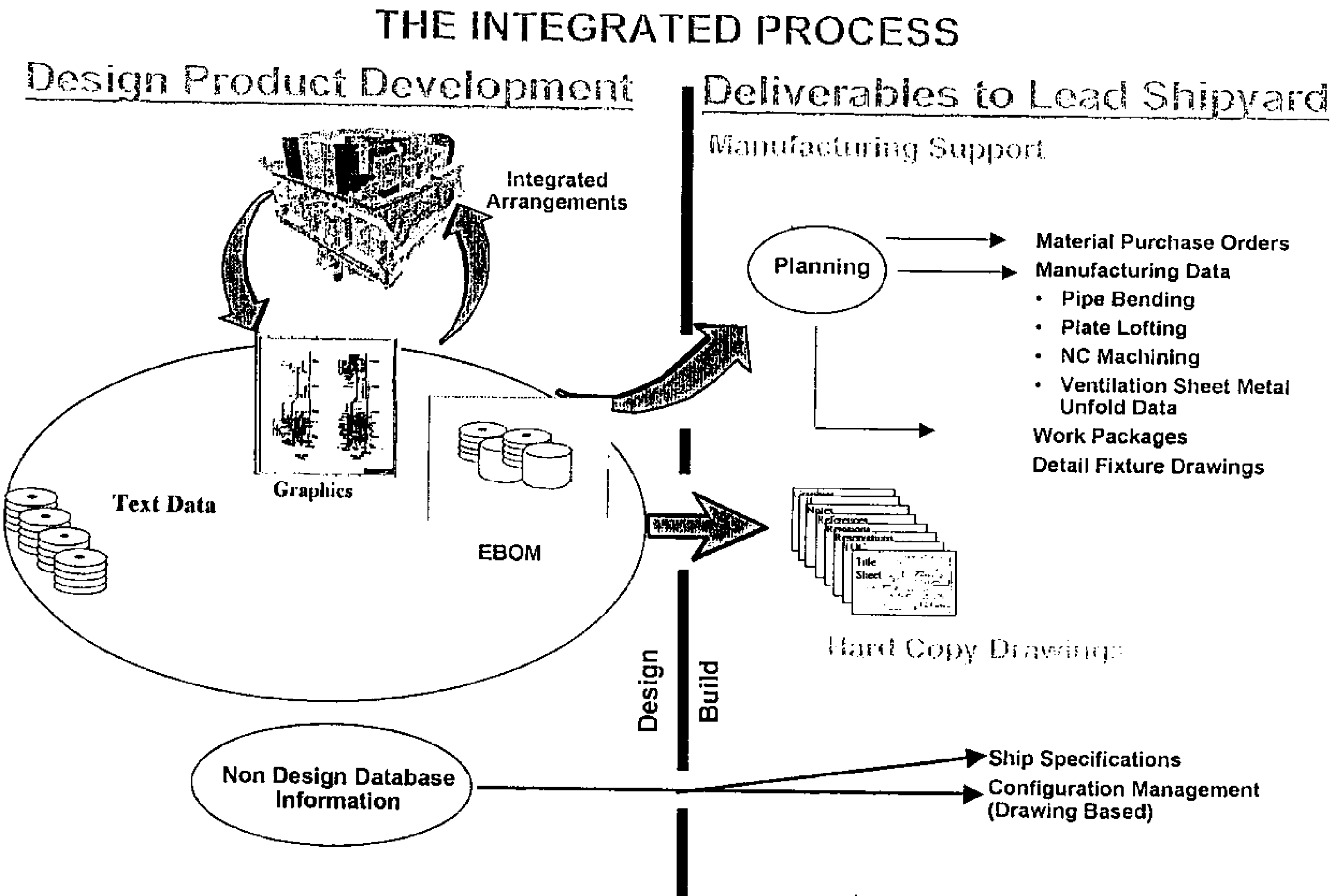

THE INTEGRATED PROCESS
Design Product Development
Deliverables to Lead Shipyard
Manufacturing Support
Integrated Arrangements
Planning
Material Purchase Orders
Manufacturing Data
• Pipe Bending
• Plate Lofting
• NC Machining
• Ventilation Sheet Metal Unfold Data
Work Packages
Detail Fixture Drawings
Graphics
Text Data
EBOM
Hard Copy Drawings
Design
Build
Non Design Database Information
Ship Specifications
Configuration Management (Drawing Based)

Figure 5

Rather than being given a stack of drawings and told to build from these, shipbuilder representatives sat on design/build teams which planned, arranged and designed major areas of NSSN. Using the visualization tools available, these teams were able to not only make sure each area met its mission requirements, but they were also able to make sure that the area was buildable. Interference checking was done during visualizations, not on the boat during construction. Suggestions made by the design/build teams on design changes were then fed back to the design baseline, which was being developed as solid models in CATIA and stored in CDM. This process, combined with computer links between design and manufacturing databases, was allowing the shipyard unprecedented abilities to plan for NSSN construction.

It is important to realize that this entire program was taking place at EB, with input from the Navy. The NSSN program was geared, down to the smallest detail, to be designed and fabricated by one organization. A decision made by Congress in 1995 was to change this program significantly, for in that year, Newport News Shipbuilding was directed to participate in the NSSN program.

A Second Shipbuilder is Brought into the NSSN Environment

For over one hundred years, Newport News Shipbuilding (NNS) was actively involved in the design and construction of both commercial and Navy ships. Unlike EB, NNS builds both surface ships and submarines. In the early 1990's, however, NNS found itself the victim of the same market forces that were effecting EB, a declining Navy workload and fewer prospects for new Navy work. In addition, with the cancellation of the SEAWOLF program, and no NSSN successor program in sight, NNS and the Navy were realizing that this shipyard would soon lose its submarine construction capability. For the Navy, this would mean having only one shipyard capable of building submarines, EB. Realizing this, Congress passed legislation mandating that NNS participate in the NSSN program as a follow shipbuilder.

Putting business concerns and the natural rivalry between EB and NNS aside, the challenges faced by both shipyards and the Navy in implementing this new direction to the program were enormous. As was discussed above, allowing EB design and construction to tightly couple their processes and data flows, thereby taking advantage of the specific capabilities of EB, drove the cost efficiencies of NSSN. Now, NNS had to be given sufficient data to build NSSN, without introducing undue cost or schedule impact to the overall program. To begin to figure out how to get NNS the right data to construct NSSN, the involved parties, EB, NNS, and the Navy turned to the lessons learned from the last submarine data transfer program, SEAWOLF, as a starting point.

The SEAWOLF digital data exchange development was carried out in a number of working groups, discussed above. Accordingly, a number of working groups were set up with the charter of determining the most efficient ways of *transferring* data from EB to NNS for ship construction. Groups were formed for 2D Data (drawings, piping data), Textual Databases, and Engineering Reports (ERs – a method of design change control). As with SEAWOLF, the groups had members from each shipyard and the Navy. Each had a leader, met regularly, and the groups all reported back to a Steering Committee made up of EB, NNS and Navy management. Timetables were given for the working groups to complete their work, and the groups met throughout 1995 and into 1996.

While the model for the working groups was based on SEAWOLF, the technologies and end products involved were considerably more advanced. As an example, it was determined that drawings should be transferred two ways, raster and vector. Raster is akin to faxing a document, pictorial accuracy is maintained but changes are not easily accomplished. Vector drawings can be readily changed, but may not be pictorially accurate. For NSSN, it was determined that all drawings would be transferred via a raster exchange, and those that needed subsequent updating by NNS would be transferred via a vector method, IGES.

Another area that was investigated was HVAC duct information. It was determined that since both EB and NNS fabricated these from standard shape parameters, these parameters could be transferred as a series of ASCII tables which would feed sheet metal fabrication processes at NNS. Likewise, and similar to SEAWOLF, a piping transfer was planned around the specially developed IGES processors for piping information.

As EB was having impressive results with visualization on the NSSN project, the working groups planned for a transfer of visualization data to NNS. NNS was to procure an IGRIP system similar to EB's, and EB was to transfer the entire baseline of NSSN models to NNS on a regular basis. NNS would be able to use this baseline for construction planning. Figure 6 shows this visualization transfer.

As mentioned above, the transfer of Engineering Reports was the subject of an entire working group. During the construction of SEAWOLF, it was discovered that timely transfer and receipt of engineering changes was critical to keeping to the construction schedule. For SEAWOLF the transfer evolved into faxes and hardcopy mailings. With this knowledge in hand, the ER group for NSSN planned for transfers starting with faxes and progressing to direct database queries over time. The key to the NSSN transfers was that strong configuration management would be built into the process from the start.

As we have discussed what was *included* under the charters of the working groups, it is equally important to discuss what was *excluded* from the discussions of the groups. This can be summarized into two closely related areas: structural lofting data and 3D CATIA models. Early in the working group development stages, the idea of transferring 2D templates of lofted plates was considered. Due to the amount of post processing of the templates required by NNS, this idea was not pursued further. Because EB considered the 3D CATIA models of the ship's structure company proprietary, the transfer of these models was not considered. Thus, it was decided that NNS would do their structural lofting based on the drawings that they would receive.

Within each of the working groups the plan was first for agreement to be reached on an area of transfer. Any necessary software would be developed and then the process tested.

VISUALIZATION TRANSFERS BETWEEN EB AND NNS

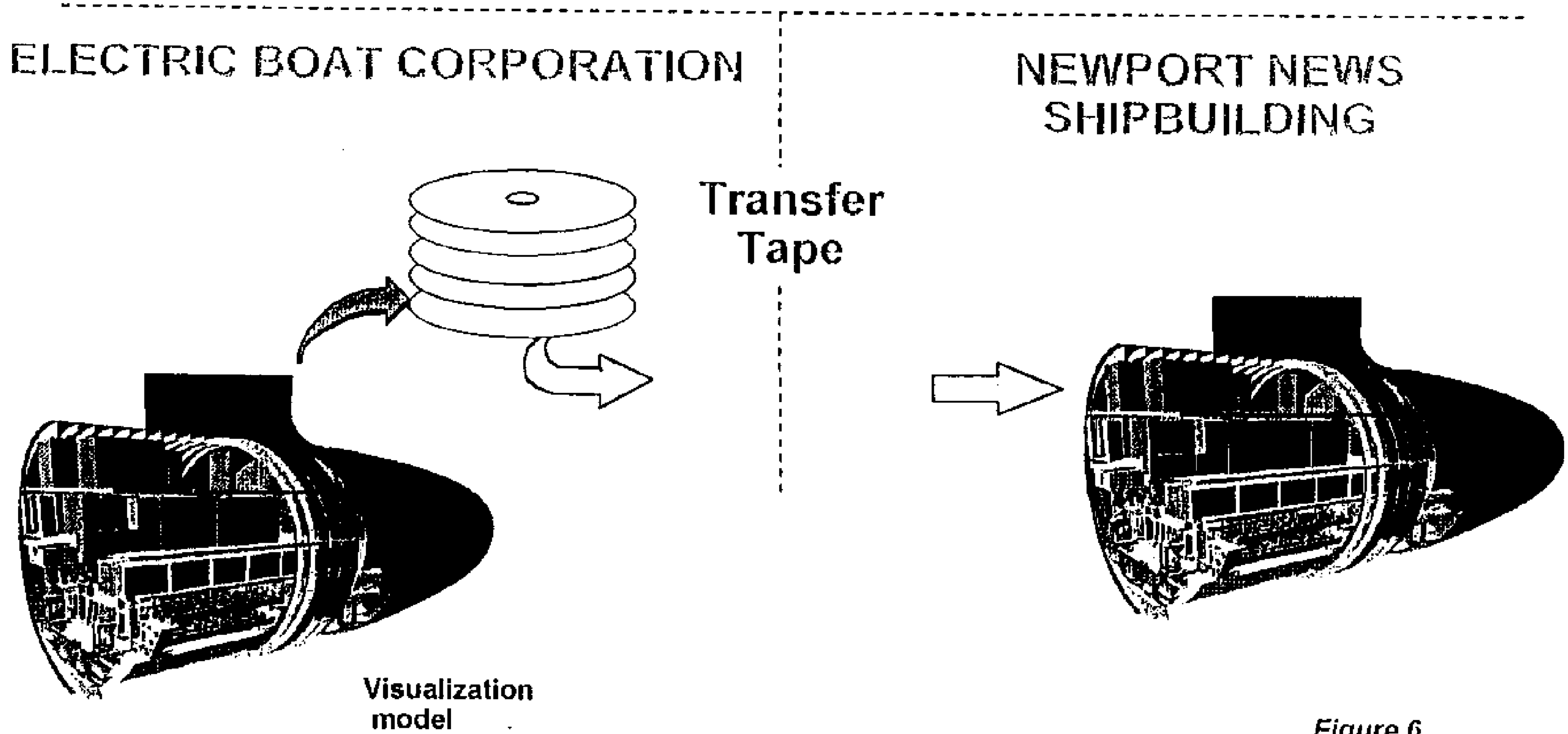


Figure 6

After the process of transfer was documented in procedures, the transfer would be placed into production. Data then could be sent from EB to NNS for ship construction. In some areas, it would be necessary for data to be transferred from NNS to EB as EB was design agent. Plans were made for these transfers as well.

From the above descriptions of the actions of the working groups, it is clear that the intent was to transfer data from EB to NNS, without integrating computer systems or processes. Although transfers would take place using wire links between the two shipyards, neither party would be utilizing the other's electronic data environment. Thus, NNS would function in the traditional role of follow shipbuilder; data would be provided by the design yard, and NNS would build from that data. The use of electronic data transfers would only make the transmission of data more efficient; it would not impact the construction process.

The Relationship Between EB and NNS Evolves to Teaming
While the various data transfer working groups were developing methods to transfer data from EB to NNS, senior management of the two shipyards and the Navy were discussing more profound changes for the NSSN program. Again, the necessity for these changes was a need to further reduce program costs. The relationship of EB and NNS as lead/follow shipbuilder meant that both shipyards would have to learn the same construction techniques and intricacies for the NSSN, thereby doubling the learning curve. Costs for this would thus be doubled. In an era of reduced funding for new ship construction, this was a situation that needed to be rectified.

The solution that was agreed to by the three organizations was for EB and NNS to *team* to build NSSN. This was, and is, an entirely new concept in submarine construction. Under this arrangement, several of the NSSN modules would always be built by EB and several would always be built by NNS. Each shipyard would alternate assembling the modules into a completed submarine, and then the assembly yard would test and deliver the ship to the Navy. The rationale behind this arrangement was that each yard would gain the construction efficiencies associated with building the same modules over and over again. These modules would thus be built in a more efficient manner as time went on, reducing program cost. EB would continue its role as design agent for the NSSN, and it would be the final assembly yard for the first ship. Figure 7 shows how NSSN would be constructed under teaming.

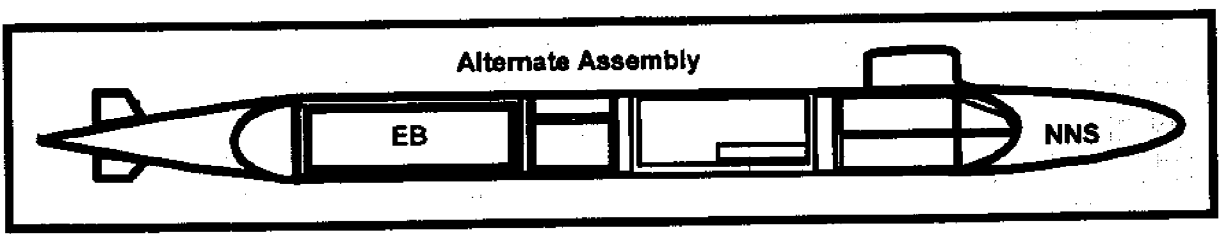


Figure 7, Showing the Break-up of NSSN Under Teaming

With the decision to team agreed to in 1996, the challenges faced by the data transfer working groups grew significantly. For teaming to deliver the projected cost reductions, every efficiency in the design/construction process had to be exploited. This included, prominently, data transfer. A major part of teaming would be for both shipyards to integrate as much as possible their manufacturing databases and achieve significant commonality of use in production systems. The environment that EB created for NSSN design/build was to be enlarged to include the NNS environment.

To support teaming, several significant changes occurred in the development process for data exchange. Since EB and NNS were planning their own destiny in the teaming arrangement, data transfer/integration became an internal issue to the teaming partners. Accordingly, the working groups ceased operations in 1996, and this activity would be picked up by one of the fifteen EB/NNS teams established to implement teaming across both shipyards. These teams were organized to oversee specific aspects of teaming such as facilities, nuclear work, ship test, and digital data transfer.

The next change to happen under teaming was the movement away from only considering data *transfers* to now consider data *sharing*. Where data transfers would actually move data from one system to another via download, neutral file, or the like, data sharing would allow the same data to be used by both parties. Data sharing could be accomplished by having access to terminals for the same system located at each shipyard or by having translators work on the fly allowing ready access to data across systems. The intent of data sharing is to drive the efficiencies gained by electronic data transfer one step further by not having multiple copies of data exist. Because of this, multiple copies do not have to be kept in synch, nor do transfers have to take place. There is only one copy of the data with the ability of multiple users, in different environments, access to it. Figure 8 shows the data sharing environment under teaming.

The last major change to happen to data transfer/sharing under teaming was that a significant increase in the amount of coding was required to support teaming. Accordingly, a series of Technical Requirements Documents (TRDs) were developed listing all of the tasks that would require coding and how it would be done. These tasks became a tool to manage the coding efforts, since they could be scheduled, monitored and budgeted. Coding was, and is, accomplished by EB, NNS, and CSC personnel. CSC is Computer Sciences Corporation, the company that EB has outsourced most of its data processing to.

Completion of each task places into production one more step in the establishment of an effective environment to construct the VIRGINIA Class of submarines; the Navy having renamed NSSN to VIRGINIA in 1998. An example of a task is the extension of pipe bend detail sheet printing to NNS with NNS specific production information included. Another is the automatic generation of table data to support HVAC fabrication at both shipyards, and yet another allows both shipyards to obtain Bill of Material data. These

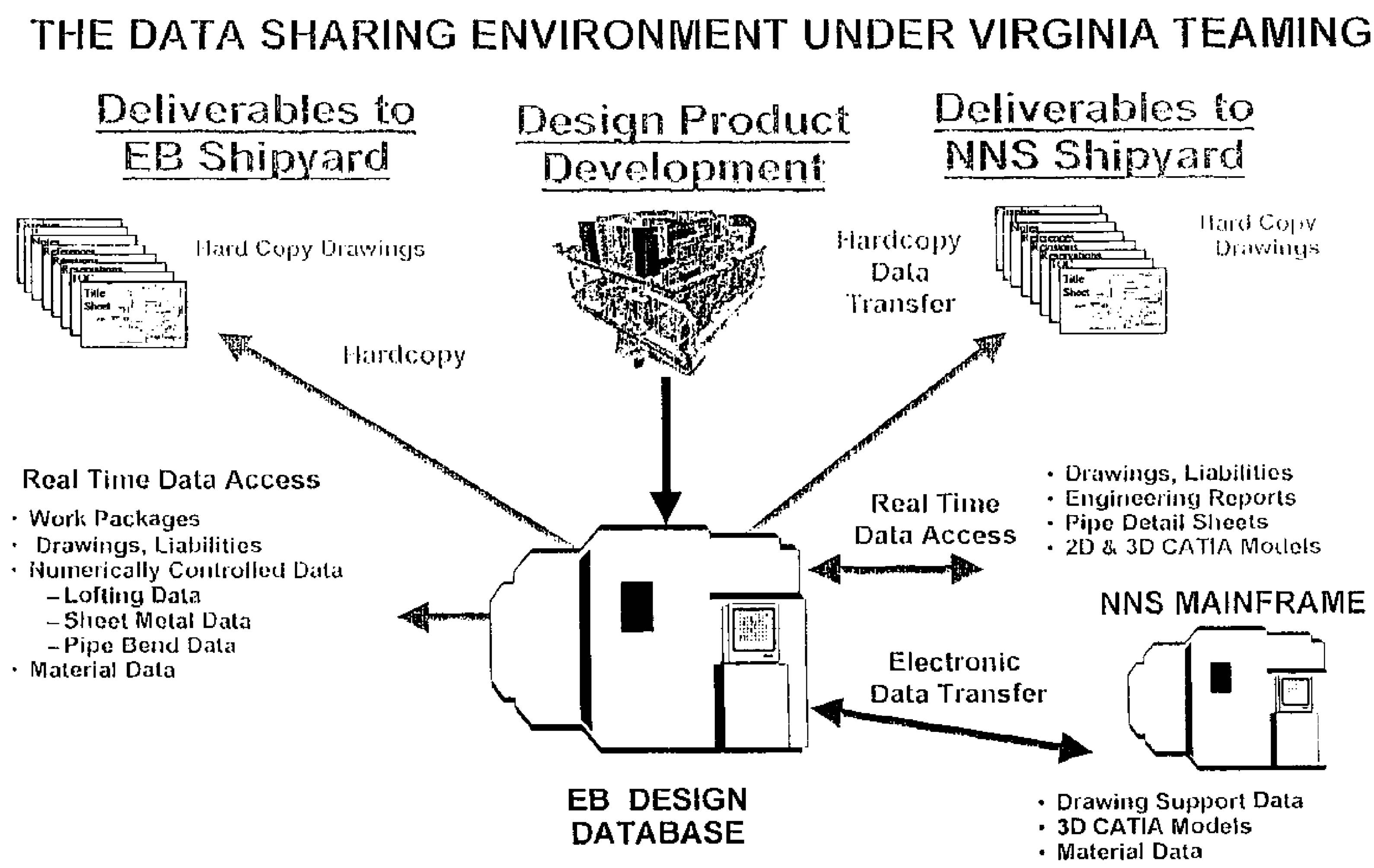

THE DATA SHARING ENVIRONMENT UNDER VIRGINIA TEAMING
Deliverables to EB Shipyard
Design Product Development
Deliverables to NNS Shipyard
Hard Copy Drawings
Hardcopy
Hardcopy Data Transfer
Hard Copy Drawings
Real Time Data Access
· Work Packages
· Drawings, Liabilities
· Numerically Controlled Data
– Lofting Data
– Sheet Metal Data
– Pipe Bend Data
· Material Data
Real Time Data Access
· Drawings, Liabilities
· Engineering Reports
· Pipe Detail Sheets
· 2D & 3D CATIA Models
NNS MAINFRAME
Electronic Data Transfer
EB DESIGN DATABASE
· Drawing Support Data
· 3D CATIA Models
· Material Data

Figure 8

tasks are all managed and reviewed by personnel from both shipyards as each affects the teaming environment.

As was noted in the discussion on the NSSN working groups, there were two areas of data that were not included in the transfer agreements, structural data and 3D CATIA models. Because teaming meant a partnership between the two shipyards, resolution to these two areas became possible. Since it is most efficient to perform structural lofting directly from a 3D model, and efficiency was a prime consideration in teaming, agreement was reached whereby the 3D structural models would be cleansed of any EB proprietary information and then transferred to NNS. NNS would then perform lofting (Figure 9) with these models for the portions of the ship that they would be building. This agreement to cleanse and transfer models was also extended to the rest of the VIRGINIA class design models. In this way, NNS would have the entire design electronically to do construction planning on.

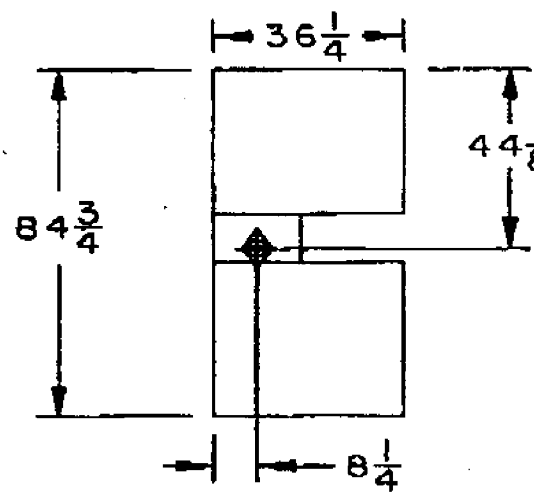


Figure 9, Typical Lofted Part

Note that above, data transfer is again being discussed. Although the goal of data flows in EB/NNS teaming is to do as much data sharing as possible, it was recognized by the involved parties that this would not happen in every case where data would be accessed by both parties. There were databases that EB did not wish NNS to have access to and vice-versa. In addition, there were production processes in place at both shipyards where the use of the same database would not be possible. In these cases, to the maximum extent possible, an electronic transfer would be implemented. With a transfer, the receiver of data could use the data configured in the necessary way to support production.

Structural lofting and construction visualization were two such cases. To support their entire construction operation, NNS maintains its own CATIA system, not directly connected to the EB environment. It is in this system that NNS performs VIRGINIA class lofting and related activities. To support these functions, a transfer of CATIA models is accomplished from the EB CATIA environment to this system at NNS. The transfer occurs on a regular basis, and NNS configuration manages the data in their system.

As with any large-scale data transfer program, effective configuration management of the data before, during and after the transfer/share is crucial to the success of the program. Recognizing this, the teaming arrangements on data transfer/sharing contain extensive and comprehensive procedures on how to configuration manage VIRGINIA construction data and use it for construction. With teaming not only linking two shipyards, but placing actual construction of the same submarine in the two shipyards, effective CM of the data will be one of the keys to the success of the program. Successful completion of the teaming arrangements for data sharing/transfer will be another.

Conclusions
At present, VIRGINIA is starting the construction process. Several prototypes have been completed to show how the design/build and data sharing processes work, and many valuable lessons-learned from these have been gained. It is certain that many more will become evident as the construction process progresses. Although much will certainly be said and written about the concept of building the same submarine at two shipyards concurrently, it will be important to realize that effective data sharing will be one of the keys to make the entire process work.

Also, what is known for certain, however, is that the overall process is being made more efficient by the entire concept of data sharing and selected transfers. All involved parties can use the same database or a recent transfer of one and know that they are working to the latest information. Because so much of the information is electronic instead of paper, concern about using out of date data is greatly minimized. Since the sharing/transfers span both shipyards, this applies to personnel at both EB and NNS.

The natural question to ask then is what does the teaming arrangement, and in particular the data sharing effort, have to offer the future of Navy shipbuilding? To answer this, one does not have to look any further than the current round of programs under concept formulation or design. All are some form of collaboration between two or more shipyards, and all must be concerned with the movement of data between involved parties. The programs considered here are LPD-17 (Bath Iron Works and Avondale), CVNX (Newport News Shipbuilding and Electric Boat), and DD21 (Bath Iron Works and Ingalls). Thus it seems that teaming will be here for the forseeable future, and teaming, by definition, involves data sharing/transfer.

If one looks at the history of data sharing/transfer for VIRGINIA, one can see that it was an evolutionary process. It started out with close parallels to the work done to support SEAWOLF, took advantage of advances in technology as the working groups progressed, and ended up in the mode of data sharing as the teaming agreement was agreed to. It is safe to assume that similar paths would be taken on programs in the future; start with a successful model of past data movement and build on this. Trying to start from scratch with unproven and untested processes and technologies could end up costing more in the long run.

Finally, one should take away from this effort the thought that data sharing/transfer must be firmly rooted in the process it is supporting. Electric Boat developed a process to support a design, and this philosophy guided not only the VIRGINIA design, but also the evolution of Newport News Shipbuilding into the program. For VIRGINIA, data is shared or transferred in as close to a usable form as possible, at the appropriate stage in the process to share or transfer. Elaborate translations are not done, nor is data transferred too early or late in the process. Thus, if data is needed in wire-frame form at start of construction, it is transferred in wire-frame and at the start of construction. Nothing elaborate, just common sense applied to the *process*.

THE APPLICATION OF PRODUCT DATA MANAGEMENT (PDM) TECHNOLOGY TO IN-SERVICE SUPPORT

Tony Fry, Head of ILS (Navy), MoD Foxhill, Bath BA1 5AB, United Kingdom
Richard Spurling, Marconi Naval Systems, Barrow-in-Furness, LA14 1AF, United Kingdom

Executive Summary

This paper describes the application of an enabling technology, known as Product Data Management (PDM), to the through-life support of surface ships and submarines. The document introduces the concept, key capabilities and benefits of an In-Service Product Model, which will become the single source of information needed to operate and support a warship throughout its operational life.

The benefits of PDM technology, particularly when implemented as part of a multi-disciplined or Concurrent Engineering approach, have been validated by investment appraisals within numerous industry sectors. Typically cycle-time and cost reductions of 30-40% are being realised. Significant improvements in personal productivity can be achieved through the elimination of time spent looking for information and its subsequent validation for accuracy and applicability

In the UK naval defence sector, PDM technology is currently being used to capture and manage the evolving product definition of the LPD(R), the Auxiliary Oiler and Astute Class Submarines. It is timely therefore to examine how PDM technology could be used and developed to provide the necessary support capability, once these and future ships enter service with the Royal Navy.

The paper provides an insight into the capability of PDM to address the management of legacy data. An initial comparison of PDM functionality with that specified for the engineering aspects of Project UPKEEP has also been made.

This paper is the first in a series of documents, seminars and demonstrations that are intended to increase awareness of the benefits to the NSC of implementing PDM. A practical example of how PDM could be used within an In-Service environment is being developed by HILS (N) and Marconi Marine, for demonstration in March 1999. The technology demonstrator will utilise real data extracted from the product development phase of LPD(R).

This document was sponsored by HILS (N).

Introduction

Product Data Management (PDM) is an enabling technology that helps engineers and others manage both information and processes. PDM technology keeps track of the volumes of data and information required to design, manufacture, assemble, test and commission and then operate, support and maintain products. This paper describes the use and potential benefit of PDM in support of in-service ships and submarines, during all phases of their life-cycle.

Information management within and between the Procurement Executive and Naval Support Command (NSC) is a major task, which consumes significant resources. The recent adoption of PDM technology by industry has been shown to offer significant benefits by improving data accuracy and product quality, whilst reducing costs, rework and manufacturing time. This new class of information systems is already being successfully deployed on a number of large and complex technical programmes in the aerospace, automotive and oil industries. In the shipbuilding industry, Marconi Marine, DCN and General Dynamics are all early adopters of the technology.

In July last year the Defence Secretary directed that the Strategic Defence Review (SDR) should include a Smart Procurement Initiative. The SDR supporting essay on 'Procurement and Industry' identified the following: "Central to its findings is a 'Through-Life Systems Approach' to procurement, which defines a new equipment or a new capability in the context of its relationship with other equipment and wider defence capability areas. A coherent process for co-ordinating a new equipment's requirement, linkage with the research programme, specification, acceptance and through-life management is a prerequisite for improved equipment acquisition" (Ref.1). Multi-disciplined groups using PDM processes and systems offer the potential to manage the relationships between these diverse requirements.

Commercial pressures to reduce cost and achieve better 'value for money' have extended beyond Product Development into the In-Service phase of the product life-cycle. There is an increasing trend towards the inclusion of Through-Life Support requirements into the initial procurement specification. This places greater emphasis on life-cycle costing as a design consideration. PDM technology provides the inherent configuration management capability to support this requirement. Effective management of support-related data is a prerequisite to reducing through-life support costs.

1. Strategic Defence Review and supporting essays published by the Stationery Office Ltd.

MoD are proactive in their adoption of CALS and CALS-like initiatives. The CALS acronym may not be used in all cases, but the concepts are being used to progress Concurrent Engineering (CE) and Business Process Re-engineering (BPRE) strategies that aim to provide improvements in competitiveness in an increasingly tough global market-place. PDM technology provides the core capability to create, manage, share and re-use information within all those organisations involved in complex programmes, often described as the 'extended enterprise'. This environment is also capable of supporting the MoD requirement for a Contractor Integrated Technical Information Service (UK) (CITIS (UK)) initiative.

Through its adoption of PDM technology and revised business processes, the UK shipbuilding industry has risen to the challenge of significantly reducing cost and cycle-time for design/build contracts. The challenge now facing MoD is how can they utilise the resultant electronic product definition to reduce support costs and increase warship availability, throughout its operational life.

A number of awareness sessions and technology workshops have already been held to promote the concept of product data management within the RN Naval Support community. To further develop the concept, a technology demonstrator is being developed. This will serve the dual purpose of increasing awareness of PDM whilst providing a mechanism for the capture of a detailed Statement of User Requirements. PDM facilitates a shared information environment between MoD (PE) and NSC for all phases of the product life-cycle.

The long-standing requirement for open systems, seamless data exchange and the management of legacy data is rapidly becoming a practical proposition following recent developments in product data management technology, in particular the adoption of Internet-based standards. This paper defines how the NSC requirements might be served by these emergent technologies, which offer the potential to meet NSC data management requirements well into the next century.

Within the NSC, a recent initiative to procure new information systems and technology has stalled in the face of technical difficulties and protracted project timescale and scope. PDM appears to offer a solution to some of the difficulties encountered by Project UPKEEP. Specifically, PDM addresses the central issue of configuration management and has the capability to collate and manage data from a number of disparate systems. An additional user benefit of PDM is the ability to relate part information to its associated graphical representation and/or other attribute information (e.g. weight, material, supplier etc.) for ease of part identification. The ability to share information electronically within a distributed environment is also a key advantage.

What is PDM? - Definition and Key Capabilities

All the information that may be needed throughout a product's life-cycle can be managed by a PDM system. PDM is not limited to managing data in the acquisition phase but, according to user needs, can manage product conception, detailed design, prototyping and testing, manufacturing or fabrication, operation, maintenance and update.

PDM is the enabling technology that helps manage, share and re-use both data and processes throughout a product life-cycle. In today's extended enterprise PDM provides the capability to:

- enable integrated product team working;
- manage information, processes and people;
- provide a communication focus;
- ensure information integrity;
- provide information for decision-making;

- support cultural change;
- deliver direct and indirect benefits;
- manage product configuration

For the purpose of this paper, the term 'Product' will be used to refer to the ships, submarines and major equipment operated by the Royal Navy. The sub-set of applications, product information and processes that relate specifically to the maintenance and. support of these ships and submarines will be referred to as the 'In-Service Product Model' (see Annex 1 - Glossary of Terms).

Figure 1. In-Service Product Model Extended Enterprise

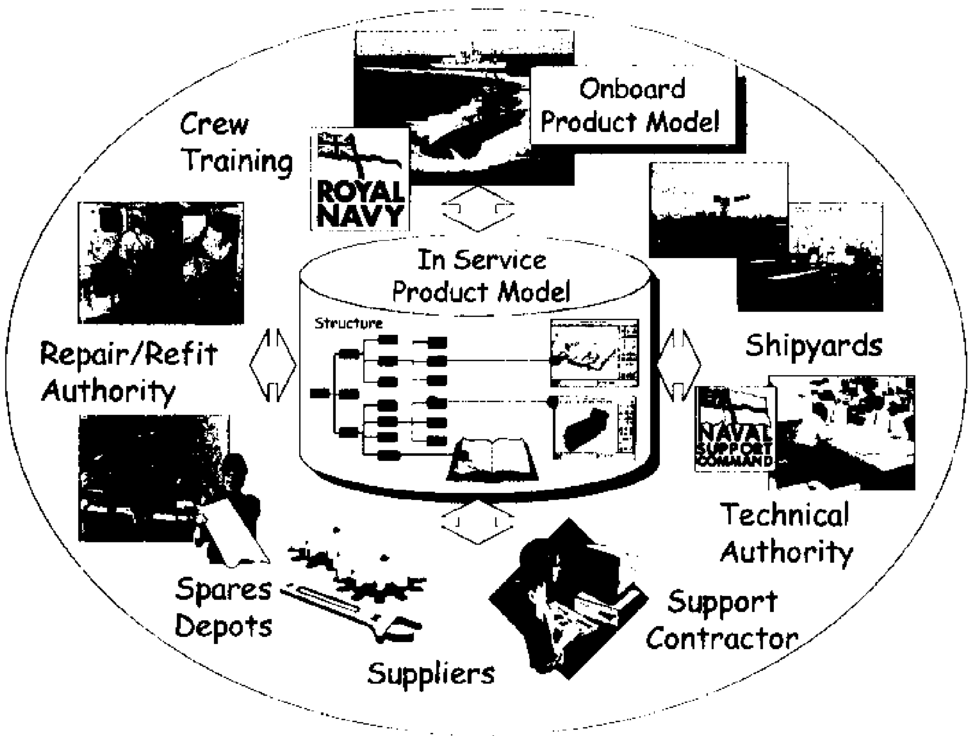


Figure 1 describes the 'extended enterprise' that it is envisaged would be involved in the creation, management and re-use of an In-Service Product Model. The main benefit of a common data source under configuration management control is that everybody has access to the same, up-to-date and accurate information.

PDM systems and processes provide a structure in which all types of information used to define, manufacture and support products are stored, managed and controlled. Typically, PDM will be used to work with electronic documents, digital files and database records.

These can include but are not limited to:

- Customer requirements.
- Product configurations.
- Part definitions and other design data, including part /stock numbers.
- Specifications.
- 3D CAD models and associated drawings.
- Images (scanned drawings, photographs, etc.).
- Engineering analysis models and results.
- Logistic analysis and spares provisioning.
- Manufacturing process plans and routeings.
- Maintenance schedules.
- Computer Numerical Control part programs.
- RAMP data files.
- Software components of products.
- Electronically stored documents, notes and correspondence.
- Electronic Technical Documentation (IETPs).
- Audio and live video annotations.
- Project plans.

The product life-cycle process is managed, as well as the product information. PDM systems control product information states, approval processes, authorisations and other aspects related to overall configuration management. PDM systems also have the capability to manage different life-cycles for different types of information. By providing data management and security, PDM systems ensure that users always get and share the most recent, accurate and approved information.

Data from other computer applications is controlled by the PDM system's assumption of roles of file access and saving. This is accomplished either by embedding:

- PDM commands in other applications that create data (such as CAD, word processors, spreadsheets, desktop publishing or specialised software);
- commands from those applications into the PDM system.

Legacy documents, such as drawings on paper or aperture cards, or other hard copy records can be captured by scanning and storing them within the electronic vault as images. Some users prefer to convert images of textual documents by Optical Character Recognition (OCR) into computer readable text. It is feasible to convert some Raster (scanned) images of drawings into vector data via software, while some users prefer to capture data by digitisation techniques.

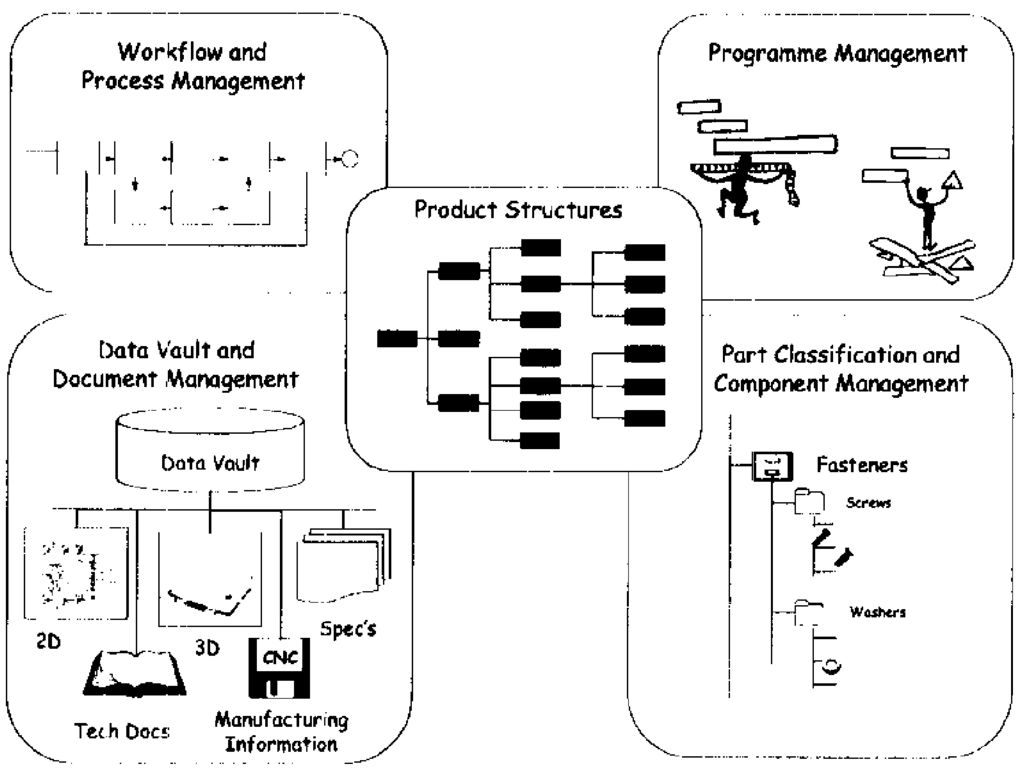
An electronic vault is used as a repository to control all kinds of product information. The vault is a data store that contains some data within itself and controls other externally-generated data by managing access to it. Two types of data are stored:

- Product data generated in various applications, such as specifications, CAD models, CAE data, maintenance records, and operating manuals.
- Meta-data, which is data about PDM-controlled information.Meta-data is stored in a PDM database and supports the functions performed by the PDM system.

The functionality of PDM systems falls into two broad categories: User functions and Utility functions. User functions provide the user's interface to the PDM system's capabilities, including data storage, retrieval and management. Different types of users exploit different sub-sets of the user functions. These functions are divided into five categories:

- Data Vault and Document Management.
- Part Classification and Component Management.
- Product Structure Management.
- Workflow and Process Management.
- Programme Management.

Figure 2. PDM User Functions



Utility functions provide support that facilitates the use of the system. Utility functions interface with the operating environment and insulate its functions from the user. Tailoring permits systems to operate in conformance with the user's environment. The graphical user interface can be made role specific, i.e. presenting the user with access to only those applications needed to perform his/her task, e.g. Maintenance Engineer, Planner, Operator, Configuration Control Engineer, Estimator, Commodity Manager, etc.

Utility functions include:

- System Administration and Security.
- Communication and notification (e.g. E-mail).
- Data Access and Distribution.
- Data Translation (one application to another).
- Image Services (e.g. visualization, view and mark-up).

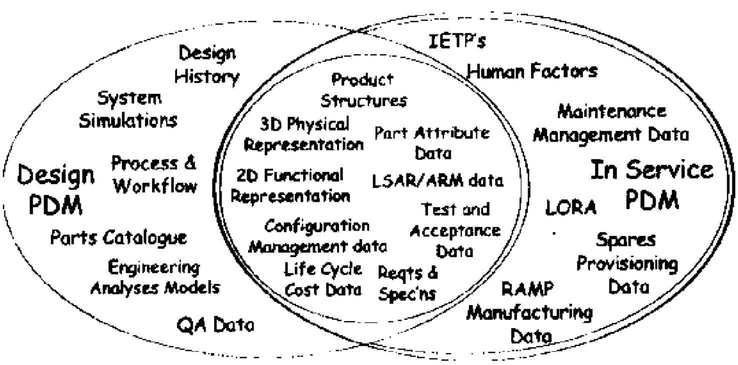
The Proposed Role of PDM in the NSC

This section of the White Paper describes a 'vision' of how PDM could be used in the Naval Support Command (NSC) to improve warship availability and reduce through-life operating costs.

Defence Standard 00-60 defines the Ministry of Defence (MoD) requirements for the application of Integrated Logistic Support (ILS) principles and through-life management of equipment. It provides the discipline for ensuring that supportability and cost factors are identified and considered during the design stage of equipment, so that they may influence the design with the aim of optimising the Life-Cycle Cost (LCC). It is intended to be used, whenever relevant, in all future designs, contracts, orders, etc. and whenever practicable by amendment to those already in existence.

Consideration is being given to the way in which a sub-set of the product development information, suitably enhanced by specific through-life support information, can be used to satisfy the requirements of Defence Standard 00-60 in both the design and in-service phases of the product life-cycle. The figure below indicates the overlapping data requirement for design and support.

Figure 3. Overlap between Design and Support Data



It is envisaged that the comprehensive data set to support both the design and in-service support phases will contain, but not be limited to, the following information:

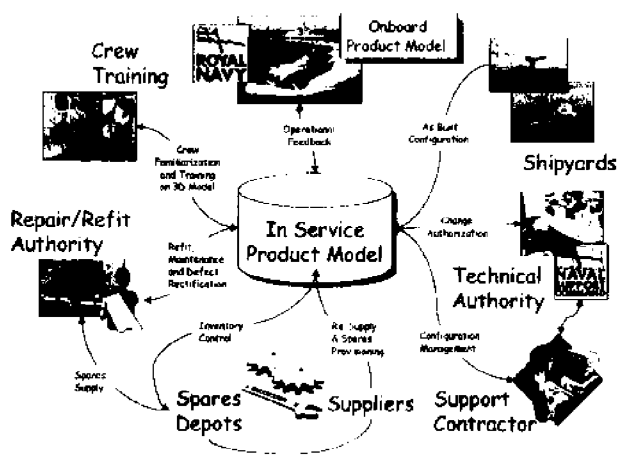
- Full Electronic Product Definition (functional and physical representations).
- Configuration Management Data (inc. Version and Revision, Effectivity, Alternatives and Substitutes).
- Manufacturing Data (inc. RAMP).
- Logistic Support Analysis Record (LSAR).
- Availability, Reliability and Maintainability (ARM) Data.
- Maintenance Management Data.
- Spares Provisioning Data and Tools.
- Interactive Electronic Technical Manuals (IETMs).
- Training Documentation (including simulations and 'virtual reality' representations).
- Test and Evaluation Documentation and associated Data.
- Life-Cycle Cost Data.
- Human Factors Data.

The anticipated users of an In-Service PDM fall into seven main categories:

- Support Agencies - Technical/Design Authority.
- Shore-based Provisioning Depots.
- Repair/Refit Authority.
- Contractors/Suppliers.
- Training Establishments.
- Onboard.
- Disposal Authority

Each of the above will require access to information as outlined on following page.

Figure 4. 'Virtual' Extended Enterprise access to the In-Service PDM



The entire Support organisation can be treated as a 'virtual' extended enterprise, each has access to the relevant information within the In-Service Product Model. Shared access ensures the various elements of the enterprise have timely and accurate information for their decision-making and planning.

The management of legacy data is a major consideration for the NSC. The introduction of the latest 'Internet-based' PDM systems makes linking disparate databases a practical possibility. Legacy data exists in a variety of formats which makes integration by traditional 'point to point' methods extremely difficult, expensive and in some instances impossible.

Modern information systems make the exchange of data between disparate elements of the 'virtual enterprise' possible through the use of neutral data formats and 'de facto' standards for data exchange, such as the STEP Application Protocols and CORBA.

This capability supports the CALS objectives for the creation of a Contractor Integrated Technical Information Service (CITIS), allowing all project data to be held, managed and distributed electronically.

PDM and the Potential for Project Upkeep

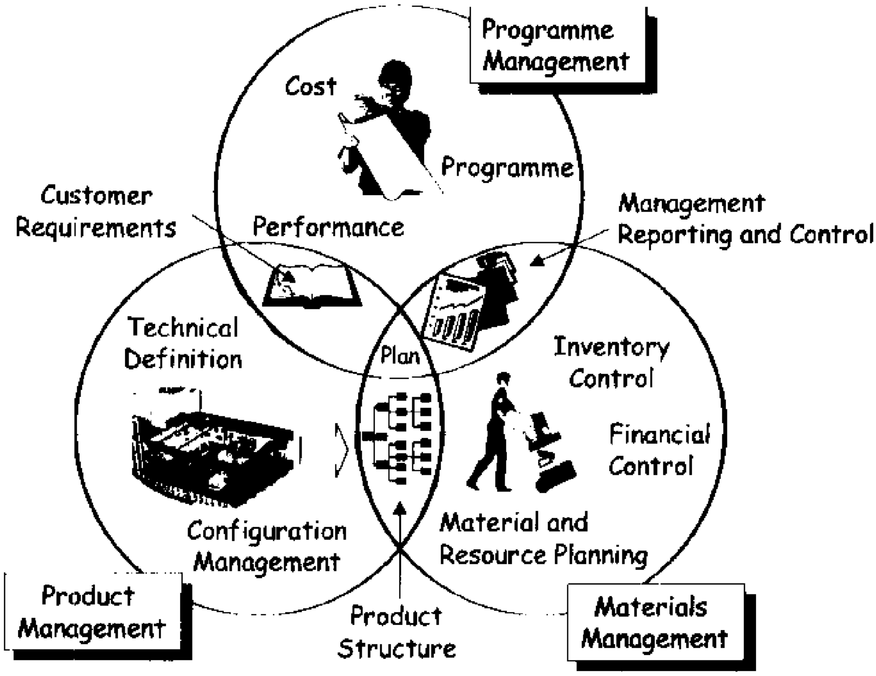
Project UPKEEP addresses the provision of maintenance, update and repair services to ships and equipment, together with the associated spares purchasing, supply and accounting within the Naval Logistics area. It is envisaged that Project UPKEEP will support the following:

- HM Ships, Submarines and RFAs.
- Shore Establishments.
- Store Depots and Warehouses.
- Central Administration.
- Elements of the Fleet Air Arm.
- Elements of the Royal Marines.

As currently defined, Project UPKEEP represents an enterprise-wide integrated information systems capability. The functionality offered by such an integrated business solution typically consists of three major elements:

- Programme Management (Planning, Budgets, Management Control and Reporting, etc
- Product Management (Technical Definition, Configuration Management).
- Materials Management (Material and Resource Planning, Financials, Inventory Control Part Management, Supplier Management, etc.).

Figure 5. Integration of key business functions



The functionality offered by PDM systems has been developed to manage, share and re-use the product definition and provide the underlying configuration management of product-related data. Data Accuracy and Configuration Management are the most important aspects to consider when designing information systems. To ensure maximum flexibility in the management of data it is important to be able to treat each element as an object in its own right, each with its own life-cycle.

In addition, the following basic features of PDM systems make the management and control of data easier and more readily accessible:

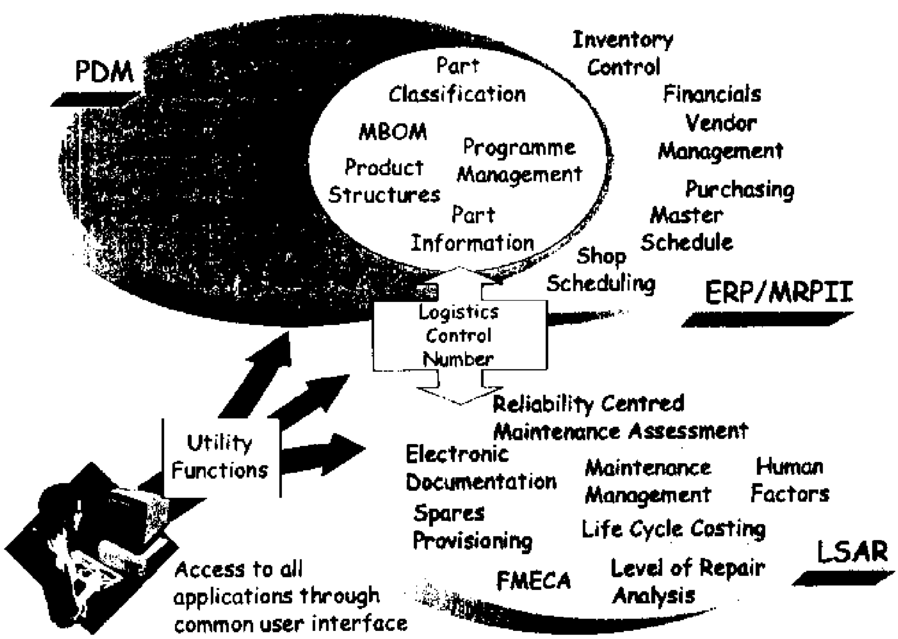
- Access Control and Life-Cycle Management (Change Control).
- Product Structures - used for data navigation and part identification.
- Visualization - identification and understanding of parts within the context of an assembly.
- Distributed Vault - to allow data distribution across an extended enterprise.
- Workflow - to manage the product definition and change management process.

Table 1. Assessment of Project UPKEEP requirement

UPKEEP Requirement	COTS Solution Application Type	COTS Solution Example
High Level Planning	Planning Application	Artemis Plan View MS Project
Configuration Management	PDM	Windchill Metaphase, Sherpa
Maintenance Planning and Management	PDM	Windchill, RCMS Metaphase, Sherpa
Equipment and Supply Chain Management	MRPII/ERP	SAP R3, Baan Compass Contract
Physical Stock Management	MRPII/ERP	SAP R3, Baan Compass Contract

The Table above illustrates how the Project UPKEEP requirement can be satisfied by 'Commercial-off-the-Shelf' (COTS) applications. Each application is accessed through a single common Graphical User Interface (GUI). The user need not be aware which application is used for which purpose. Access is managed by core PDM utility functionality.

Figure 6. Single Common GUI provides access to all applications



Business Benefit

The implementation of PDM requires a major commitment to process and technology change within an organisation. It is usual to undertake an investment appraisal prior to entering into this commitment. The benefits of adopting PDM have been proven in a variety of industries and can be summarised by the following:

- Improved Configuration Management.
- Improved Data Accuracy and Access.
- Reduced Cycle-Time.
- Reduced Cost.
- Improved Quality.
- Effective use of Resources.
- Improved Communication.

These are equally applicable to the Naval Support Command, resulting in increased warship availability and reduced through-life operating costs.

Linking PDM with Concurrent Engineering techniques to build complex products such as ships, aircraft and automobiles has been shown to offer reductions in Cycle-Time and Cost of the order of 30-40%.

At a business level, one of the major benefits of PDM is the ability to manage information throughout the whole product life-cycle, from cradle to grave. It supports the sharing of information between all involved in the project, i.e. customer, contractors and suppliers, within the 'virtual extended enterprise'. The ability to exchange information electronically between enterprise partners is consistent with the aims and objectives of CALS.

Within an organisation, the adoption of PDM can significantly improve productivity through the re-use of data and the adoption of common standards. The ability to share information and associate it with product development process enhances the corporate knowledge. Teamworking, empowerment and improved communications are all facilitated by secure and controlled access to a common source of accurate information.

Personal productivity can be significantly enhanced by the implementation of PDM. Studies have shown that the time spent looking for and validating information can represent 60% of a typical working day. Access to information through PDM can reduce this time by half. Powerful search capabilities provide rapid access to information wherever it may be held. The ability to support data exchange between systems also avoids the need for re-keying information.

Improved quality and reduction in rework are benefits that can be directly attributable to the implementation of PDM. Experience in both aerospace and shipbuilding indicates that reductions in rework of up to 90% can be achieved. The ability to manage 'As designed', 'As built' and 'As maintained' configurations supports faster response to customer queries, leading to reduced cycle-time.

PDM supports process improvement through the automation of activities and workflows. Document management and engineering change control benefit from the ability to leave behind the traditional paper chase. Accurate Bills of Material (BoM) associated with automated work orders provide a capability to ensure the right information and materials are available at the right time. Experience in industry has shown that BoM accuracy in excess of 98% is typical. Access to process-based metrics provide the basis for continuous improvement.

Conclusions and Next Steps

Affordability is a prime requirement to be satisfied by any defence programme. The pressures on the national defence budget have resulted in even greater emphasis being placed on achieving 'value for money' in all areas of defence procurement.

In his introduction to the Strategic Defence Review, the Secretary of State for Defence identified the need to "...apply modern management methods to ensure that we deliver results as efficiently as possible to the taxpayer". The appointment of a tri-service Chief of Defence Logistics creates a single point responsibility "...who will be responsible for delivering best business practice throughout our support services".

PDM has been implemented as an essential element of Business Process Re-engineering at all the leading shipyards. The challenge for the NSC is to build on this technology and ensure the seamless transfer of information between design and support. This represents a low risk enhancement of proven technology, which has been shown in related industries to deliver significant business benefit.

Any solution developed for the NSC is also likely to be applicable to the other services as part of a meeting the tri-service support requirement defined in the SDR.

Recent developments in information systems provide for increased flexibility in database integration. This means that business solutions are easier and quicker to implement, leading to reduced costs and increased functionality. It is appropriate to investigate how these advances can be used by the NSC to define an enterprise information strategy, capable of meeting its requirements well into the next century.

HILS (N) are developing a PDM demonstrator, which will utilise real data from the LPD(R) contract, to show 'proof of concept' within the extended NSC community. It will be used to increase awareness of PDM and can be used to elicit a Statement of Requirements from potential users, e.g. technical support authorities, repair and refit authorities, training establishments, nominated suppliers, contractors, and onboard. This will feed into the enterprise information strategy for the whole of NSC.

Pending the definition of an overall strategy, it is recommended that work continue in deploying PDM for the in-service support of LPD(R) and AO. This will provide a combination of improved definition of requirement and early benefits in the quality of support data. The inherited design data will be enhanced with support-specific data such as ILS, RCM and RAMP.

Annex 1 Glossary of Terms

AO	Auxiliary Oiler
ARM	Availability, Reliability and Maintainability
BoM	Bill of Material
BPRE	Business Process Re-engineering
CAD	Computer-Aided Design
CAE	Computer-Aided Engineering
CALS	Continuous Acquisition and Life-cycle Support
CE	Concurrent Engineering
CITIS (UK)	Contractor Integrated Technical Information Service (UK)
CORBA	Common Object Request Broker Architecture
COTS	Commercial-off-the-Shelf
GUI	Graphical User Interface
HILS (N)	Head of Integrated Logistic Support (Navy)
HTML	Hyper-Text Mark-up Language
IETM	Interactive Electronic Technical Manual
IETP	Interactive Electronic Technical Publication
ILS	Integrated Logistic Support
ISO	International Standards Organisation
LCC	Life-Cycle Cost
LPD(R)	Landing Platform Dock (Replacement)
LSAR	Logistic Support Analysis Record
MoD (PE)	Ministry of Defence (Procurement Executive)
NSC	Naval Support Command
OCR	Optical Character Recognition
PLCS	Product Life-Cycle Support
PDM	Product Data Management
RAMP	Rapid Acquisition of Manufactured Parts
RCM	Reliability Centred Maintenance
RFA	Royal Fleet Auxiliary
SDR	Strategic Defence Review
STEP	STandard for the Exchange of Product Model Data

A PRODUCT MODEL FOR SHIP FUNDAMENTAL DESIGN BASED ON WORKFLOW ANALYSIS

Hiroyuki Yamato, Takeo Koyama, Akira Fushimi, Hiroshi Masuda and Akifumi Iwashita,
University of Tokyo, Tokyo, Japan

Abstract

In general, the product model may be defined based on the physical structure of the product. However, the product model must comply with the design procedure. In this paper, the product modeling methodology is discussed to include the design process based on the workflow analysis. The design workflow is represented according to IDEF3 methodology to define the process and data transition. General Product Modeling Environment has Common Frame Library, which is a collection of the commonly usable classes of parts of ship. GPME has been used to add a specific class to represent the intermediate product definitions in the design process as a part of Extended Frame Library. And the effectiveness of the methodology is illustrated by the corrugated bulkhead structure design.

Introduction

The product modeling is the main concern for the informed shipbuilding industry. The single source data may be most effectively available for the Computer Integrated Shipbuilding from very basic conceptual design through the ship production and even for the maintenance. The product model must be easy to get and to give data in every stage of the design, production, operation until decommission. The product model can be mainly achieved by examining the physical structure of the product. However, it is rather difficult to take account of the work process, especially the design process, into the product model structure.

On the other hand, the workflow analysis is becoming one of hot issues recently in order to streamline the design and production process for higher efficiency in time and labor. Some effective methodologies such as IDEF3 have been proposed. The workflow analysis enables to clarify knowledge concerned with the work process. The result of workflow analysis is useful for the development of a product model to support the design process.

In this paper, authors would like to discuss the workflow analysis and product modeling methodology in the ship structural design to include the design perspective into the product model. To represent the workflow by IDEF3 to capture the data handled in each activity, and results obtained by the IDEF3 analysis is combined to the product model generation. General Product Model Environment was used to add the class corresponding with design procedure.

Workflow analysis

The workflow analysis is necessary to clarify the intermediate definitions in the design process. That is because the requisite intermediate definitions deeply depend on the design workflow and they are influenced by the way of design. In this chapter, the design workflow analysis methodology will be presented.

Formal workflow representation

The design process is not clearly defined especially in industries whose products are complex as shipbuilding industry. Ambiguity or options to take may exist in many aspects of the design process. Even though skilled designers cannot state their way of design in a concrete and unique manner. However, to implement the workflow on the computer, it is necessary to describe the design workflow in a formal way. And not only the process but also the transitive intermediate product data must be represented clearly.

IDEF3 Process Description Capture Method

In this analysis, IDEF3 Process Description Capture Method [1] was used. IDEF3 is one of IDEF, an acronym of Integrated Computer-Aided Manufacturing DEFinition, methods that are used to perform modeling work in support of enterprise integration [2]. IDEF3 provides a mechanism for collecting and documenting processes. It enables to express knowledge how a particular system, process or organization works accurately.

As shown in Figure 1, IDEF3 has two description modes to show the work and object transition in the process. Those are a Process Flow diagram and an Object State Transition Network, OSTN, diagram. The capability of describing in two different perspectives is a great advantage of IDEF3.

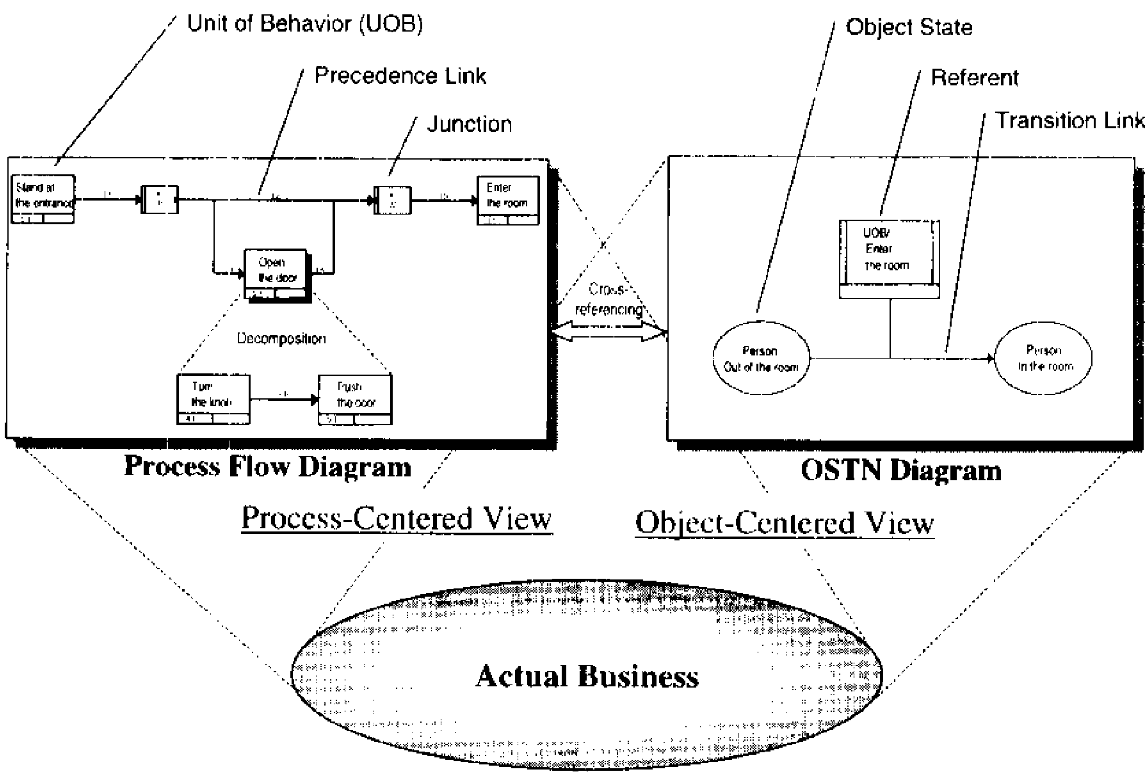


Figure 1. Concept of IDEF3 Description

Process Flow diagrams show the workflow as a chain of units of behavior, UOBs, to capture process-centered views of processes. Precedence links represent the order of UOBs and Junctions do the logic of process branching, 'and', 'or', 'synchronous and', 'synchronous or' and 'exclusive or.' The shadow of UOB indicates that one or more decomposition diagrams are associated with the UOB for more detailed description.

OSTN diagrams capture object-centered views of processes, summarizing the allowable transitions that objects can undergo throughout a particular process. The relation to Process Flow diagrams can be visible by using Referents, each of which corresponds with one of the UOBs in Process Flow diagrams. Referents are also used in Process Flow diagrams. OSTN diagrams are typically developed only for the important objects of the process description. OSTN diagrams are most often developed after Process Flow diagram; however, it is easier to begin with OSTN diagrams in some cases. There is no definite procedure to develop those two diagrams.

Each element in the two diagrams can have Elaboration to store detailed documentation. The elaboration document typically includes listings of the objects, facts and constraints, also a textual description of that element.

IDEF3 application to the design process analysis

As stated above, IDEF3 methodology was used in this analysis. But IDEF3 was designed for the analysis of generally various business processes not only for the design process analysis. It is necessary to determine how to apply IDEF3 methodology to the design process analysis for making the analysis more effective. So authors defined some policies of IDEF3 application to the design process analysis as following.

UOB elaboration

In this research, especial UOB elaboration documents are attached to each UOBs. They enable readers to understand the meaning of UOB more clearly. Figure 2 shows the UOB elaboration document form used in this analysis. It includes a textual general definition of activity and listings of the work items, created data at this activity, reference data created in preceding activities and external information & knowledge. The work items become UOBs in the decomposition diagram of that UOB.

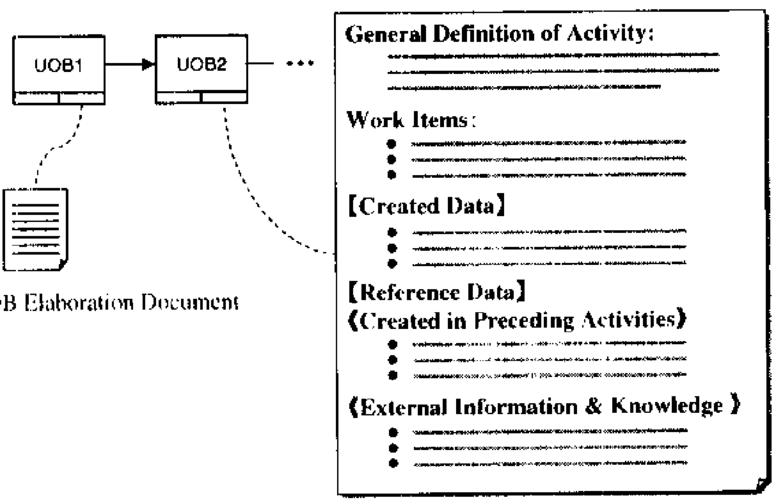


Figure 2. UOB elaboration document form for the design workflow analysis

OSTN diagram

In this analysis, as the objects of interest are the product definitions, OSTN diagrams should be developed for capturing the transition of the product definitions through the process. It may be natural that existent drawings and technical documents in actual work are made objects of OSTN diagrams. But it is not convenient for the development of product model because the product definitions for designers are elements represented in drawings and documents, not drawings and documents themselves. The object of OSTN diagram in this analysis is regarded as a conceptual design object for the designer, such as the midship structure, the bulkhead structure, the transverse web structure, and so on.

ProSim

ProSim [3], a tool to support IDEF3 methodology, was used in this paper. ProSim is a software tool developed by the Knowledge Based Systems, Inc., and this provides the environment for developing IDEF3 descriptions. The IDEF3 analysis was achieved very easily by using this software.

Workflow analysis of a bulk carrier hull structural basic design

In this chapter, the workflow analysis of a bulk carrier structural design will be presented. Figure 3 is a picture of a typical bulk carrier considered in this analysis.

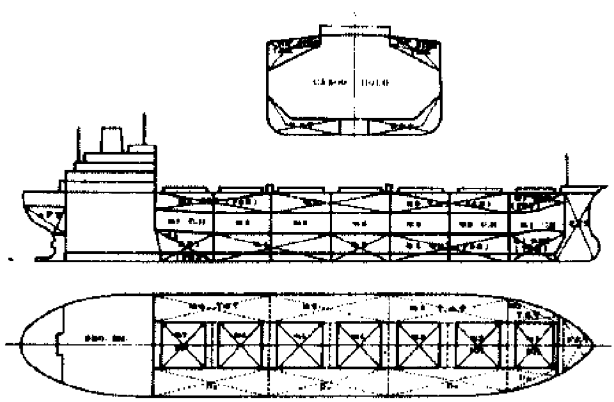


Figure 3. Picture of a typical bulk carrier

Hull structural basic design

In this research, the phase of hull structural basic design was analyzed. As shown in Figure 4, it locates between a preliminary design and a detailed design in whole hull structural design. The output of the hull structural basic design is called 'Key Plan', that is basic definitions of hull structure and contains diagrams of midship section, construction profile, deck plan, shell expansion, and so on. The intermediate definitions of hull structure in the basic design are not clear and accurate because the definitions of hull structure in this phase is mostly rough and abstract. The fact is a significant factor why systems in this domain are not integrated well and cannot provide sufficient support for design work.

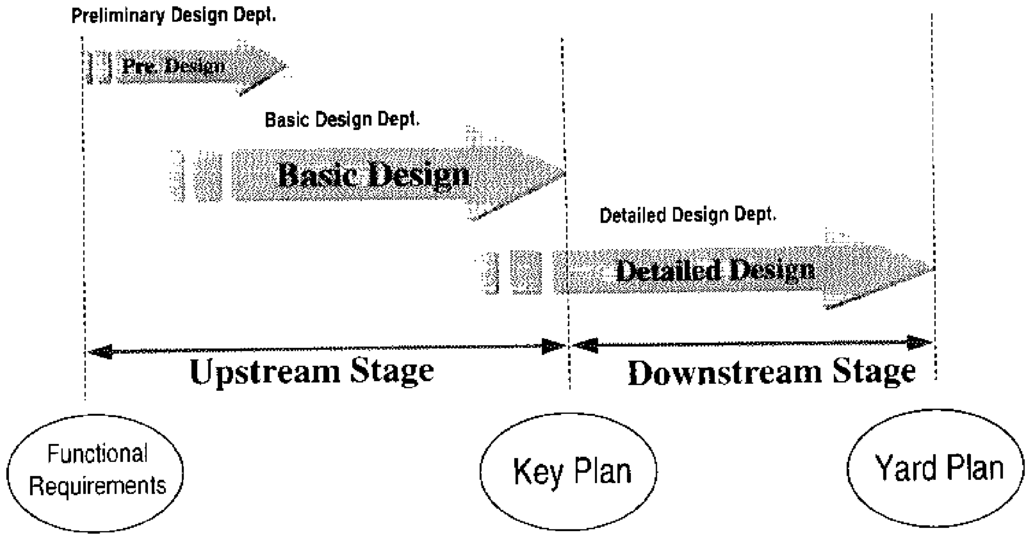


Figure 4. Hull structural design

Captured descriptions in this analysis

In this section, some of the captured diagrams in this analysis will be presented to show how to apply IDEF3 methodology for the design workflow analysis.

Figures 5a and 5b together show a top-level Process Flow diagram of hull structural basic design. It shows a workflow from the design policy decision till getting approval of the key plan. This is the most abstract but overall description of the hull structural basic design. Each UOB which has shadow contain a decomposition diagram for more detailed description.

Figure 6 shows the Process Flow diagram of midship section longitudinal member design, which is a decomposition diagram associated with the UOB 'Design midship section longitudinal member' in the top-level Process Flow diagram. Much more detailed decomposition diagrams exist under the shadowed UOBs. A lot of decomposition diagrams are captured like this.

Figure 7 shows an OSTN diagram for the midship structure definitions. Schematic descriptions of each oval were also represented in Figure 7. The latter part of the diagram is not included in this Figure to avoid lengthy description. The development of OSTN diagrams is based on the descriptions of Process Flow diagrams. By making OSTN diagrams, the transition of product object definitions became clearly understandable like this.

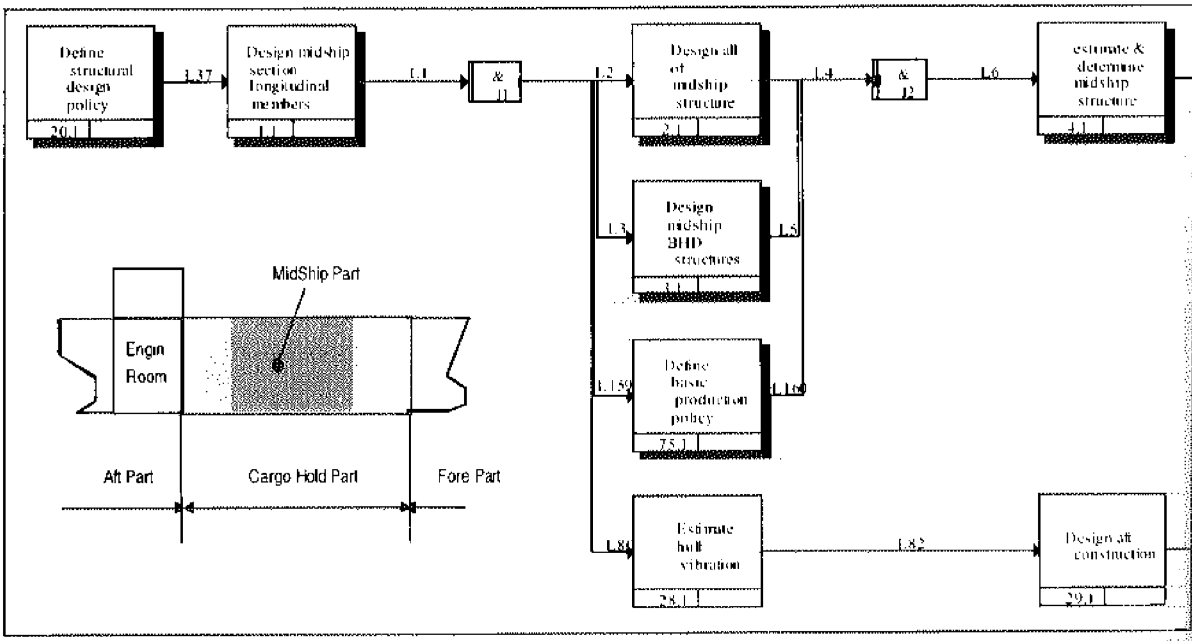


Figure 5a. Process Flow diagram of hull structural basic design (1 of 2)

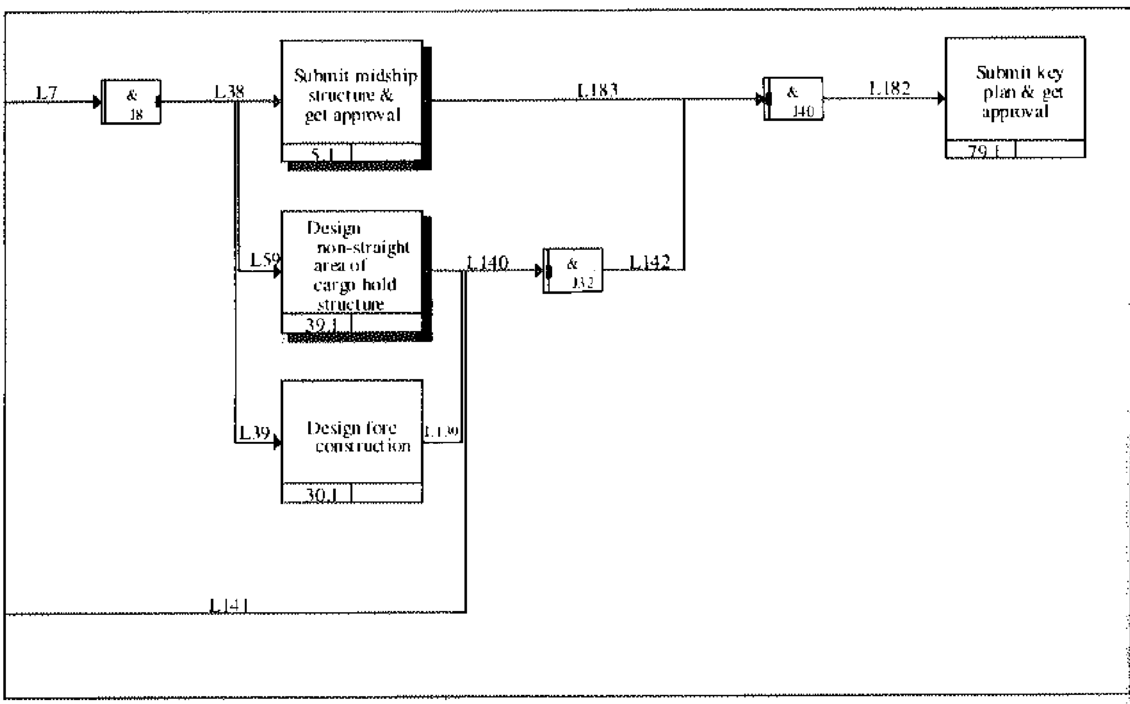


Figure 5b. Process Flow diagram of hull structural basic design (2 of 2)

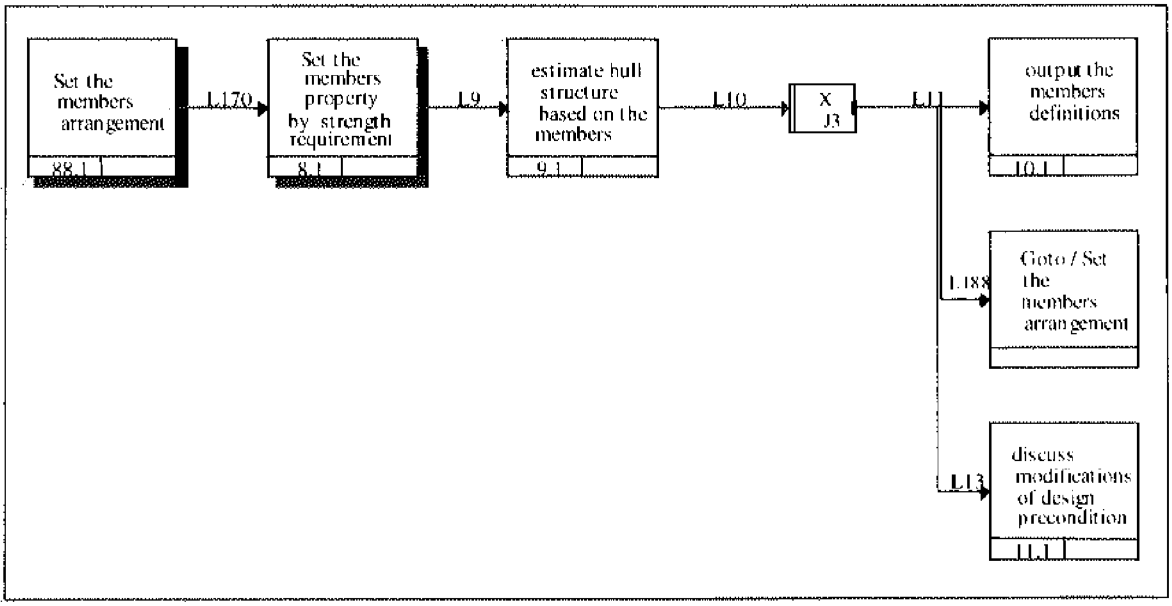


Figure 6. Process Flow diagram of midship section longitudinal member design

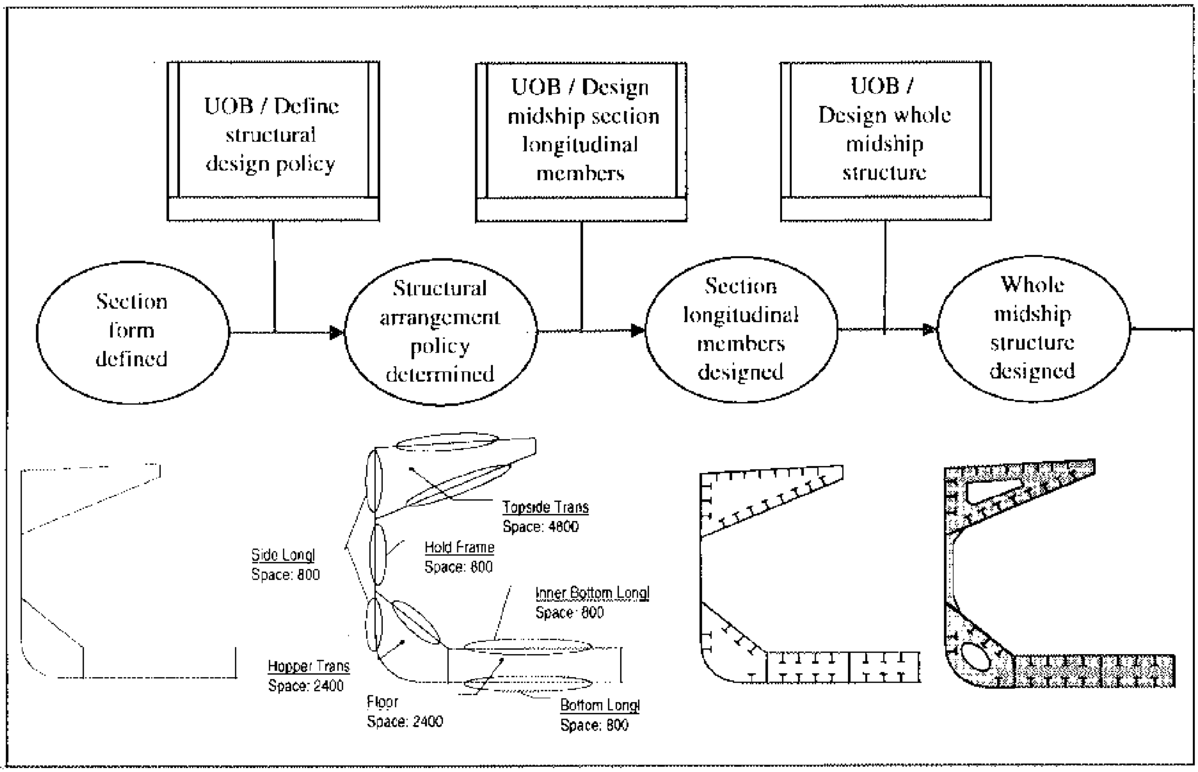


Figure 7. Transitions of Midship structure definitions (fore part only)

EFL development based on the workflow analysis

GPME in this paper has the first version of Extended Frame Library for Shipbuilding, EFL/S. The first version of EFL/S can represent the basic hull structure. However, it is not sufficient for the ship design process to represent intermediate definitions of the product under design. Additional EFL development for more effective design support should be necessary, and this can be accomplished through the workflow analysis of the design process.

In this chapter, an overview of GPME and its frame libraries is described first and a suggestion and a prototype of extension for bulk carrier transverse corrugated bulkhead structure definitions is shown next as a concrete example of the additional EFL development.

GPME and its frame libraries

GPME is an environment to provide an object-oriented product model efficiently usable in the wide variety of the automotive, construction, shipbuilding and so forth [4] [5] [6]. GPME is a product of the consortium research project executed by Ship and Ocean Foundation and major Shipbuilders in Japan. The architecture of GPME is shown in Figure 8.

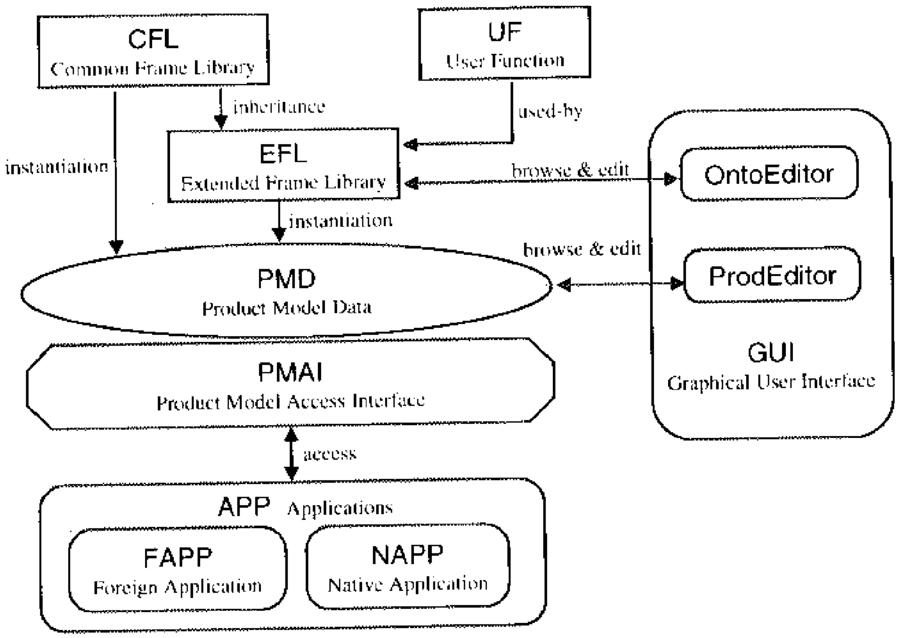


Figure 8. GPME reference architecture

GPME furnishes Common Frame Library to produce Extended Frame Library by OntoEditor. The product has the common structure in the form of the object-oriented database. The commonality may be used very conveniently in each field of industry and users need only to add their own specialty for their own product model to represent specific data in the product. GPME gives the environment to give user's own product model very efficiently. CFL is to represent common parts of the product in the same industry. EFL should be added on CFL. The end-user can define product models in a graphical way in OntoEditor and the system provide the data structure in an object-oriented database. And the product instance can be browsed with ProdEditor. GPME has been evolved in the consecutive research project nick-named Advanced-CIM after the end of the original project in 1997. However, the original GPME without sophisticated CFL was used in this paper. The original GPME has the first version of EFL/S. Figure 9 shows distinctive classes introduced to the structural design FL in EFL/S. The intermediate definitions cannot be completely represented and the hull structural basic design are not sufficiently supported by this FL.

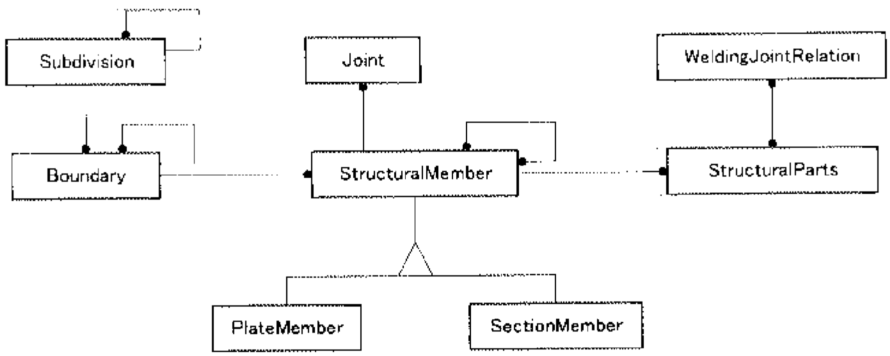


Figure 9. Distinctive classes of the structural design FL in EFL/S

An example: EFL to design corrugated bulkhead structures

Figure 10 shows the corrugated bulkhead structure in a bulk carrier. It is composed of upper and lower stools, and a corrugated plate.

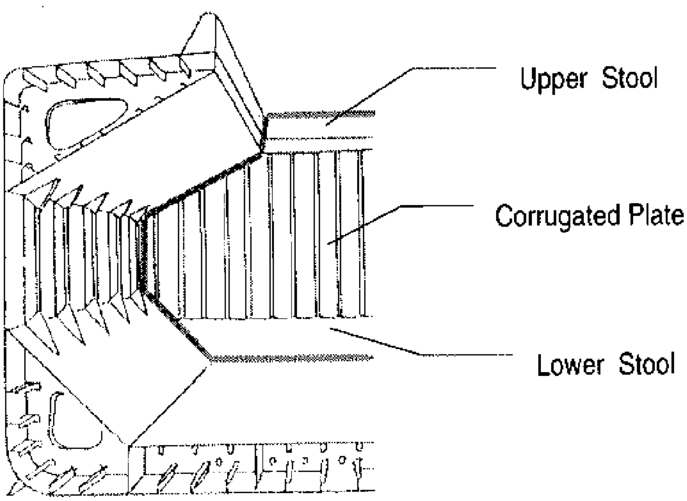


Figure 10. Picture of a corrugated bulkhead structure in a bulk carrier

Figure 11 shows the transition of all corrugated bulkhead structures captured in this analysis. According to the diagram, the following design process can be read and understood:

Before the hull structural basic design, only the positions of all bulkhead structures have already been determined. In the process of hull structural basic design, the midship bulkhead structures, i.e. bulkheads in the midship area, are designed first and the other bulkhead structures are designed later. This is shown in the mezzanine three ovals.

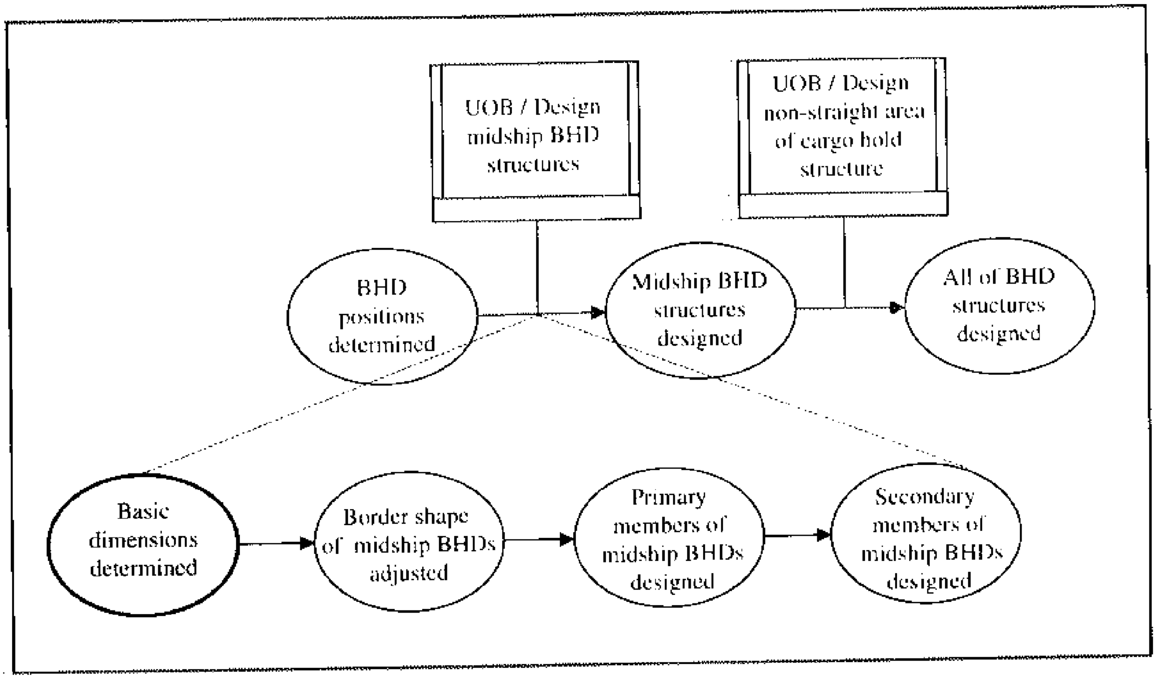


Figure 11. Transition of all corrugated bulkhead structures definitions

Actually, more detailed object state transition is included in the UOB 'Design midship bulkhead structures' in the lowest four ovals. It indicates that four detailed object states exist between the object state 'BHD positions determined' and the object state 'Midship BHD structures designed'. This transition expresses the following:

In the design process of midship bulkhead structures, the basic dimensions of bulkhead structure shape is determined first. The border shape adjustment for individual bulkhead, which means the modifications of bulkhead structure shape based on the basic dimensions, is executed secondly. The configuration and attributes of primary members are designed thirdly. And the arrangement and attributes of secondary members are designed lastly

It should be considered that definitions of the basic dimensions in the object state 'Basic dimensions determined' are actually intended for not only the midship bulkhead structures but also the other bulkhead structures. And the basic dimensions are designed by strength requirements. In strength requirements, all of bulkhead structures in a bulk carrier are regarded as one of the two types, i.e., watertight bulkhead and deep tank bulkhead. So two definitions of the basic dimensions normally exist. Each individual bulkhead structures are designed on the basis of the two definitions.

According to the above interpretation that is extracted from the OSTN diagram of corrugated bulkhead structures definitions, authors considered that the object structure of corrugated bulkhead structures definitions in the design process should be as Figure 12. It is composed of objects that express individual bulkhead structures and objects that contain the basic dimension definitions of bulkhead structure.

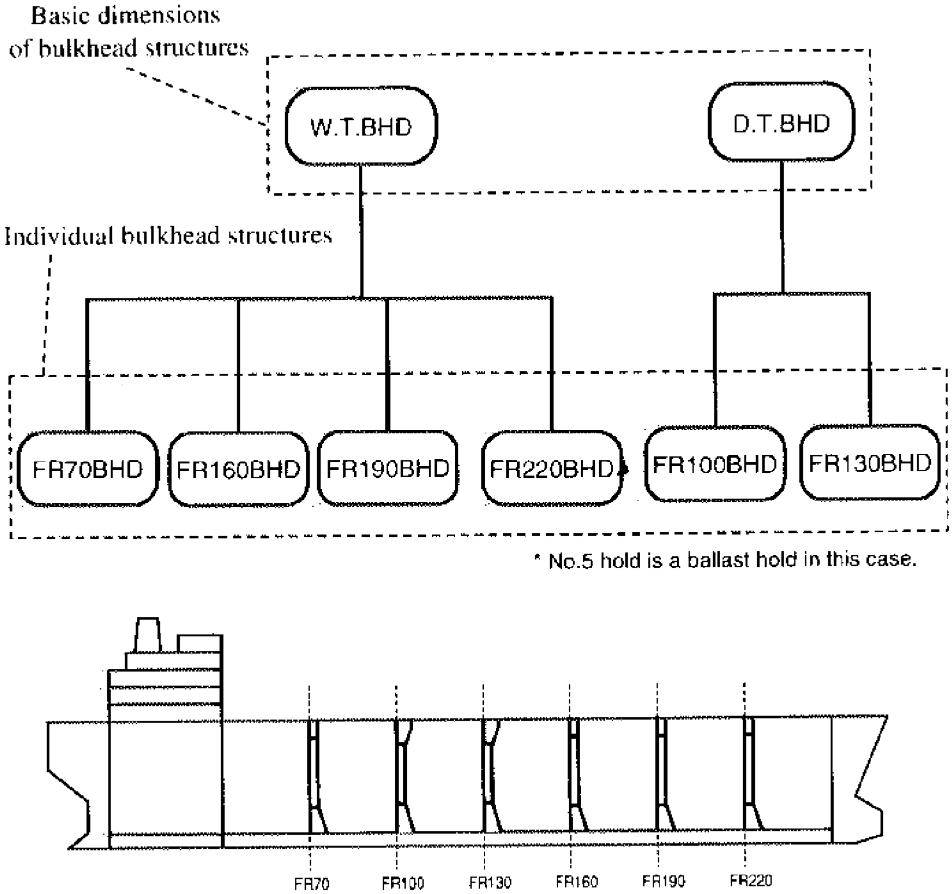


Figure 12. Object structure of corrugated bulkhead structures definitions in the design process

The first version of EFL/S cannot support for representation of these instances. Authors tried to design additional classes to the first version of EFL/S for expressing and handling them. The key features of extension are as follows:

- Add a class that expresses a unit of individual corrugated bulkhead structure
- Add a class that contains the definitions of basic dimensions
- Add a function for rough shape calculation of individual bulkhead structures on the definitions of basic dimensions

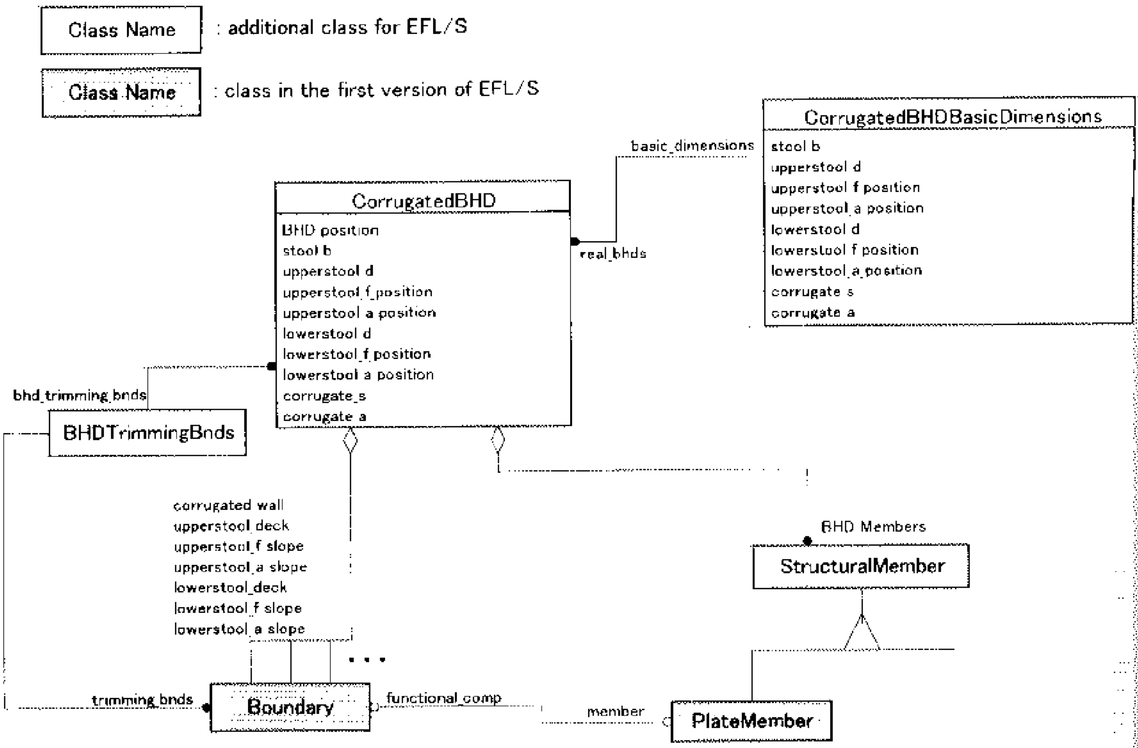


Figure 13. Extension for corrugated bulkhead structures definitions

Figure 13 illustrates the extension for corrugated bulkhead structures definitions. Classes that are expressed by gray boxes are included in the first version of EFL/S and the other classes and relations are extension that is suggested in this paper.

This extension was developed on GPME and the sample program using the additional classes was also created. Figures 14, 15, 16 and 17 show the sample program result browsed with ProdEditor. In this sample program, trough shapes of individual bulkhead structures are automatically calculated on the definitions of basic dimensions, i.e., W.T.BHD and D.T.BHD. Figure 14 shows base shapes of all bulkhead structures. Figure 15 shows trimmed shapes of all bulkhead structures. Figure 16 shows a base shape of one bulkhead structure at frame number 70, i.e., FR70 BHD. And figure 17 does a trimmed shape of FR70 BHD. Those Figures show that the EFL obtained as shown in Figure 13 can support the design nicely.

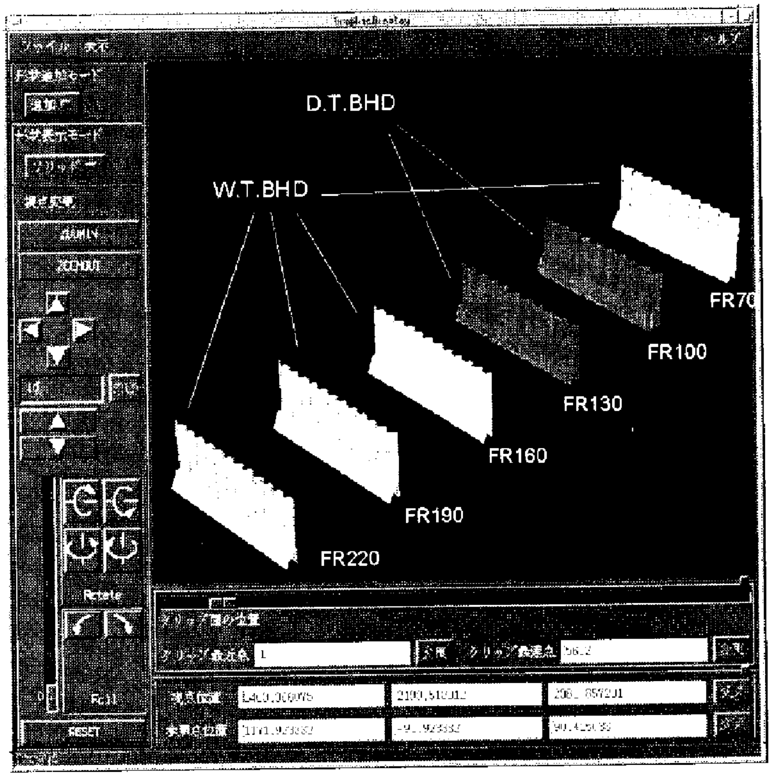


Figure 14. Sample of BHD rough shape calculation 1 (all BHDs, base shape)

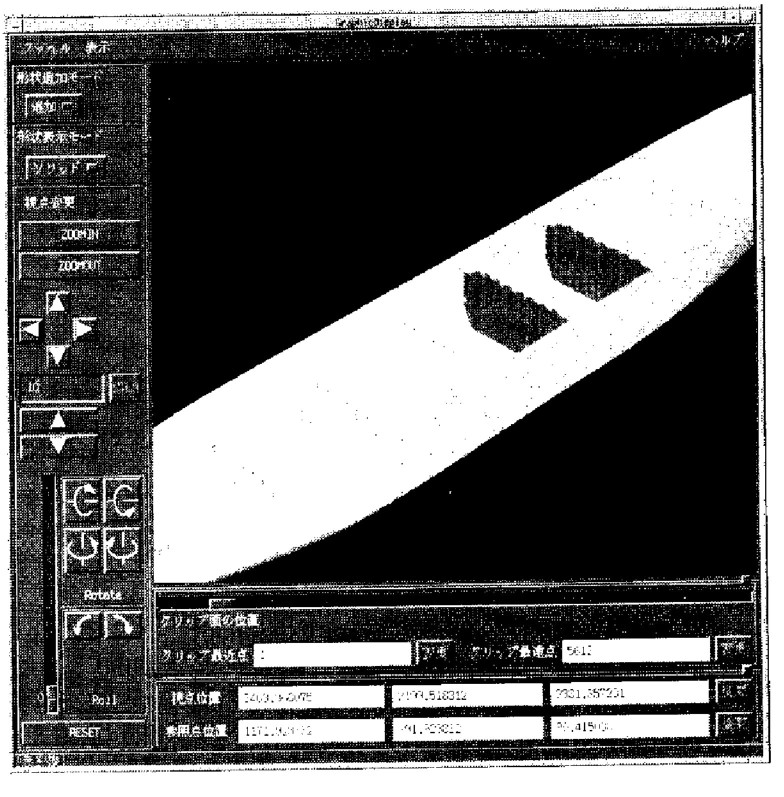


Figure 15. Sample of BHD rough shape calculation 2 (all BHDs, trimmed shape)

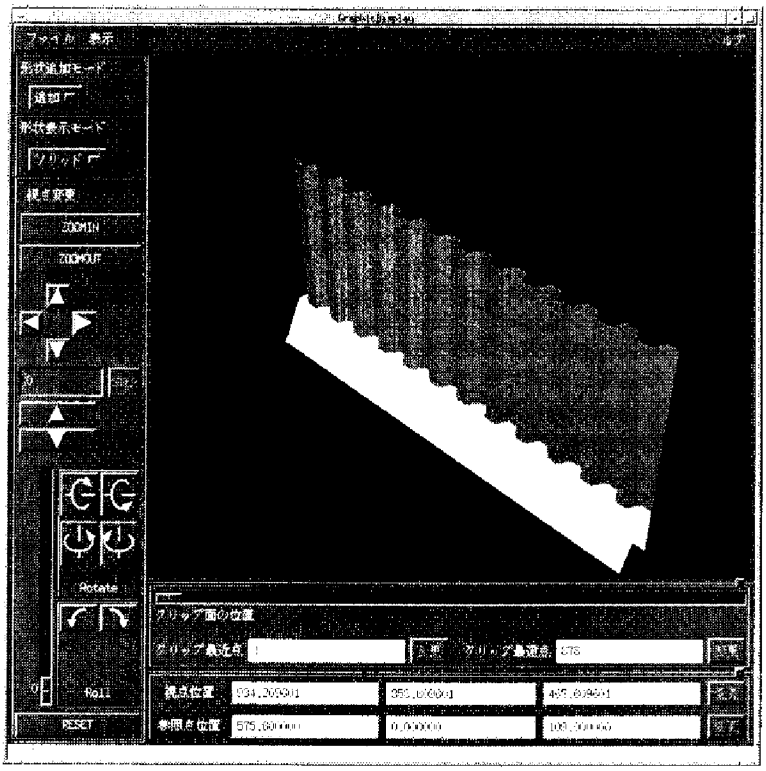


Figure 16. Sample of BHD rough shape calculation 3 (FR70BHD, base shape)

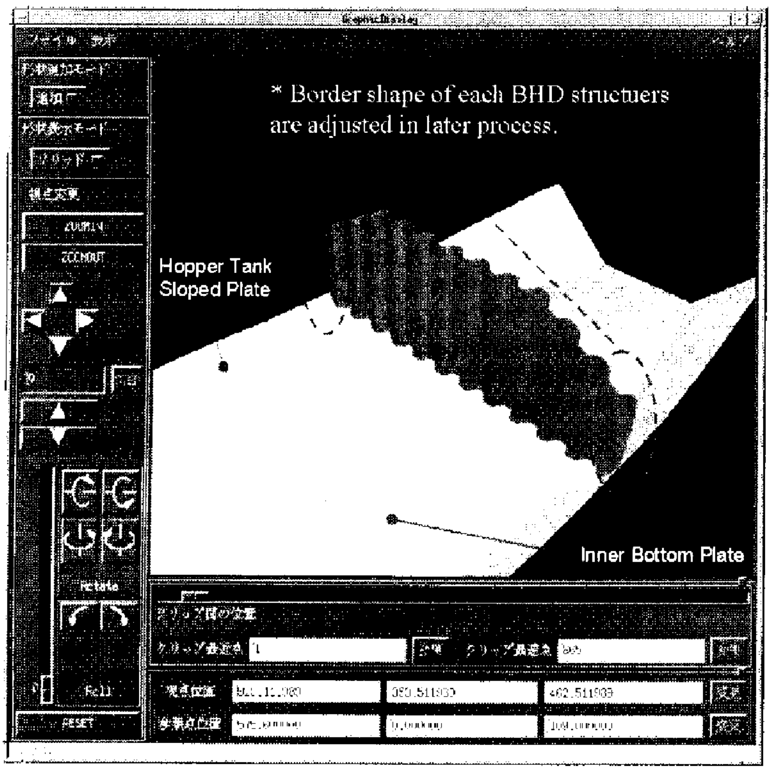


Figure 17. Sample of BHD rough shape calculation 4 (FR70BHD, trimmed shape)

Conclusion

In this research, firstly, Authors tried to clarify intermediate definitions of hull structure in actual design process by the workflow analysis of hull structural basic design in order to develop an effective product model. The workflow analysis based on IDEF3 methodology was made.

And secondly, Authors tried to discuss how to develop additional FL to the first version of EFL/S on GPME based on their analysis. In this paper, Authors exhibited a suggestion and a prototype of extension for bulk carrier transverse corrugated bulkhead structures definition as a concrete example of the additional FL development.

Finally, Authors have obtained following conclusions:

- Design workflow analysis with IDEF3 methodology is effective to clarify intermediate definitions of product in actual design process.
- The workflow analysis results can provide useful information for EFL development in GPME.

Acknowledgement

Authors would like to express their cordial thanks to the Ship and Ocean Foundation and Toyota Caelum Inc. for providing the development environment for prototyping. Authors would also express their gratitude to Mr. Ken Ito from MHI for his help in this work. And the work presented is supported by the Ministry of Education, Science, Sports and Culture under a Grant in Aid for Scientific Research.

References

1. Richard J. Mayer, et al. (1995), "IDEF3 Process Description Capture Method Report", Knowledge Based Systems, Inc.
2. http://www.idef.com/, IDEF Methods Homepage.
3. http://www.kbsi.com/Tools/Prosim.htm, ProSim/Procap Homepage.
4. Ship & Ocean Foundation (1997), "1996 Report on Research and Development of General Product Modeling Environment".
5. Y. Nagase, et al. (1997), "GPME Modeling Methodology and Applications", the proceeding of ICCAS '97 in Yokohama, Japan.
6. H. Kawazoe, et al. (1997), "GPME: Information Modeling and System Integration Environment for Shipbuilding", the proceeding of ICASS '97 in Yokohama, Japan.
7. James Rumbaugh, et al. (1991), "Object-Oriented Modeling and Design", Prentice Hall, Inc.

SHIPBUILDING INFORMATION INFRASTRUCTURE PROJECT (SHIIP)

Tom Rando, Electric Boat Corp., Groton, CT
Lisa McCabe, Electric Boat Corp., Groton, CT

Introduction

The Shipbuilding Information Infrastructure Project (SHIIP) is a three year, $22 million project funded jointly by the Defense Advanced Research Project Agency (DARPA) MARITECH program and by industry. The MARITECH program, begun in 1993, is devoted to improving the competitiveness of the U.S. shipbuilding industry. The SHIIP project is led by Electric Boat Corporation (EBC). The SHIIP team consists of shipbuilders (Bath Iron Works, NASSCO, Atlantic Marine, Avondale, Todd Pacific) and technologists (NIIIP/IBM, Computer Sciences Corp., Deneb Robotics, STEP Tools Inc., Data Access Technologies).

The goal of the SHIIP project has been to support the integration of systems technologies within the U.S. shipbuilding industry. The SHIIP project has sought to identify, develop and deploy standards-based protocols that can be adopted by the U.S. shipbuilding industry as a whole. A standards-based protocol may be endorsed by an international standards body, it may be an emerging standard, or it may be a de facto standard. The salient pre-requisite has been that the protocol be implemented by more than one vendor so that the use of the protocol does not unduly tie a shipbuilder to any one technology provider. To a certain extent, U.S. shipbuilders have been at the mercy of systems technology providers. In its current state the industry does not have the wherewithal to make the technology providers respond to its needs. Rates of production are low. The number of U.S. shipyards is small and compartmentalized; first tier yards have systems technology requirements that are quite different from those of second tier yards. Moreover, the industry is highly competitive and has rarely spoken with one voice.

Recent advances in systems technologies, especially technologies developed for the Internet and based on the Java programming language, have for the first time made it possible for an enterprise to deploy an information infrastructure on which its industry-specific application components can be assembled. The shipbuilding information infrastructure provides the systems integration foundation which formerly was available only within high-priced, proprietary, monolithic software applications. Many of these technology advances were developed and proved out by the National Industrial Information Infrastructure Protocols (NIIIP) project.

SHIIP Deployment

Although it is an infrastructure deployment project, the SHIIP project has chosen a particular shipbuilding business domain in which to focus its efforts. To date most of the systems technology investment in the shipbuilding industry has been in support of design and engineering processes. At Electric Boat there is substantial amount of design information available in digital form. However, there has been very little activity devoted to making this information available to the shipyard work force in ways that would streamline the production processes. While deploying broad-based information infrastructure technology, the SHIIP project has also used the infrastructure to support applications which make more information available to shipyard team leaders, foremen and mechanics.

The SHIIP project is a shipyard initiative. The SHIIP project works closely with EBC's production information systems team. In cycle 2 the SHIIP project selected the Foreman Work Assignment application, which was under consideration by EBC's production systems developers (see Figure 1). The SHIIP team participated in the gathering of functional requirements, which was led by EBC's operations. After the first round of functional requirements was collected, the SHIIP project led the development and demonstration of a prototype deployment of the Foreman Work Assignment application. This work was performed in parallel with the ongoing architecture definition work of the supporting information infrastructure. The demonstration was then used to elicit feedback from the users in order to refine the functional requirements. At this point the application development was turned over to EBC developers for production deployment. Finally, the architecture and selected components have become available for re-use in subsequent production applications.

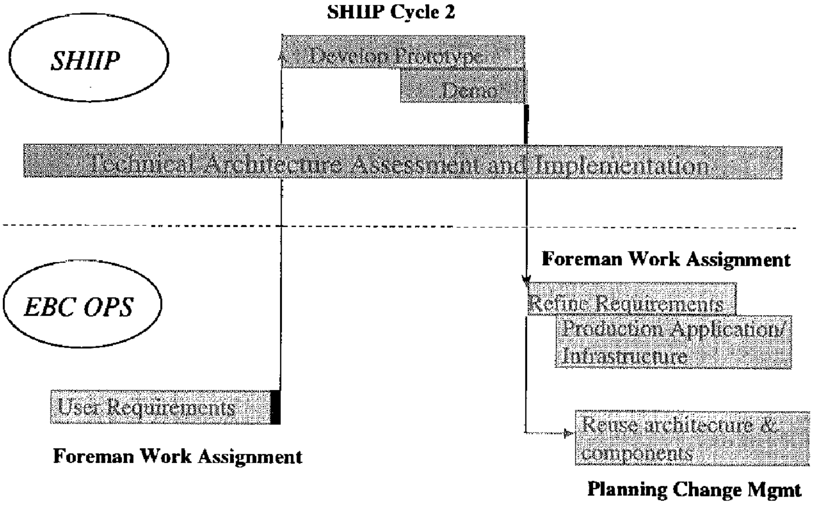


Figure 1 - SHIIP Deployment Process

In some ways, the SHIIP infrastructure is a comprehensive Intranet for shipbuilding as shown in Figure 2. Central to the infrastructure are the communication protocols that connect the SHIIP services and applications. Although several protocols were chosen, they are all practical, Internet protocols, and the objective of the project has been to enable only required services and to find the most appropriate protocol for each service. Table 1 summarizes the protocols selected:

Table 1 - SHIIP Communication Protocols

Protocol	Purpose
Hypertext Transfer Protocol (HTTP)	Web server support for the other protocols
Remote Method Invocation (RMI)	Distributed object computing
Hypertext Markup Language (HTML)	Document content without structured data
eXtensible Markup Language (XML)	Document data intended for human reading as well as computer interpretation
Lightweight Directory Access Protocol (LDAP)	Directories for the location of enterprise data objects
Virtual Reality Markup Language (VRML)	Distributed 3D visualization for moderate sized models

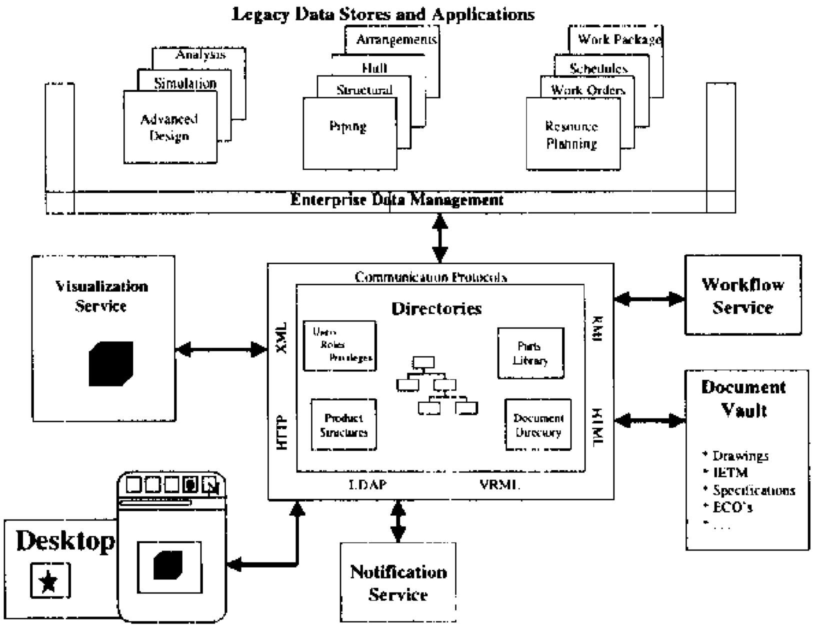


Figure 2 - SHIIP Intranet

Since the SHIIP infrastructure is comprised of distributed services and information objects, a directory service is essential so that users can readily locate objects as needed. These directories can be as straightforward as the directories of users and their roles or as complex as directories that represent the ship's product and parts objects and their relationships.

One of the most significant services of the infrastructure is the enterprise data management service. The purpose of this service is to provide uniform access to enterprise information, which may be stored in legacy datastores as diverse as file systems, hierarchical databases, relational databases or legacy applications. This information also spans a number of business process domains -- such as up-front requirements and concept formulations, physical ship design information, and ship assembly and outfitting information. The selection of technology for the enterprise data management service was the most difficult challenge of the project.

A service that provides users with timely notification of significant business events has been identified in the Integrated Development Environment (IDE) concepts of operations described in a number of recent naval shipbuilding programs. The SHIIP project has designed and demonstrated such a service using software agents.

The user interface into the shipbuilders' Intranet is a key element. The SHIIP project has designed and demonstrated a user interface component called the SHIIP Desktop. The SHIIP project has designed a Desktop with two modes, and both have been demonstrated in the second year of the project. In conjunction with providing the user interface, the SHIIP Desktop supports the strong authentication of users to the enterprise system.

The service most in demand by the shipyard work force is the distributed visualization of the ship product model. The SHIIP project has identified a number of use cases for distributed visualization. These use cases have been formulated in detail, and appropriate visualization technologies have been mapped to each use case. The conclusion of the SHIIP project is that it would be a mistake to attempt to apply any one visualization technology to all use cases.

The remainder of this paper describes how each module of the infrastructure was deployed in the second year of the SHIIP project.

Enterprise Data Management: CORBA or Java?

An enterprise data object (sometimes referred to as a Business Object) is a specialized type of data object. An enterprise data object inherits the features that are needed in order to be used in an enterprise software application: persistence, transactions, security, and load-balancing. What signifies that an object is an enterprise data object is that a large number of users need to have access to the object simultaneously. Such an object is significantly different from an object that has been designed to run in a standalone personal computer application.

If a user today processes an enterprise data object, such as an assembly in the ship's product structure, the expectation is that that assembly will still be available tomorrow (persistence). If a user moves a part from one assembly to another, the expectation is that if the system fails after the part is deleted from the original assembly, the part will not disappear from the product structure (transactions). If a user is not authorized to change parts in the product structure, the expectation is that the system will prohibit that activity (security). And finally, if a hundred users need to access parts at the same time, the expectation is that the system will not make ninety-nine users wait until the first user is finished (load-balancing).

So called "middleware" technology represents the state of the art in enterprise data management today. This technology supports a multi-tiered architecture, in which applications are separated from the underlying database in order to make it possible to keep database management functions independent of application or business logic. One goal is to begin to free application developers from the overwhelming complexities of today's information technologies. There are three prominent middleware technologies:

Microsoft's DCOM, the Object Management Group's (OMG) Common Object Request Broker Architecture (CORBA), and Sun Microsystem's Enterprise JavaBeans (EJB). Because of its commitment to standards-based, non-propriety technologies, the SHIIP project has focused on CORBA and EJB as the two alternatives for evaluation. The objective was to identify the systems requirements of the shipbuilding industry and to determine if either technology is better suited to satisfy these requirements. As part of the evaluation process, the SHIIP project implemented two prototypes of the Foreman Work Assignment objects – one using CORBA business objects and one based on the EJB specification as shown in Figure 3. As a result of the evaluation and of the prototype work, the SHIIP project concluded that EJB is the technology that is best-suited to satisfy the requirements enterprise data management for the shipbuilding industry.

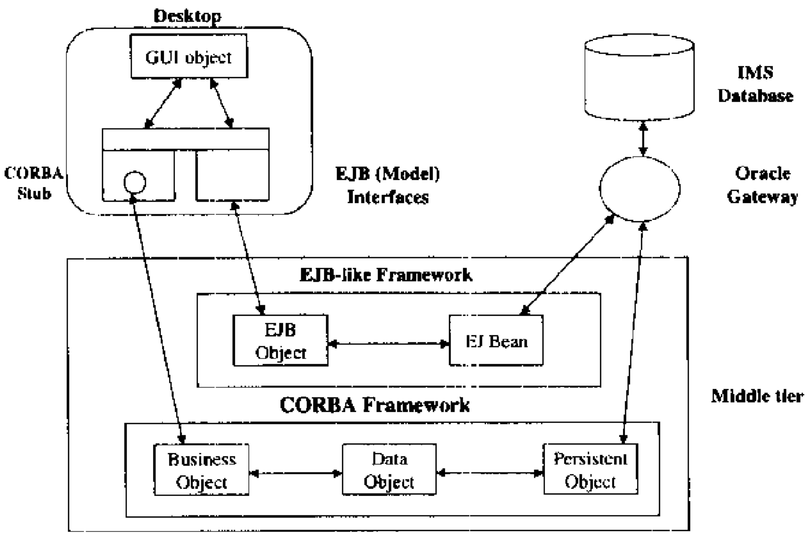


Figure 3 - SHIIP Middle Tier

The goal of the CORBA standard is to support the technical interoperability of distributed objects. At the time the OMG was formed, there was a perception that industry's highest priority was to be able to construct complex systems from software modules written in different programming languages. Moreover, the OMG felt that the requirement for "technology independence" was incumbent on them, that is, in order to be universally applicable, OMG standards should not be tied to any implementation technology. When the SHIIP project began, CORBA was the only available distributed object standard. One objective of the project has been to determine whether CORBA could be used to benefit the shipbuilding industry. Consequently, a substantial amount of the prototyping in the first two years of the SHIIP project was based on CORBA. The results of that prototyping and investigation has produced a number of lessons learned:

- Technology independence, while useful in theory, should not be pursued at the expense of developing working implementations.
- Technical interoperability is not a requirement for the development of an industrial information infrastructure. Supporting applications and software components for the information infrastructure are homogeneous enough to be implemented in a single implementation technology.
- OMG, while advocating separation of services, has produced, in the arena of domain standards, wrappers for monolithic applications.
- OMG has been hampered by the inability to reach timely consensus.

The Enterprise JavaBeans specification was published in April, 1998 – near the end of the second year of the SHIIP project. Java technology has a different prime requirement than CORBA technology. Java is dedicated to portability, the notion of software development whose goal is "write once, run anywhere." Our experience has been that this capability is much more valuable to the shipbuilding industry than technical interoperability. Across alliances of shipyards, throughout the maritime supply chain, and within a single shipyard, it is very important to have access to software that can be run on any hardware/operating system platform available. Java provides a universal computing environment.

In addition, Java and Enterprise JavaBeans brought along some unexpected benefits that quickly persuaded us to turn to this technology to provide the foundation of the shipbuilding information infrastructure. Java provides an implementation technology. Java is not a conceptual object definition language. In fact, Java incorporated the strengths and purged the weaknesses of the C++ programming language with such success that Java developers are consistently more productive than C++ developers. These productivity gains are not only widely reported but were also noticed within the SHIIP project. Sun Microsystems has also been very successful at delivering timely specifications (and implementations). Finally, Java comes at such a low cost of entry that it promises to be adopted by second tier as well as first tier shipyards.

In this development cycle, the project prototyped two middleware technologies, CORBA business objects and EJB objects. The business object infrastructure was used to implement the Foreman Work Assignment application. With this application, a foreman can access EBC's MRP system through the SHIIP Desktop in order to assign work items to members of his crew. Ultimately, mechanics could use the system to designate their own work plans. The first challenge was to retrieve the Work Order data from the legacy IMS database on the mainframe. An Oracle procedural gateway was used for the prototype. Two teams of developers produced Work Order business objects for the middle tier application. One team modeled the objects using OMG's Interface Definition Language(IDL), and the other modeled the objects using Java and the EJB specification. Since there was no commercial EJB server available at the time, SHIIP developers simulated the EJB environment by adhering strictly to the EJB specification. On the client side, the EJB specification was also used to define the remote interfaces to the Work Order objects. Even the CORBA client stubs were wrapped with these interfaces. In this way it was possible to develop one set of user interface components that supported both prototypes.

Notification Service

Figure 4 illustrates an overview of the SHIIP Notification Service. This includes work that has been completed as well as work that is scheduled to be done in the final year of the project (which is indicated by the shaded box). The foundation of the Notification Service is the delivery system. Notification of interesting business events is performed by an asynchronous messaging system. The message conveys information regarding the occurrence of a business event to a user. The messaging system must be portable enough to deliver messages to a wide range of clients. The first prototype was based on the CORBA Event Service. This approach depends on the availability of CORBA clients throughout the shipyard. The next prototype will be implemented using the Java Messaging Service (JMS). Both the CORBA Event Service and JMS implement a publish-and-subscribe design pattern. After a business event is generated, it is turned over the messaging system. The messaging system permits users to subscribe for types of events. When events of this type are received at the messaging system, they are delivered to interested users. If there are no subscribes for a message, it is not delivered. The final delivery to the user will be selected by the user and can range from an email message, to a pager message, to a notification on the Desktop.

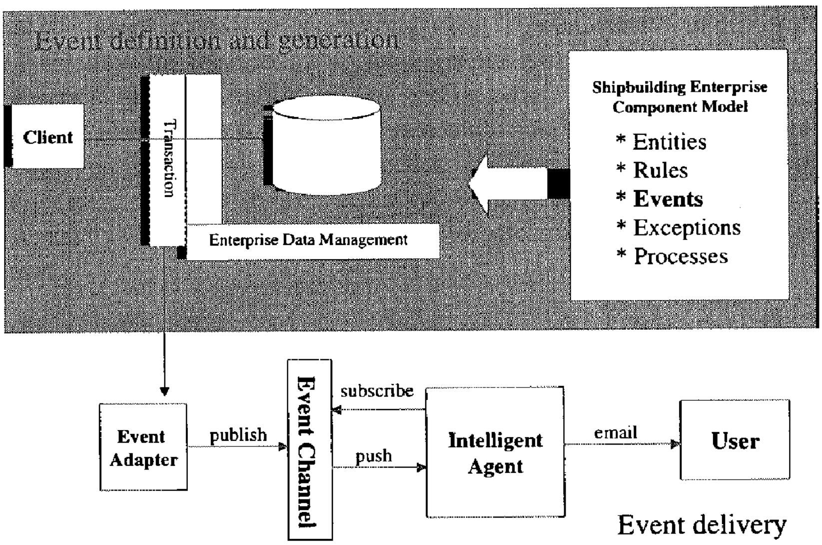


Figure 4 - SHIIP Notification Service

One innovation of the SHIIP Notification Service is the use of software agents. A software agent is a small software application that can be highly personalized to meet the needs of an individual user. Software agents are often rule-based and are intended to play the role of "agents" for their users. The enterprise notification service is an especially apt application for software agents. An automatic notification system may threaten to deluge users with unwanted messages. The ability to subscribe only for designated types of events may not be sufficient to protect a user from such a deluge. The notification agent can be configured by each user to provide a much more intelligent filter for potential messages. Through a simple interface, the user can specify not only what type of events are of interest but also under what conditions the user wishes to be notified or not be notified of the event. The agent makes these decisions based on rules that have been added to the agent and based on the attributes attached to the event. For example, a foreman may request to be notified if a stop work order is attached to one of his jobs – but only if that job is currently on the schedule horizon.

The next version of the prototype will focus on the event definition and event generation portions of the service. Until now, events have been generated by polling the database for significant state changes. This approach is non-obtrusive but may impose burdensome overhead for a database with a high transaction volume. In the next cycle, the Notification Service will become more integrated. The design and prototyping of the Notification Service has demonstrated a need for an explicit model of the events that are of interest throughout the shipbuilding process. Following traditional practice, the SHIIP project has developed object models (data dictionaries) for the entities and relationships that are contained in the enterprise data management system. Object definition languages, however, do not typically provide facilities for the modeling of Business Events. The SHIIP project has begun to define a methodology for a formal definition of Business Events.

The inclusion of Business Events as part of the shipbuilding object model suggests that the enterprise data management system should be more closely integrated with the Notification Service. The use of JMS in the next version of the Notification Service prototype is designed to accomplish that goal. With this approach it will be possible to implement transactions that encompass both database updates and resulting events. The transaction will guarantee that if an update is requested but fails, the message will not be delivered.

SHIIP Desktop

One responsibility of the SHIIP Desktop is to authenticate the user, that is, to guarantee to the enterprise systems that the user is really who he claims to be. One typical method of authentication is to provide a user id and password. This approach tends to become inconvenient to users when they are suddenly given access to scores of services through the information infrastructure. Inevitably, different services end up using different user id's; and as services demand new passwords, it becomes challenging to keep track of which passwords are currently in effect for which services. Another method relies on public key/private key technology. With this method, each user is provided with a private key (often on a floppy disk), the corresponding public key is managed in the enterprise person directory. This method becomes inconvenient when the user needs to move about the company. In this scenario, the user must bring his private key file with him and install it on each client machine that he uses.

The SHIIP Desktop has prototyped a third method, based on iButton technology. The iButton is button-sized Java computer. It can connect through an inexpensive adapter to a serial or parallel port on a PC. It is powered by its host computer. The iButton is capable of storing a user's private key, distinguished name, and the unique identifier of the iButton. In addition, it can generate time stamps. With these capabilities, the iButton can generate a string of characters that represent the user's "credentials". Figure 5 illustrates the authentication interactions that have been prototyped by the SHIIP project.

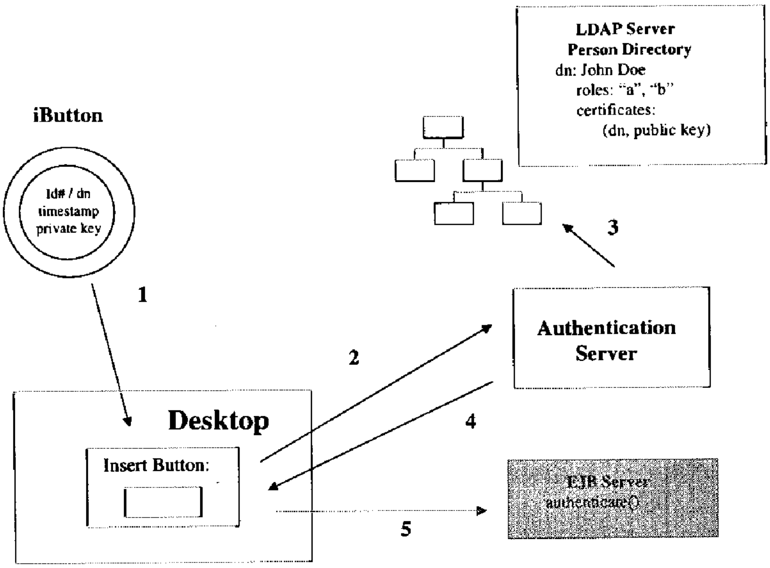


Figure 5 – Authentication

When the user starts the SHIIP desktop, an authentication window prompts the user to attach his iButton to the client computer. The authentication window drives an authentication applet, which asks the iButton to provide the user's credentials. The iButton generates a string with the user's distinguished name, the ring id, and a time stamp. The iButton encrypts part of this string using the user's private key; this represents an authentication message "signed" by the user. The authentication applet sends the credentials to the authentication server. The authentication server is responsible for determining whether the user is, in fact, who he claims to be. The authentication server relies on the enterprise Person Directory to make this determination. The Person Directory is implemented using the LDAP protocol and an LDAP server. LDAP is the Internet protocol for directory services. The directory entry for the user is located in the directory based on user's distinguished name. The user's directory entry manages the user's public key, which is sent back to the authentication server. If the authentication server can successfully decrypt the credentials with the user's public key, then the user is strongly authenticated. The authentication server allows the user to access the Desktop.

The LDAP server is also the service that manages the roles that the user is authorized to play in the enterprise. This service is the foundation for managing access to enterprise services based on role. Access control within the information infrastructure will be designed and prototyped in the next year of the project. Also scheduled for the next prototype is the integration of the strong authentication service with a commercial off-the-shelf EJB server.

The SHIIP Desktop is illustrated in Figure 6. As the figure shows, the SHIIP Desktop is designed to work in conjunction with a Web browser. The SHIIP Desktop is implemented using JavaBeans technology. The Desktop is divided into two regions. The Type Manager Bar houses icons which control operations applicable to "types" of objects, such as creating, deleting, and locating. For example, the Work Order (WO) type manager would be used to create a new Work Order. The palette region houses business object instances, icons, and lists of icons. The Work Order user interface component is used to present the contents of Work Order to the user. Every business object has two possible representations on the Desktop – as a complete user interface and as an icon. Each icon represents a business object. Double clicking the icon causes the Work Order user interface to open; dragging and dropping the icon adds a Work Order to some other Desktop object. The Desktop also manages lists of business objects, which represent results sets from some query against the system.

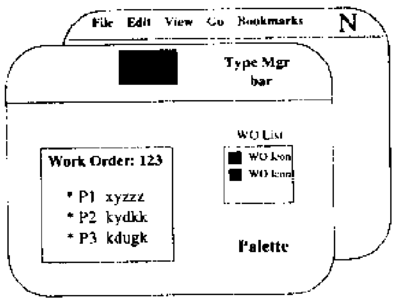


Figure 6 - SHIIP Desktop

The Desktop presents a familiar graphical user interface in which objects appear to be resident in the user's personal computer but are, in fact, dispersed across the enterprise. When a user drags a file icon to trash can on a stand alone PC, the file itself is on the PC, and so is trash service which deletes the file. When a user drags a Work Order icon to the trash bin on the Desktop, the Work Order may actually be an object that resides on an enterprise server and is stored in a mainframe database, and the same is true of the service which deletes the Work Order. The Desktop extends the range of the personal computer to provide the authorized user with access and control over every component in the enterprise's information infrastructure.

In order for the Desktop to be an effective user interface into the enterprise information infrastructure, there are a number of technical challenges or requirements that must be satisfied. The Desktop must, first, provide an interface to distributed business objects. It must be practicable to deploy across thousands of personal computers. It must enable the co-operability of business objects from a variety of application domains. It must be able to represent objects implemented across the entire range of enterprise component and Web technologies. Finally, the Desktop must be customizable to suit the requirements and preferences of the entire population of end users.

As an element of the enterprise information infrastructure, the Desktop will potentially be deployed on hundreds, possibly thousands, of computers. In a production environment the result is a substantial configuration management burden. Whenever there is a new version of client-side code, every client must be re-configured. If the reconfiguration process should ever falter, the result is that there will be client computers in the field with incompatible combinations of application code. Some combinations may seem to work for some scenarios but mysteriously fail for other scenarios. Some combinations may not work at all. In any case, the potential disruption to production users is a nightmare. In the virtual enterprise, where resources are managed by entirely different support staffs, the problem is even more severe.

The SHIIP project has deployed two styles of Desktop and has attempted to document the architectural considerations that identify the scenarios in which one style is preferable to the other. The second style of Desktop has been called the lightweight Desktop. It is illustrated in Figure 7. One use case that was identified as part of the SHIIP project was delivery of work packages to the mechanics through a browser. The idea was that this approach would streamline the current process in which voluminous printed documents represented the only format for trade work packages. There were many complaints about the paper-based process:

- it was not possible to print only the page or two at a time that were needed;
- it was difficult to trace the cross-references throughout the many pages;
- the volume of paper led to inefficiencies;
- it was time-consuming to track down reference documents.

This use case presented requirements that were quite different from the Foreman Work Assignment application. The application was largely read-only; at most the user would need to fill in a form or two. Use of the system would be more intermittent and users would expect a quicker response time for these short-lived sessions. The process could be further streamlined if the client was a handheld or portable computer. For the Electronic Work Package task, the Lightweight Desktop was used. The Lightweight Desktop is based on the Web browser. Only HTML pages are delivered to the client. There is no requirement for the client to download or process sophisticated JavaBeans applets.

Nevertheless, it was possible to reuse significant portions of the Foreman Work Assignment objects for the Electronic Work Package application. As illustrated in Figure 7, the Electronic Work Package uses the EJB objects related to Work Orders. Work Orders are stored in an IMS database on a mainframe. Work Order data is accessed through an Oracle procedural gateway in order to populate the EJB business objects. These objects could be accessed directly by the Desktop client. However, in the Electronic Work Package application these business objects are accessed by a server-side Java program known as a servlet. The servlet is a client to the business objects, and it translates Work Order data into HTML forms, which are sent to the Lightweight Desktop. In the next development cycle, XML will be used in place of HTML. XML is an Internet protocol that has the added feature that it can represent data that can be interpreted by computer applications.

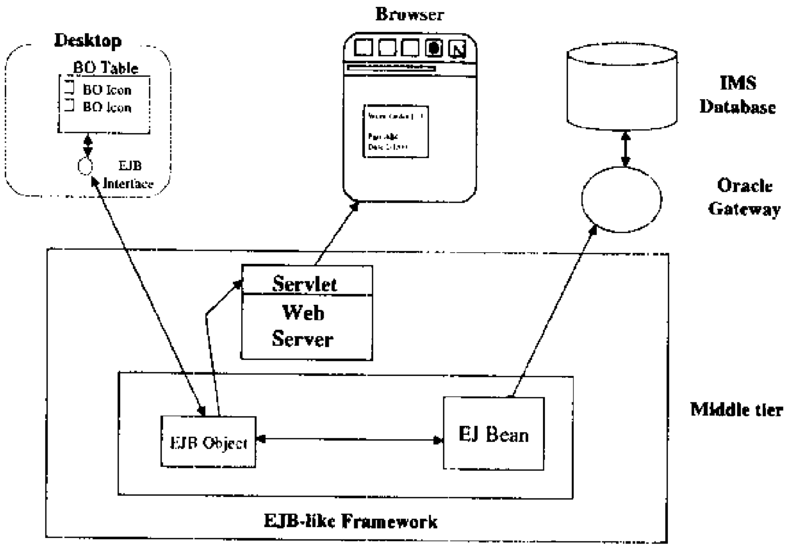


Figure 7 - Lightweight Desktop

Distributed product visualization

One of the most promising opportunities for increasing shipyard efficiency with improved information sharing is distributed product visualization. The SHIIP project conducted numerous workshops with shipyard personnel in order to identify the shipyard's visualization requirements. From these meetings it became clear that the shipyard's visualization requirements were not exactly the same as the visualization requirements of the design and engineering processes. Yet most of the visualization research and development among shipbuilders has been driven by design and engineering organizations. The shipyard visualization requirements can be summarized as follows.

Efficient display of large models

In order to aid in the installation and outfitting processes, it is necessary to provide visualizations up to the size of a work area. A ship may be the most difficult of all industrial visualization challenges, being comprised of as many as ten million parts. The efficient display of large models entails two aspects: time to load the model and response time when viewing.

Ability to extract a portion of the ship for viewing

In today's shipbuilding design applications, the ship is represented as CAD models which are organized in accordance to the needs of the design work force. This organization cannot possibly suit the needs of all production and test processes. The shipbuilding visualization system must be capable of representing the ship in a number views for a number of different users. Some of these views include:

- planner's view by schedule activity,
- operations' view by work order by hull,
- operations' view by work area,
- testing's view by system by test section.

In order to support the diversity of required views, the visualization must be capable of providing an arbitrary visualization based on query that can designate items at the part level.

Guarantee that the visualization accurately reflects the design configuration baseline

Because of the duration of the ship's design life cycle and the inevitability of changes to follow-on ships, there are often several versions of the design of the ship available at any time. Different versions are applicable to different hulls. In practice, the visualization service will likely be loosely coupled with the design databases. In any case, the system must guarantee that when the shipyard worker requests a visualization, it corresponds to the latest version in the design configuration baseline.

Enable the user to navigate from the visualization to the appropriate design data

The visual representation of the ship is very valuable information; however, there is other valuable information available in the design and engineering databases. The system must provide links so that the user can retrieve design and engineering data for each part in the visualization.

Visualization Use Cases

Three visualization use cases were identified by the SHIP project, and solutions were prototyped in this development cycle. The scenarios are illustrated in Figure 8. The first, and most demanding use case, involves the production startup team meeting. The idea is that the team leader or foremen uses the visualization service to conduct a briefing with the shipyard work force. In this use case, it is necessary to visualize an entire work area of the ship and to highlight work items that are scheduled for installation. It should be possible to use this meeting to schedule the sequence of imminent installation steps and to illustrate possible interferences. The design for this service depends on a dedicated visualization server that is capable of displaying thousands of pieces at a time. In addition, the SHIP prototype employed a cache (library) of visualization data that had been extracted from the company's CAD models. (In the next development cycle, the SHIP project will focus on the synchronization of the visualization and CAD data.)

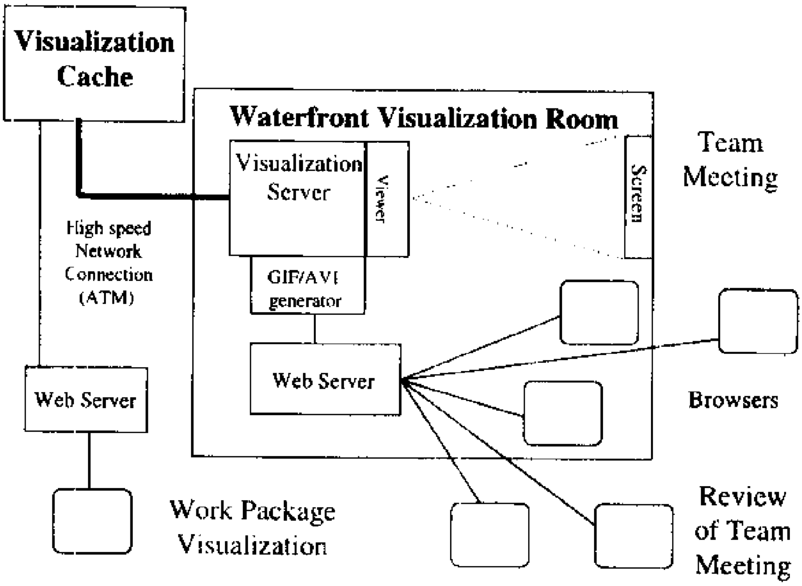


Figure 8 - Shipyard Visualization Use Cases

A second use case involves the situation in which a user needs to review the contents of a team meeting visualization session from a remote client PC. In the SHIIP prototype this was accomplished by enhancing the visualization server with the capability of freezing each view of the session as a two-dimensional image. This image is sent to a client PC. Since the image format is GIF, only a browser is needed to display it. The image contains controls that enable the user to manipulate the view in three dimensions. In this way, the browser provides all the view navigation capabilities of a 3D viewer. This approach is especially promising for very large models because the size of the image that moves over the network is independent of the size or complexity of the model. All geometric processing takes place on the visualization server. An entire work area can be displayed on a laptop.

The third use case entails the visualization of a single work package by a mechanic responsible for performing the work. This use case differs from the other two in that the size of the typical visualization is substantially smaller than the work area visualization. In fact, for a work package visualization, today's technology is a adequate for client-side viewing of the 3D model. In this scenario, the complete 3D visualization of the work package is downloaded to the client. The SHIIP has prototyped the use of the Virtual Reality Modeling Language (VRML2.0) for work package visualization. There are some benefits that accrue from the use of VRML (as compared to the use of the 2D images). VRML viewers are based on Internet standards and are widely available. VRML 2.0 supports the interaction of the visualization session with other Java processes on the client. Finally, the client-side processing of model visualization contributes to the load-balancing of the enterprise's computing resources. If all geometric processing is performed on the visualization servers, these machines could be overburdened. If the client can be used to perform the visualization processing for an entire category of visualizations, more server resources will be freed up. There is no additional imposition on client resources because the same clients would be used for the 2D GIF-style visualization. The SHIIP visualization systems have been architected so that the same underlying visualization data can be used in either scenario – making it much easier to switch from image to model visualization for the purpose of the balancing visualization load among resources.

Conclusion

At the end of its second year the SHIIP project has successfully demonstrated the use of several advanced technologies for the shipbuilding industry. The work of the SHIIP project has formed the basis of the Advanced Shipbuilding Enterprise's (ASE) Strategic Investment Plan for Systems Technology. The ASE will be the primary vehicle for collaborative research and development for U.S. shipbuilding. As a result, the SHIIP project has provided a foundation on which systems technologies can be applied to improve the competitiveness and the quality of the U.S. shipbuilding industry into the twenty-first century.

BREAKING THE SHIP PRODUCT MODEL INFORMATION BOTTLENECK

Dan Wooley, Newport New Shipbuilding, Newport News, Va.
Mark Pettitt, PE, Intergraph Corporation, Huntsville, Al.

Abstract

Today's ship modeling systems require extensive formal training and months of intensive use before an experienced ship designer becomes fully proficient. A casual user (a person that only needs access to the model a couple times a week) is forced to rely on an expert user to get to the model data. This is an inconvenience to both the casual user and the expert who is distracted from primary production functions. In addition to the difficult user interface, today's ship modeling systems require a very expensive workstation. These two facts limit access to the rich set of data contained in the ship model. The next generation ship product modeling system, Global Shipbuilding Computer Aided Design (GSCAD), is being designed to dramatically effect the two problems described above. First, it is being designed to run on a Windows NT personal computer. This will reduce the cost of the workstation required to access the ship model and greatly increase the number of workstations able to access the model. Second, it is being design with a very familiar (Microsoft Windows) and intuitive user interface. This will reduce initial training costs and enable casual users to access the ship model without the aid of an expert user.

This paper examines the key technical innovations in computer hardware and in software User Interface design that combine to break the ship product model bottleneck and describes the potential impact on ship design costs and schedules.

Product Model Definition

The term "Product Model" over the last few years has been over used. It has almost as many meanings as there are people that using the term. For context of this paper the term product model means a complete description of a product. For a ship product model this includes whole ship, system and sub system functional requirements and behavior, complete physical description, complete fabrication and assembly information, and complete operational and maintenance information.

People sometimes think that the ship product model concept is a new idea, but the product model has always existed, usually in the form of paper documents. Only since the development of computer tools that can capture the ship representation have users dreamed of a product modeling system that can contain the complete ship product model.

Current Capabilities

Today's ship product modeling systems, to varying degrees, are capable of capturing a large portion of the ship product model. But tons of paper documents are still being creating because of the current limited access to the digital ship product model. There are several important barriers that today's product modeling systems present that prevent the necessary broad access to the data.

The ease of use of today's system is a huge barrier. Today's product modeling systems require weeks of formal training and months of constant use before a user is considered productive at creating and accessing the ship product model data. As a result of this learning curve, only a few highly trained and very specialized users have direct access the product model data. Thousands of other potential users involved in the design, planning, management, construction, operation and maintenance of a ship are forced to rely on the documentation, mostly on paper, generated by these product modeling specialist.

The cost and specialized nature of the current product modeling system's computer hardware is another major barrier to broad access to the ship product data. Most of today's product modeling systems only run on powerful UNIX workstations. These workstations are expensive to acquire and maintain. The number of workstations that a project can afford is therefore much smaller than the number of people that need access to the product model. This ratio can be as high as 1 workstation to 50 people.

The previous generation product modeling systems were even worse than the current generation in terms of both ease of use and number of workstations. For example, in the mid 1980's Newport New Shipbuilding was using the product modeling system VIVID® to design the Seawolf submarine. VIVID's® user interface, although state of the art at the time, was very difficult to learn. The formal training required 2 or 3 weeks, depending on the user's specialization, and the user required a minimum of 6 months of constant experience with the system to gain full productivity. VIVID®, at that time, ran on an IBM mainframe and, based on today's standards, used very expensive workstations. Only 52 workstation were available. These were used three shifts a day and there was significant un-met demand for access to the product model data. Expanding the number of workstations, however, was impractical, due to the cost of the additional workstations and increased the main frame capacity.

Future Vision

It may be several computer system generations before we develop product modeling systems that are as easy to use as paper documents. However, significant progress is being made. The next generation ship product modeling tools will significantly reduce the product model access barriers. With the next generation product modeling systems the ease of use will allow user to focus on the product definition, with less time devoted to how the computer tool is used. The learning curve for the next generation product modeling tools will be a fraction of the learning curve of today's product modeling tools. The GSCAD learning curve for an experienced ship designer is expected to be 5 to 6 weeks as compared to 6 months with today's tools. Figure 1 illustrates this difference. This will allow engineers to develop the design directly in the product model, as opposed to today's paradigm where the engineer sketches a design on paper and a product modeling specialist builds the model.

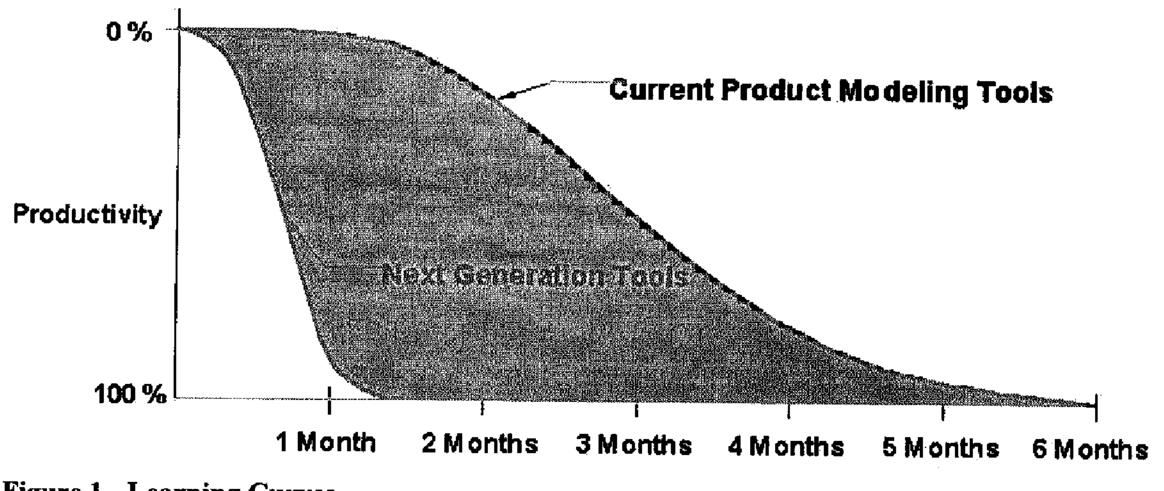


Figure 1 - Learning Curves

The dramatic improvement in ease of use of the next generation product modeling systems will reduce one of the primary product model access barriers. It will enable engineers to develop ship designs; managers to review and monitor design progress; planners to develop build strategies; customers and suppliers to participate in design decisions; logisticians to develop operation and support information; and shipbuilders to extract manufacturing data directly in the product modeling rather than relaying on the product modeling specialist to create documents.

The productivity improvement would be comparable to the dramatic impact that the personal computer and office automation tools have had on the way we do business. In the early 1980's word processing was done by a group of dedicated data entry personnel. Documents were first hand written and then submitted for processing. A finished document was returned on paper. The document was then marked up and submitted for correction by the data entry personnel and a finished paper document was the result. Today documents are developed directly by the authors and distributed as digital files over the Internet.

The next generation product modeling tools will have a similar impact on the ship design, planning, construction and support processes. Today's ship design process is not much different from the 1980 word processing example. The ship design is developed on paper or multiple independent analysis tools. The design is then communicated to the product modeling specialist who creates the ship model. Drawings are then generated and marked up by the engineers. The product model is then updated and final drawings generated. With the next generation tools the engineers will create their designs directly in the product modeling tool, product modeling specialist will add detail, reviewer, customers, and manufacturing personnel will all access the product model data directly. The need for drawings will be greatly reduced. In many cases, product model "snap shots" will be used in place of formally configuration managed drawings.

The hardware requirements of the next generation product modeling will tear down the other major product model access barrier. Today's product modeling tools all require expensive UNIX workstations and separate networks. The next generation product modeling tools will execute on the same personal computer that the user accesses office automation and internet access tools. No longer will projects be faced with the 50 to 1 users-to-workstation ratio. Instead, 1 to 1 ratios will be the norm. The requirement to acquire and maintain separate network software for the product-modeling tool will no longer exist. All this results in lower equipment and software costs but more importantly it results in improved access to the ship product model. This improved access will enable major process improvements that result in lower costs and shorted schedules.

As the computing power of personal computers continues to go up and the price and physical size continue to come down further advances in accessibility can be expected. One example already being contemplated is wearable computers for onboard maintenance and configuration control.

Global Shipbuilding Computer Aided Design (GSCAD)

The Global Shipbuilding Computer Aided Design (GSCAD) product modeling system is being developed by Intergraph and a international team of world class shipbuilders. GSCAD is being developed using the latest Microsoft technology. The Microsoft Windows user interface is very familiar to any Microsoft Office user. The Microsoft technology allows the system to execute on an inexpensive NT network using low cost personal computers. The minimum hardware requirements have not been determined at the time of writing this paper. However, they are expected to be within the range of office automation personal computers.

The user interface for GSCAD has been designed to guide the user through the tasks of creating, modifying and managing the ship product model. The key functional concepts are; task focus, intelligent commands, flexible access to product data, and common tools across task environments.

Task Focus

At the very heart of the user interface design are the Task environments. Each Task environment is a separate application designed to support a specific task. There are currently over 60 Task environments designed for GSCAD. Figure 2 shows the GSCAD main menu "Tasks" option. This option is used to switch between Task environments. The user simply selects the Task environment desired and the system opens the new environment keeping the views and workspace content the same. One of the over 60 GSCAD Task environments, for example, is the Molded Form Task environment. Within the Molded Form Task environment the user creates and modifies plate systems (Decks, Bulkheads, Shell), stiffener profile systems, beam profile systems, openings, and seams. Pipe is routed in the Route Pipe Task Environment. The Task Environments are not just a command hierarchy. The software is designed to permit the different Task Environments to be enhanced at a different pace. The application is not a single massive whole, but a sum of many component parts that work together through Component Object Model (COM) standards. The careful division of functionality into environments is based on a deep understanding of how ships are designed in the real world.

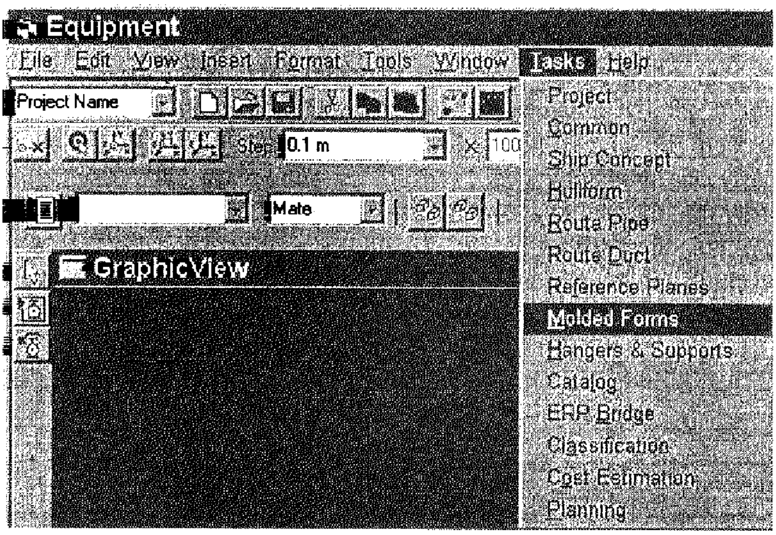


Figure 2 - Tasks Main Menu Pulldown

Intelligent Commands

Each command in a Task Environment presents the user with a ribbon bar that guides the user through the creation or modification process. The ribbon bars follow the concept of layered exposure of complexity. The goal is to present only what the user needs to see and no extraneous commands or controls. For example, the Molded Form Task environment allows the user to create plate systems such as decks, bulkheads, floors, platforms, girders and the shell. The surface definition of a plate system can be as simple as a single plane or as complex as a skinned surface. The surface geometry complexity therefore determined the organization of the plate system commands. Each surface type has its own ribbon bar. This way a user creating a simple single plane plate system does not have to wade through all the commands and options presented to a user creating a non-linear projected surface.

The GSCAD user interface also subdivides a given lower-level task as represented by a command into steps and then presents only the controls needed for the current step. This simplifies the display to show only what the user needs to see as he accomplishes the task. SmartStep is the term given to this step-oriented interface. Each GSCAD ribbon bar reads from left to right and begins with a series of SmartSteps. Each SmartStep changes the ribbon bar to show the commands and options for the specific step. The SmartSteps for the planar plate system command is used to illustrate this point.

Figure 3 is the ribbon bar for creating and modifying a planar plate system. The four SmartSteps are Settings, Plane Definition, Boundary Definition and Finish. Each SmartStep changes the remainder of the ribbon bar to present the user with commands and options needed for that step. The first SmartStep in every GSCAD ribbon bar is the Setting step. This step presents a multi tab form that allows the user to modify the non-geometric properties of the object being created or modified.

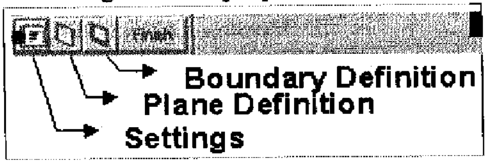


Figure 3- Molded Form - Planar Plate System Ribbon Bar

Figure 4 is the ribbon bar displayed when the Plane Definition SmartStep is selected. It presents five additional commands that can be used to define the single plane. Each of the additional command will display additional input options and user prompts.

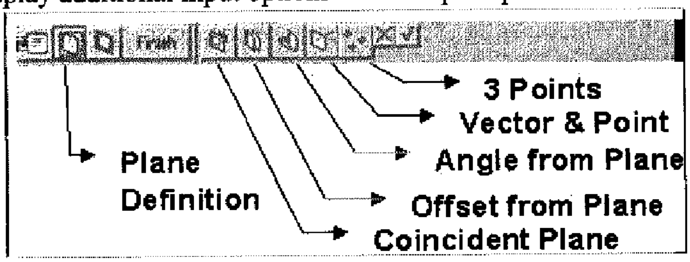


Figure 4 - Plane Definition SmartStep Ribbon Bar

Figure 5 is the ribbon bar displayed when the Boundary Definition SmartStep is selected. It presents the user with two additional commands that allow the user to select bounding objects by name or to define a bounding curve using a 2D sketching metaphor. Both of these are optional boundary definition methods. The default method and prompt is to select the bounding objects in the graphic views.

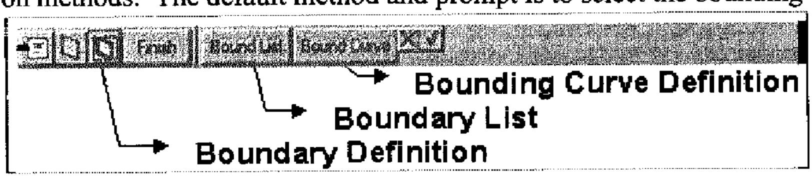


Figure 5 Boundary Definition SmartStep Ribbon Bar

Once an object has been created, SmartSteps are also used to modify the object. The modification process begins with the user selecting an object in the graphical view or by name in the workspace navigator. An edit ribbon bar, very similar to the create ribbon bar, is presented to the user. The user selects the SmartStep that deals with the aspect of the object that needs to be changed. The system then presents the existing object's current definition for the selected step. The user can then modify the definition. For example, suppose that a transverse bulkhead plate system existed that was defined with a single plane on frame 25 and a boundary consisting of the shell plate system and the main deck plate system. If this plate system is selected an edit ribbon bar is displayed that looks just like the planar plate system create ribbon bar described above. If the user selects the plane definition step, the existing plane definition is highlighted. In this case the transverse plane on frame 25 would be highlighted. If the user selects a new reference plane, for example frame 30, then the transverse plane on frame 30 is highlighted. If the user then selects the finish step, the transverse bulkhead plate system is moved to frame 30 and its boundary is recomputed and any plate systems or profile systems that are bounded by this plate system are recomputed.

Flexible access to product data

The ship product model database is huge. Users need to view different aspects of the product model depending on their assigned task. Traditional file-based CAD models of the ship data have proven too inflexible for the many different design needs. The files (rigid collections of design data) bring either too much or too little information to the user's workstation. The GSCAD product model database design permits users to load only that portion of the model that is of interest for the task at hand. The user can select to load information by a very flexible query that can include system, assembly, and design responsibility assignments. Only the visual representations of the selected objects are initially loaded to the PC. As the user selects objects to edit, the GSCAD system automatically retrieves other related data and objects from the database as necessary to support the edit. In ship product model a change in one object may cause another object to change. The GSCAD associative system's automatic data loading and change propagation makes selective data loading by the user possible.

Another traditional file-based CAD problem is that write access is locked on the entire file, that contains a large portions of the design, thereby limiting concurrent design access. In GSCAD, loading the information to the PC does not write lock the data. Write locks are created only when the user attempts to edit the information. When information that has been loaded in the PC memory is referenced (as input for a design operation), the system automatically assures that it is up to date with the database. All additions and changes the user makes on the PC are immediately posted to the database to support concurrent design with other users.

GSCAD provides a shipyard control over the automatic propagation of changes by creating boundaries of responsibility. The shipyard may elect to automatically stop associative-driven changes based on access control. A notification is provided in the form of a "To Do List" to permit the responsible individual to review and accept changes in his area of responsibility. For example, an outfitter may locate a piece of equipment at a specified distance from a bulkhead. If the structural designer moves the bulkhead, the equipment whose position depends on the bulkhead gets an automatic associative update. However, since the structural designer does not have write access to the outfitting equipment, the system records the requirement to update position in a "To Do List". The outfitter must issue the permission to carry out the change.

Command Tips and Messages

As with all Microsoft Windows applications a ToolTip and Tool Message are presented when the mouse is held still over a command so the user does not have to remember the command's icon. Figure 6 is an example of a ToolTip and Tool Message.

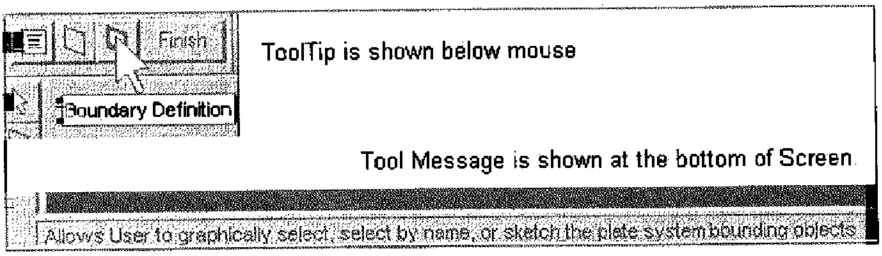


Figure 6 - ToolTip & Tool Message Example

The system also provides prompts at the bottom of the screen once a command has been selected. At each step in a task the system prompts the user with what it is expecting next. This frees the user from remembering the command sequence. An example of a system prompt is shown in Figure 7 below. The example is from the Planar Plate system command. Once the user has finished the plane definition step the system moves to the boundary definition step and displays the system prompt shown.

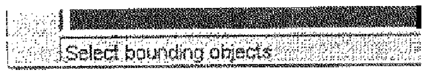


Figure 7 - Boundary Step Prompt

<u>Common tools across Task Environments</u>

Another guiding principle of the GSCAD user interface is to accomplish similar tasks in a similar way across all the Task environments. A common set of tools are available to the user and are presented as needed regardless of which Task environment the user is in. The follow paragraphs describe some of these common tools.

QuickPick is a common tool used by most of the GSCAD Task environments. The nature of ship design requires users to display thousands of objects in a single view. Graphically picking an individual object can be very difficult. As the user moves through the SmartSteps, each Task Environment tailors the select command based on the currently needed objects. This filters out many of the objects in the view leaving only the needed object types as pickable. Among the many pickable objects in a view QuickPick allows the user to quickly select the desired object. Within a GSCAD graphical view when there is more than one pickable object under the mouse the system first highlights the object on top. If the mouse is held still for a moment a question mark appears next to the mouse icon. If the user left mouse clicks a QuickPick window is displayed. As the user, with the mouse up, moves the mouse over the QuickPick window's numbered boxes the object associated with the numbered box highlights in the view. The user selects the numbered box associated with the desired object. Figure 8 illustrates the QuickPick process.

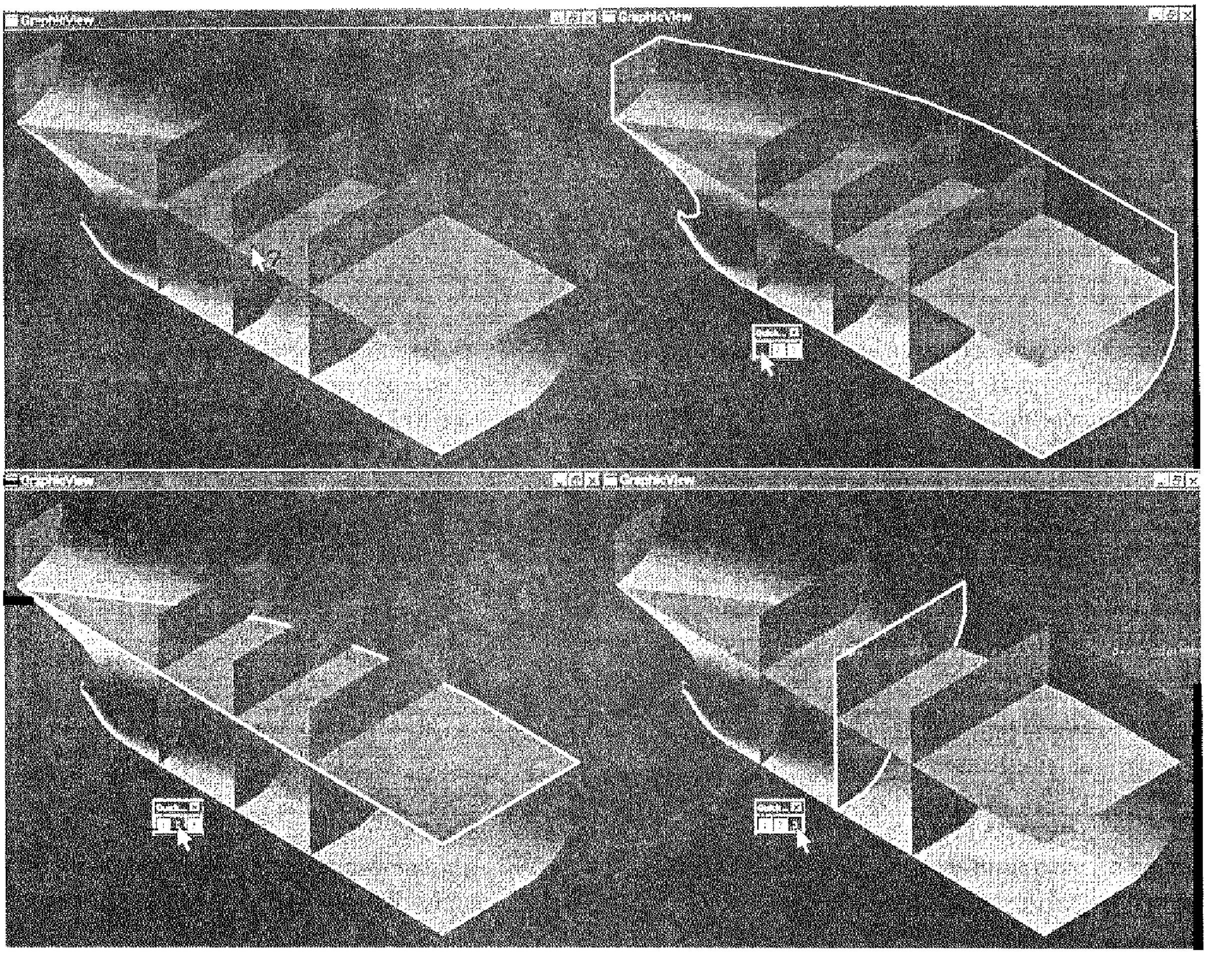


Figure 8 - QuickPick Example

SmartSketch is a common capability that allows the user to define geometric constraints for the object being created or modified. The Task environment determines what constraints are available to the user. The system anticipates what constraint the user may want based on the objects under or near the mouse and displays a small symbol next to the mouse. If the user left mouse clicks while the constraint symbol is displayed then the constraint is established. The user can control which constraints are active. Figure 9 shows the SmartSketch form that controls the active constraints for the 2D sketcher environment. It shows the constraint and the symbol that is displayed by the system when it determines that the constraint can be placed.

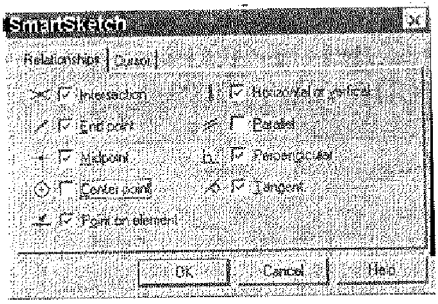


Figure 9 - SmartSketch Constraint Form

Figure 9 is an example of SmartSketch at work within the 2D Sketching environment. The image shows the 2D Sketching environment being used to define the geometry of an opening on a deck. Next to the mouse is the symbol for the end point constraint. The highlighted vertical and horizontal lines will be connected at their end points if the user left mouse clicks while the end point symbol is displayed. Once a constraint has been placed a symbol is displayed to indicate that the constraint has been placed. In figure 10 the connection constraint is represented by a small square over the connection. The horizontal / vertical constraint is represented by a small cross on the element that has been constrained. The user may delete a constraint by selecting the constraint symbol and pressing the delete icon or key.

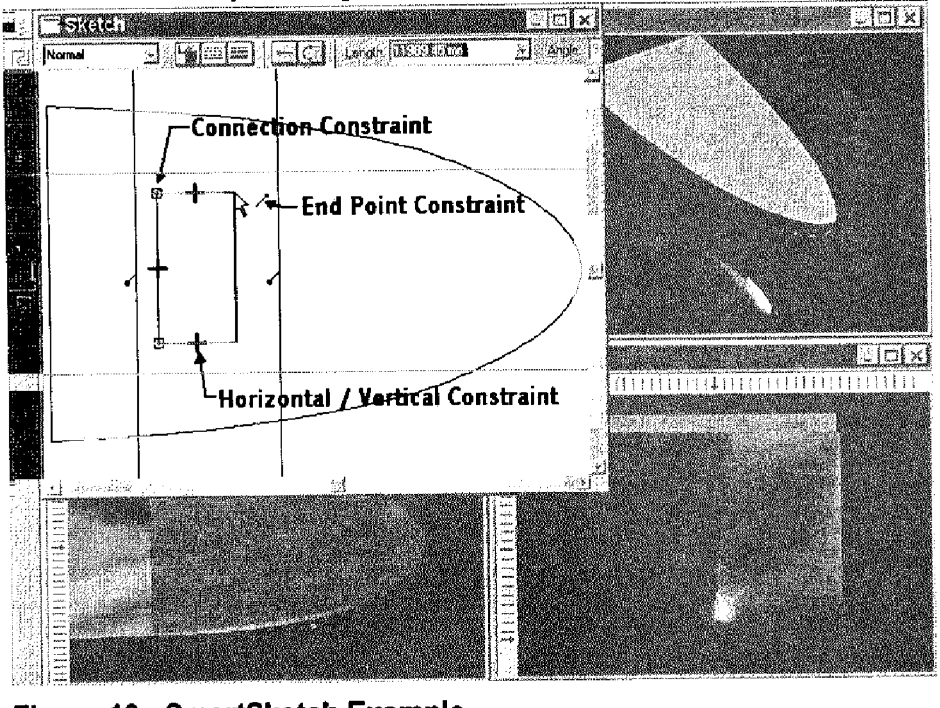


Figure 10 - SmartSketch Example

Pinpoint is another common tool that allows the user to obtain location coordinates within a graphic view. The user invokes Pinpoint from the main menu tools option while in any Task environment. The Pinpoint ribbon bar is then displayed. Pinpoint displays X, Y, & Z values of the mouse location relative to a user-defined target location. The default target location is the origin of the active reference plane system. The target can be set to any pickable object in the view. Figure 11 shows the Pinpoint ribbon bar and the X, Y, & Z coordinate display.

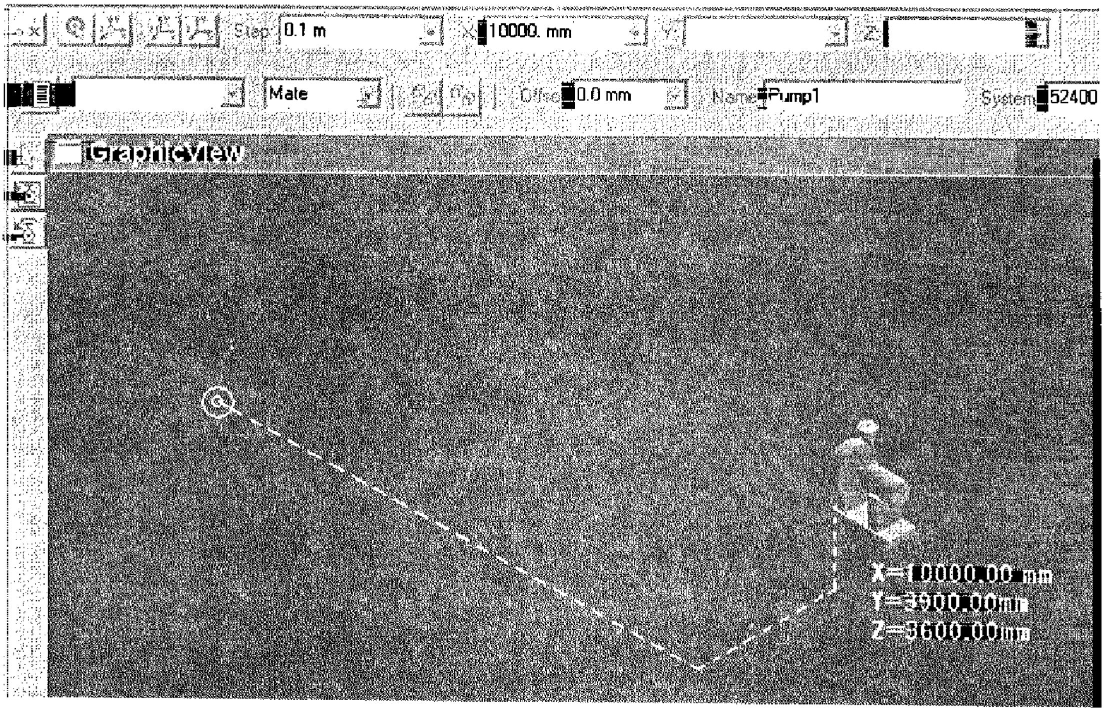


Figure 11 - Pinpoint Example

The Step option on the Pinpoint ribbon bar allows the user to establish a grid of points that the mouse will snap to. The user can also constrain the mouse location by entering a value in the X, Y or Z fields on the ribbon bar. In the figure above the user has entered a step value of 0.1 meters and an X value of 10 meters and therefore only the Y & Z values will be allowed to change and their values will round to the nearest 0.1 of a meter.

Point On Element is another common tool, similar to Pinpoint, that allows the user to obtain a location on an existing curve. The user selects an existing curve and then defines a target on the curve. Unlike Pinpoint's display of XYZ coordinates next to the mouse, the Point On Element displays the girth distance from the user defined target to the mouse location. The existing curve can be an edge, a landing curve, seam, or intersection curve.

Conclusion

GSCAD is a next generation ship product modeling tool that provides a quantum step forward in ease of use over today's ship product modeling tools. The ease of use of the system will reduce the costs associated with a product modeling tool learning curve. But more importantly the ease of use will make the rich collection of data contained it the ship product model more accessible to the decision makers throughout out the ship's life cycle. This improved access will enable significant process improvements and eliminate the need for many of the documents produced today to communicate the ship design.

GSCAD's low hardware requirements and Microsoft Windows environment will allow companies to combine office automation and a powerful product modeling tool on a single personal computer. It will dramatically increase the number of ports that can access the rich set of data contained in the product model.

ON THE DEVELOPMENT OF ACIM PRODUCT MODEL AND ITS VERIFICATION THROUGH PRACTICAL APPLICATIONS

Yutaka Nagase, Toshiyuki Amemiya, Fumiaki Tanigawa, Yuichi Sasaki, Koichi Ujigawa, Ship & Ocean Foundation, Tokyo, Japan

Abstract

Advanced CIM (ACIM) for shipbuilding project sponsored by the Nippon Foundation, with participating 7 major shipbuilders in Japan, was launched in 1997 with plan for 3 years.

The vision of ACIM project is to enable highly effective, efficient and flexible collaborative engineering environment for shipbuilding on the basis of the knowledge sharing. One of the most important R&D tasks in the project is to establish the practical product model for shipbuilding, which expresses the ontology of shipbuilding and plays the key role for the knowledge sharing in ACIM environment.

When developing the workable product model, maximum use was made of GPME (General Product Modeling Environment) which has been developed in the preceding CIM project, detail of which was reported at ICCAS'97. GPME is composed of two major components; one is common ontological information model for assembly industries implemented as so-called CFL (Common Frame Library) and the other is a product model development environment, i.e. a set of tools for extending the class libraries. In the GPME project, CFL was extended into EFL/S (Extended Frame Library for Shipbuilding) which includes ontology for shipbuilding. However, it was limited extension to verify the usefulness of GPME through the execution of small-scale sample applications.

In ACIM project, extension is being tried of EFL/S to much more practical level. The product model is realized by combined use of CFL and EFL/S and becomes a base for the engineers' collaboration. EFL/S is intensively extended in two directions. One is to enhance the product model so as to be rich in modeling flexibility; this leads to the easy design alteration and simulation. Another is to extend the expression capability so as to express most of the hull structures including details and almost all outfitting of actual ships. Although the development is still under way, the practical EFL/S will be completed by March 2000. The characteristics of ACIM product model will be discussed in detail.

The functionality of extended product model is verified through the execution of newly developed 2 applications, i.e. application for block breakdown and process planning. The paper also describes the features of these practical applications.

Introduction

The core of ACIM project is related to the product model in the same way as in the past R&D for CIM for shipbuilding which has been conducted by the project teams of Ship & Ocean Foundation for the past 10 years. In ACIM project, concepts of intelligent agent and process model are introduced to support collaboration by sharing knowledge, and this paper will show ACIM product model and practical application based on its product model, while overview and collaborative engineering of this project are referred to papers submitted by other authors. The concept of the product model had already been reported at ICCAS'94 and ICCAS'97. The concept of product model now being presented is a developed one in which ontology of the shipbuilding was expressed with the

Entity-Relationship model, and it is realized in the computer based on the object-oriented concept. The product model includes not only data but also behavior. By having data and behavior together, more advanced model at higher level can be expressed than those with the usual database.

To share knowledge among participants, they and computer systems should have same ontology, or in other words, product model based on ontology is necessary to share knowledge. Frame Library (FL) has been defined as a class library to build a product model, and "Extended Frame Library for Shipbuilding (EFL/S)" which is Frame Library that was extended for the shipbuilding in GPME project is now being developed as ACIM EFL/S of higher level, which it is hoped to be used actually in the shipbuilding industry.

Expansion of Frame Library

In GPME project, Common Frame Library (CFL) has been defined for assembly industries on a general basis, and it consists of 6 parts (Common, Structural design, Piping design, Fabrication & assembly, Production planning & control, Production resources). Among these, it appears that "Production planning & control" and "Production resources" are generally applicable to practical use in shipbuilding, since there are few characteristic aspects in these fields of this industry comparing with others. Therefore, these 2 parts of GPME CFL can be adopted as they are. Accordingly, in the present project, remaining 4 parts of CFL are being expanded to EFL/S to build the product model of the utility level. As there exist few shipbuilding-own features in Common part of EFL/S, this will be excluded from this paper, and expansion of remaining 3 parts of EFL/S is to be shown.

ACIM EFL/S for Structural Design

As have been developed by GPME project, there exist advanced data structure and capacity in EFL/S to realize the capability of variety which could not be implemented by legacy CAD system. GPME EFL/S is said to have a superior flexibility in the model construction and adjustment.

However, for the expression of the details or special structure, the expression capability of the GPME EFL/S is not sufficient from the viewpoint of practical use. This is due to the fact that GPME EFL/S had not necessarily been expanded for the practical use and the extent of its expansion had been limited, although it was developed for the shipbuilding industry.

GPME EFL/S for structural design was reported at ICCAS'97 in detail. Following the same procedures, this EFL/S is being expanded further for practical use in ACIM project. There are two targets in the expansion of ACIM EFL/S for structural design. One is "To improve the manipulation of the model" and the other is "To expand the coverage of the model" as shown in Fig. 1.

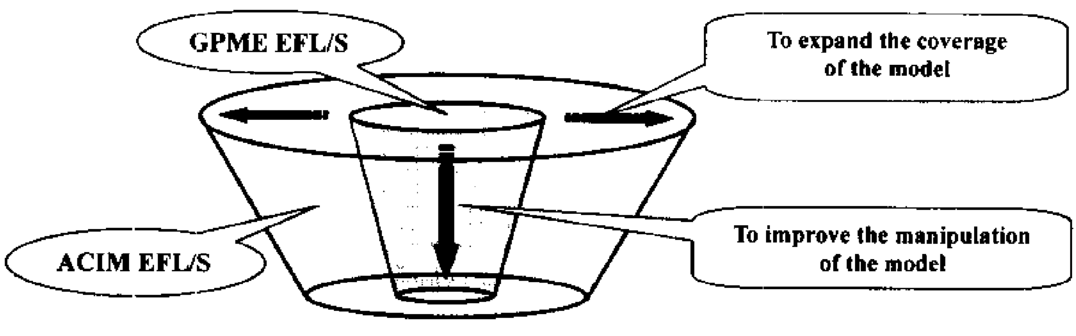


Figure 1. Targets to Expand EFL/S

"To improve the manipulation of the model" aims at more advanced product model by improving the manipulations such as defining or modifying the ship data in the product model more easily and efficiently. "To expand the coverage of the model" aims at the sufficient coverage to express the whole actual ship data in the product model.

1. To Improve the Manipulation of the Model

Many functions are necessary to support the work which ACIM EFL/S for structural design (called "ACIM Structural FL" hereinafter) intends to cover. Functional requirements for ACIM Structural FL, especially from the viewpoint of advanced manipulations, have been recognized as follows.

Flexible counter action in case of design change:

At the early stage of the design, changes of the product model occur frequently since the design proceeds in trial and error method. Usually, when the designer makes a change in design, consequent changes should be made in other part where contradiction occurs. In other words, when the related design is left as it is, contradiction will arise. Therefore, the design which has been affected by the design change has to be surely modified to keep consistency. If such consequent changes are done one by one manually, there always exists the possibility of human errors. Thus, ACIM Structural FL is required to automatically adjust such related members in accordance with the rules based on the common knowledge and practice of the shipbuilding.

The model which has the common knowledge of the shipbuilding:

There is various common knowledge in shipbuilding; for example "Longitudinal frame goes through the transverse bulkhead". Each kind of structural member, such as longitudinal frame, bulkhead, etc., has individual characteristic which is regarded as common knowledge in shipbuilding. Without expressing such common knowledge, for example when two structural members intersect each other, it can not be distinguished in the product model which one should penetrate the other one. Therefore, ACIM Structural FL is required to have such knowledge which is common in shipbuilding, and it can construct the product model of ships efficiently, meeting the common knowledge in shipbuilding.

To keep the consistency between structural design and assembly method:

The assembly method is expressed in the GT code and the network (assembly tree) of the intermediate products. Without expressing clear relationship between these objects and structural members, when any change occurs in assembly methods, inconsistencies between assembly method and structural members/parts, which are once determined according to the assembly methods before changed, can not be detected in the product model. To solve such problems, ACIM Structural FL is required to recognize the affection on structural design by the change in assembly method.

To meet these requirements, ACIM Structural FL has been expanded as follows.

- To introduce the relative definition.
- To expand the application of trim relation.
- To define affections among partition walls, structural members and structural parts.
- To keep the design intention.
- To keep the information which is the basis to define layout of structural members.
- To express the role of structural members.
- To keep the consistency between structural design and assembly methods.

2. To Expand the Coverage of the Model

In ACIM project, it is required to have the FL of practical level. To meet this requirement, ACIM Structural FL has been expanded on large scale in expressing capability of hull structures including details. As a result, almost all the hull structures which are found on merchant ships are expressed practically in ACIM Structural FL. The structures that are expanded in ACIM project are as follows:

Rounded gunwale:

By expressing the plate member that is trimmed by the free edge of other members, rounded gunwale can be expressed.

Pillar:

A pillar member is added newly in ACIM Structural FL, and it can express pillar specifically.

Stiffening member connecting to more than two members:

Expansion is made to express a stiffening member connecting to more than two members, such as a stiffening member connecting outside shell and upper deck.

Structural details:

ACIM Structural FL is expanded to express the existence, type and shape of structural details as shown below. Thus, no lack of information will be ensured in the course of data exchange with the existing shipbuilding CAD system.

- Structural details of stiffening member
- Structural details of face member
- Structural details of pillar member
- Opening, air hole, drain hole, etc.
- Scallop
- Slot, collar plate
- Doubler
- End bracket
- Tripping bracket
- Appendage: No information on its shape, but its location is expressed.
- Pad
- Rib
- Joint

ACIM EFL/S for Outfitting Design

1. Requirement

GPME CFL contains the FL for piping design (called "GPME Piping FL" hereinafter) which has the remarkable features such as:

- To express piping systems as design objects, and to be capable of constructing a model which has consistency between the piping system and piping layout.
- To express physical connections between piping elements, not as attributes of piping elements, but as independent objects. Thus to be more flexible and rich in expressing the physical connections.

However, GPME CFL and EFL/S do not have any FL for outfitting other than piping design.

In most Japanese shipyards, expectation to have an advanced system for process planning is getting higher. To fulfil this expectation, not only piping but also other outfitting elements have to be expressed in the same product model.

From these points of view, requirements to ACIM EFL/S for outfitting design (called "ACIM Outfitting FL" hereinafter) have been recognized as follows.

- To express all kinds of outfitting elements so that process planning is able to be carried out on the product model.
- To express not only piping system but also other outfitting systems as design objects, consequently to be capable of constructing a model which has consistency between the systems and layouts.
- To express not only physical connections but also other relationships as independent objects, thus assuring to be more flexible and rich in expressing various kinds of relationships.

2. Outline of ACIM Outfitting FL

To meet the requirements mentioned above, ACIM Outfitting FL has been developed as follows.

Class:

From the viewpoint of natures of information, ACIM Outfitting FL consists of 4 groups, i.e. design object, element, catalog and relationship. Furthermore, design object group consists of system specification, system and system space. Schematic object-model diagram of these groups is shown in Fig. 2.

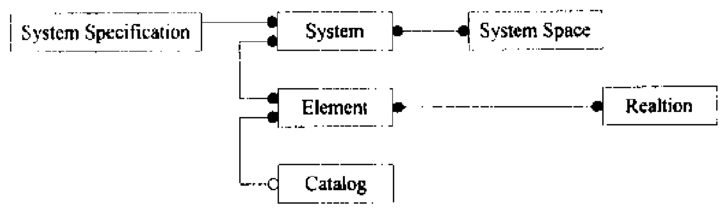


Figure 2. Schematic Object-Model Diagram of ACIM Outfitting FL

On the other hand, from the viewpoint of outfitting aspect, ACIM Outfitting FL consists of 6 fields, i.e. machinery, piping, electrical fitting, ventilation ducting, traffic fitting and supporting. Each field has corresponding groups of classes according to its characteristics as shown in Table 1.

Table 1. Summary of Classes in ACIM Outfitting FL

Group \ Outfitting field		Machinery	Piping	Electrical fitting	Ventilation ducting	Traffic fitting	Supporting
Design object	System specification		+	+	+		
	System		+	+	+	+	
	System space		+	+		+	
Element		+	+	+	+	+	+
Catalog		+	+	+	+	+	+
Relationship		+					

+: To have corresponding class(es)

The System Group consists of piping system class, electric wiring system class, ventilation ducting class and traffic system class.

The Element Group of each outfitting field has sub-classes that represent actual outfitting parts. For example:

- The Piping Element Group has pipe class, pipe equipment class and pipe fitting class.
- The Electrical Element Group has electric cable class, cable tray class and cabling equipment class.
- The Traffic Element Group has vertical ladder class, inclined ladder class, platform class, handrail class, etc.

The Catalog Group expresses the catalog information for outfitting parts such as shape or parametric pattern of shape, inlet/outlet points and directions, etc.

The Relationship Group consists of:

- physical connection between outfitting elements
- physical connection between outfitting element and hull structure
- penetration to hull structure by outfitting elements
- indirect relationship

The indirect relationship expresses situations of objects which are not physically connected but related each other in view of function, restriction, etc. Such situations are for example:

- Specific stiffener under the deck plate is arranged for reinforcement against the machinery located above. (Physical connections are not able to express this situation exactly since they only express the relationships of "machinery" - "foundation for machinery" - "deck plate" - "stiffener for reinforcement".)
- Specific two valves are installed in different piping system, but have to be located closely as they are to be operated simultaneously.
- Specific handrail should be located 4,000mm or more apart from this magnetic compass according to the rules by the regulatory bodies.

By having such indirect relationships in a model, the model is able to find which outfitting elements are affected and should be checked when modifying an outfitting element, even if they are not physically connected.

Method:

The classes in ACIM Outfitting FL are provided with methods necessary to maintain the consistency of the model structure. Such methods are for example:

- To create an element instance, simultaneously creating the association between the element and the corresponding catalog
- To delete a system instance, simultaneously deleting the elements in the system

Frame Library for Fabrication and Assembly

As for fabrication and assembly, there are various phases of engineering from the block breakdown at initial stage to the work order and instruction at actual fabrication stage. In ACIM project, from the viewpoint of their weight in the whole work and their influence on production efficiency, concentration has been made on developing FL related to the block breakdown and the process planning.

The block breakdown is named "Initial Production Design (IPD)" in ACIM project, since it is regarded to be included in the production design in wide sense.

1. Initial Production Design

ACIM EFL/S has been provided with the following two remarkable functions for the initial production design.

To express relative positions:

In general, the position of a block boundary is defined relatively from a certain object with a certain distance This is regarded as one of the intentions of the block breakdown designer. From this viewpoint, ACIM EFL/S has been required to express relatively defined position of block boundary. To meet this requirement, attributes to express the relative definition between block boundaries and any other objects have been provided for the block boundary class in ACIM EFL/S. For example, by referring to these relative definitions, it is able to modify the block boundaries maintaining the initial intention.

To recognize characteristics of structural member:

For example, stiffening members should not be divided longitudinally in their faces or webs even in case of misoperation in application systems. Also secondary structural members such as brackets should not be divided as far as possible. Instead, understanding such characteristics of structural members, block breakdown designer should judge whether to modify block boundaries or arrangement of structural members. From this viewpoint, ACIM EFL/S has been required to recognize whether the objects are dividable or not. To meet this requirement, objects in ACIM EFL/S are classified according to their characteristics if they are dividable or not. Then, it is possible to prohibit dividing a stiffening member anywhere other than its fitting line, by warning the designers if a block boundary and a bracket intersect each other or not, and so on.

2. Process Planning

Process planning is the engineering work to define order of assembling, workshop where the work to be done, and working procedures. From this viewpoint, following two key functions are implemented in ACIM EFL/S.

Coherent process planning for structural parts and outfitting parts:

To get an optimum assembly process, coherent planning with hull structure and outfitting is required. To meet this requirement, it is necessary that all the parts of hull structure and outfitting are to be expressed in the same format, as well as they are distinguished into classes accordingly.

In ACIM EFL/S, both of structural parts and outfitting parts are abstracted as a super class named "parts class" which is associated with "intermediate product class". Then, in process planning, it is able to manipulate instances of parts by any intermediate product without distinguishing the kinds of parts. On the other hand, it is also able to distinguish the kinds of parts easily since they are categorized in sub-classes (e.g. machine class, pipe class, etc.). Fig. 3 shows schematic object-model diagram of the parts class.

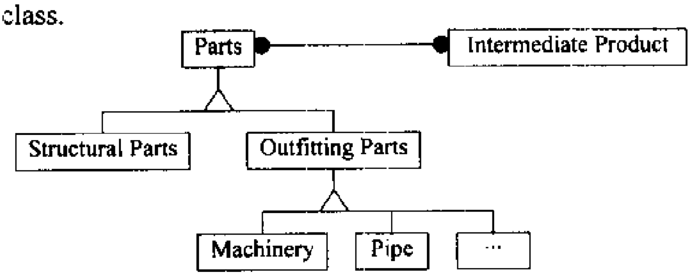


Figure 3. Object-Model Diagram of "part class"

To keep the consistency between structural design and process planning:

There are important relationships between shape of hull structure and assembly method.

Consistency of this relationship has to be kept in computer system. Fig. 4 shows a typical example in case of the type of slot and how to install the transverse frame, i.e. to set it down from above or to fit it horizontally in line with longitudinal frames by inserting into slots.

Each component in assembly process is expressed in a class named "intermediate product class", and its assembly method is expressed in an attribute of that class. Then, for example, type of slot can be defined so as to match the assembly method of the intermediate product to which the slot belongs. The detailed procedure is as follows.

- Step 1: Search slot instances which are being designed.
- Step 2: Search the intermediate product which is related to those slot instances, and get its assembly method.
- Step 3: Choose the slot type in consideration of assembly method.
- Step 4: Define the slot instances according to the appropriate slot catalog.

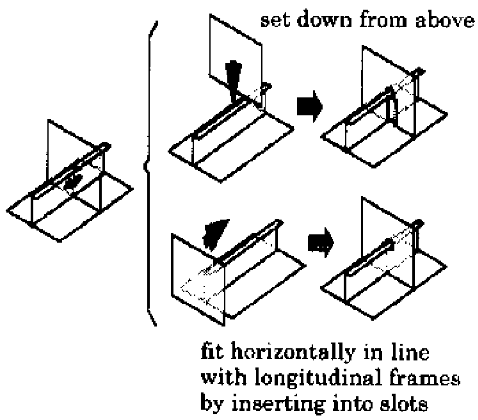


Figure 4. Slot Type and Assembly Method

Result of Expansion of Frame Library

The expansion of EFL/S is still under way, and the expanded FL is now being verified in line with the progress of the implementation. The whole object-model diagram of ACIM EFL/S is shown in Fig. 5. One sheet in the figure is for EFL/S of Common, Structural design and Fabrication & assembly, and the other is for Outfitting design. The classes that express the patterns of structural details are not shown here.

Examples of constructing and manipulating the product model of double hull tanker by using EFL/S which are already implemented are shown in the accompanying figures. By this verification, the expansion of EFL/S is confirmed to be as specified. Based on these results, the specification is now being reinvestigated for the purpose of further improvement in the function.

Fig. 6 shows that the trimmed walls follows the movement of the trimming wall, and it is obvious that the shape of trimmed ones (longitudinal bulkheads and boats'n store deck) are transformed accordingly after the movement of trimming one (transverse bulkhead). Fig. 7 and 8 are examples of structural details. Fig. 7 shows the openings and doublers, and Fig. 8 shows twisted stiffening members.

Fig. 9 shows an example of arranging some outfitting elements including those of the 6 outfitting fields. Fig. 10 shows an example of defining associations between a system instance and element instances in the traffic fitting field.

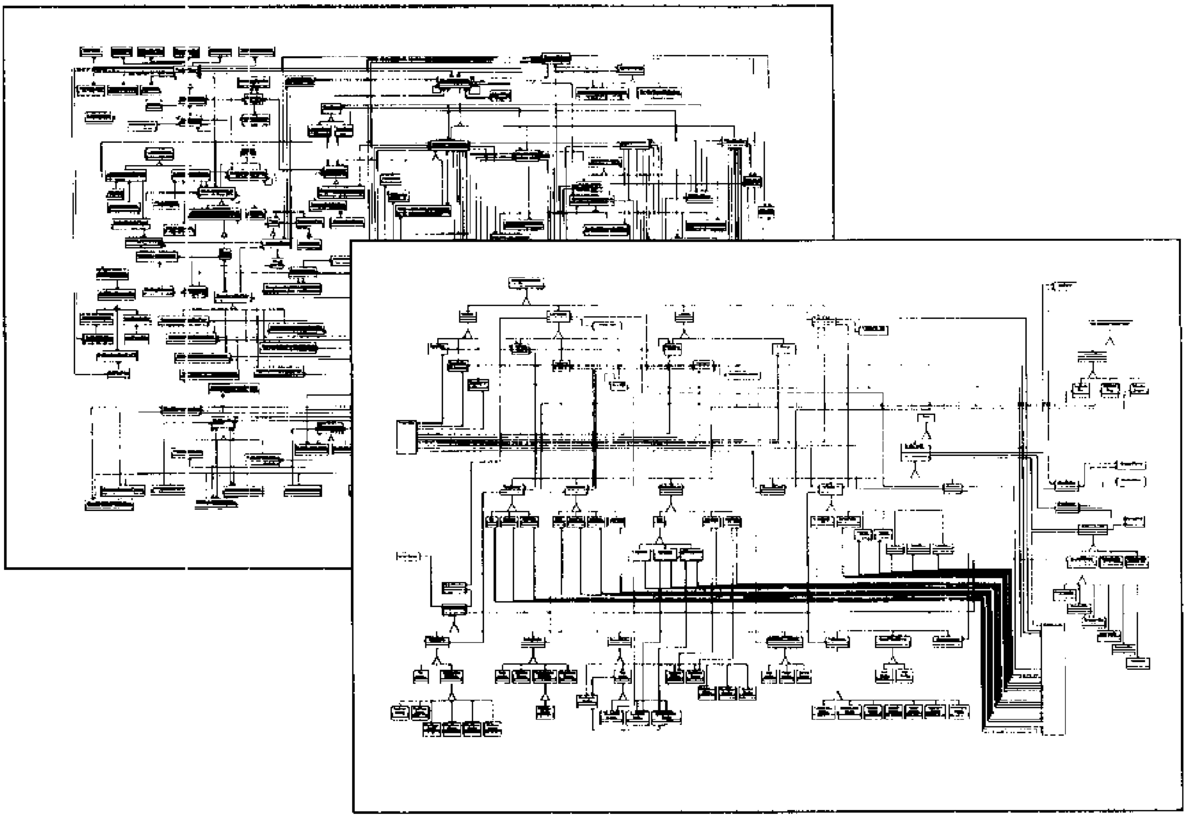
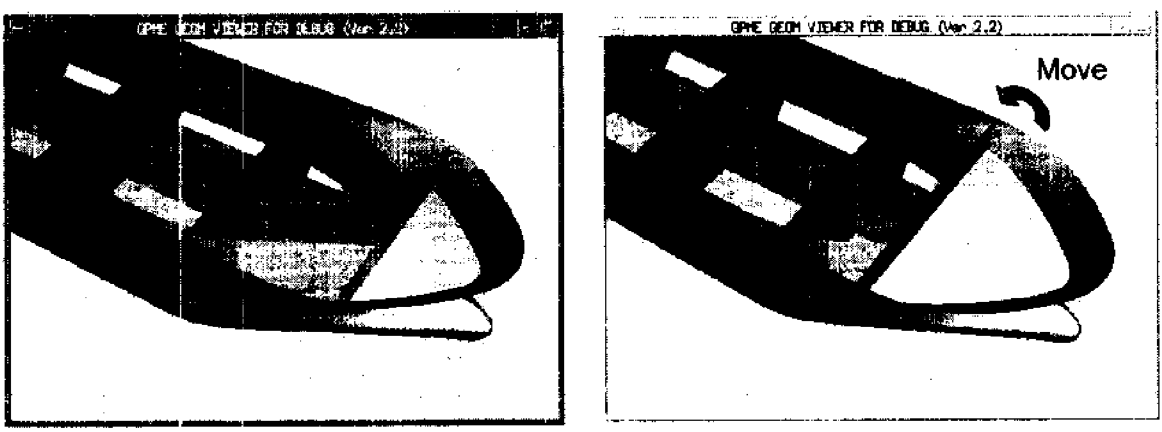


Figure 5. Overview of Object-Model Diagrams of ACIM EFL/S



Before the movement

After the movement

Figure 6. Automatic Transformation of Trimmed Walls

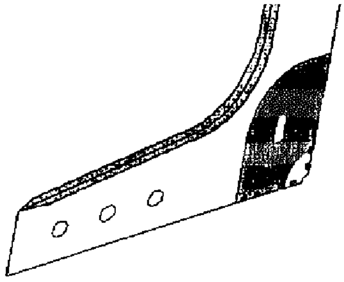


Figure 7. Openings and Doublers

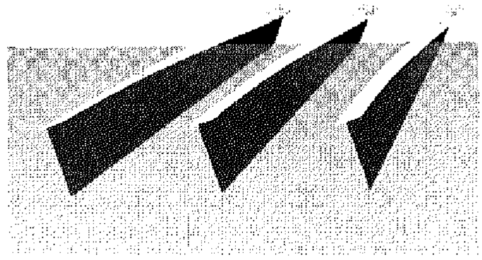


Figure 8. Twisted Stiffening Members

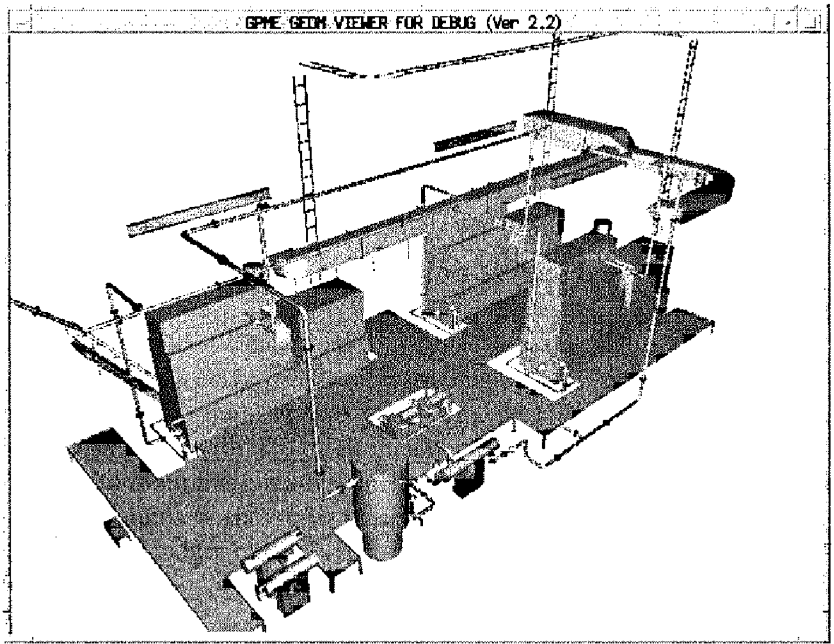


Figure 9. Arrangement of Outfitting Elements

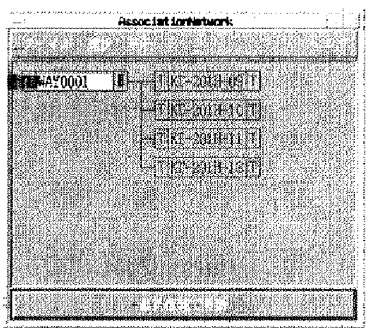


Figure 10. Association between System Instance and Element Instances

Verification of Expanded Frame Library by Practical Applications

In ACIM project, functions of the FL which are required for the initial production design and process planning have been extracted as described above. To verify especially these functions of the FL as well as others, 2 practical applications have been developed. One is an application to support initial production design so-called block breakdown (ACIM IPD; Initial Production Design), and the other is an application to support process planning (ACIM CAPP; Computer Aided Process Planning).

ACIM IPD

1. Outline of ACIM IPD

ACIM IPD is an application to define block breakdown. Operating procedure in ACIM IPD is shown in Fig. 11.

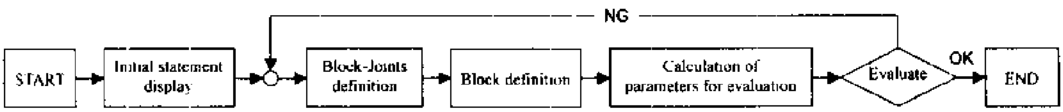


Figure 11. Operating Procedure in ACIM IPD

"Block-Joints definition" phase includes the functions such as:

- To define a transverse block boundary based on a certain transverse object.
- To define a longitudinal block broundary based on a certain longitudinal object.
- To make a block boundary following the base object.
- To define joints in the location where block boundary intersects structural member.
- To define a joint in stiffening member in the direction that crosses fitting line of stiffening member.
- Not to define joint in secondary member such as brackets.

Fig. 12 shows a screen image of ACIM IPD.

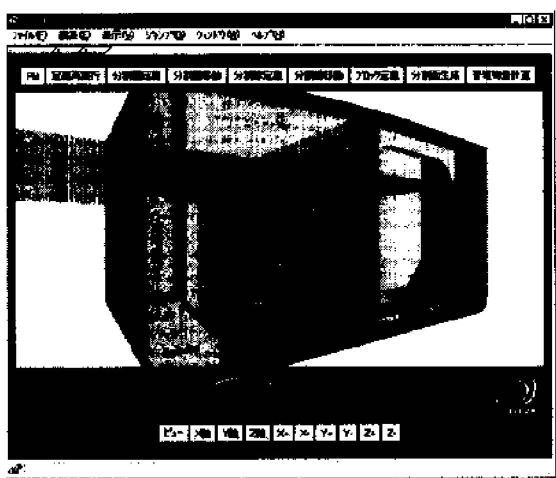


Figure 12. Screen Image of ACIM IPD

2. System Architecture of ACIM IPD

ACIM IPD is composed of APP, GF and DFF according to the ACIM reference architecture, which is referred to the papers by other authors. Fig. 13 shows the system architecture.

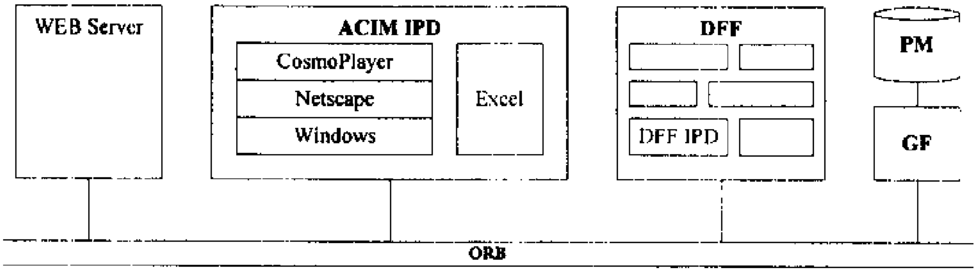


Figure 13. System Architecture of ACIM IPD

Main body of ACIM IPD corresponds to APP in ACIM reference architecture, which calls necessary services in GF and DFF to carry out block breakdown.

GF services called from ACIM IPD are such as:

- Open a product model
- Close a product model
- Start a transaction
- End a transaction

All the functions necessary for ACIM IPD are implemented in DFF, and those functions are categorized into DFF IPD (DFF for Initial Production Design). DFF IPD includes functions such as:

- To create wrapper objects of structural member instance, block boundary instance, joint instance and block instance. These wrapper objects have the methods of the corresponding instances as well as their attributes.
- To access wrapper object of structural member instance.
- To access wrapper object of block boundary instance.
- To access wrapper object of joint instance.
- To access wrapper object of block instance.

3. Verification of Expanded Frame Library by ACIM IPD

Following functions of ACIM EFL/S have been verified by ACIM IPD with satisfactory results.

- Functions of ACIM Structural FL in general
- Function to express relative position
- Function to recognize characteristics of structural member

ACIM CAPP

1. Outline of ACIM CAPP

In ACIM EFL/S, the assembly method is expressed in intermediate product class. Therefore, ACIM CAPP is regarded as an application to define the instances of intermediate product class.

2. Operating Procedure

Fig. 14 shows the flow chart of ACIM CAPP. The operation procedure is as follows.

- To read the characteristic of the intended shipyard such as assembly method practice and facility. Automatic process planning and the verification of the consistency between facility and assembly method are made based on these data.

- To read the data of hull structure design and outfitting design, and automatically define the network tree of intermediate products using rule-base which keeps various knowledge of process planning. Where it is difficult to define the network tree automatically, it shall be manually defined or modified from the automatically defined one.

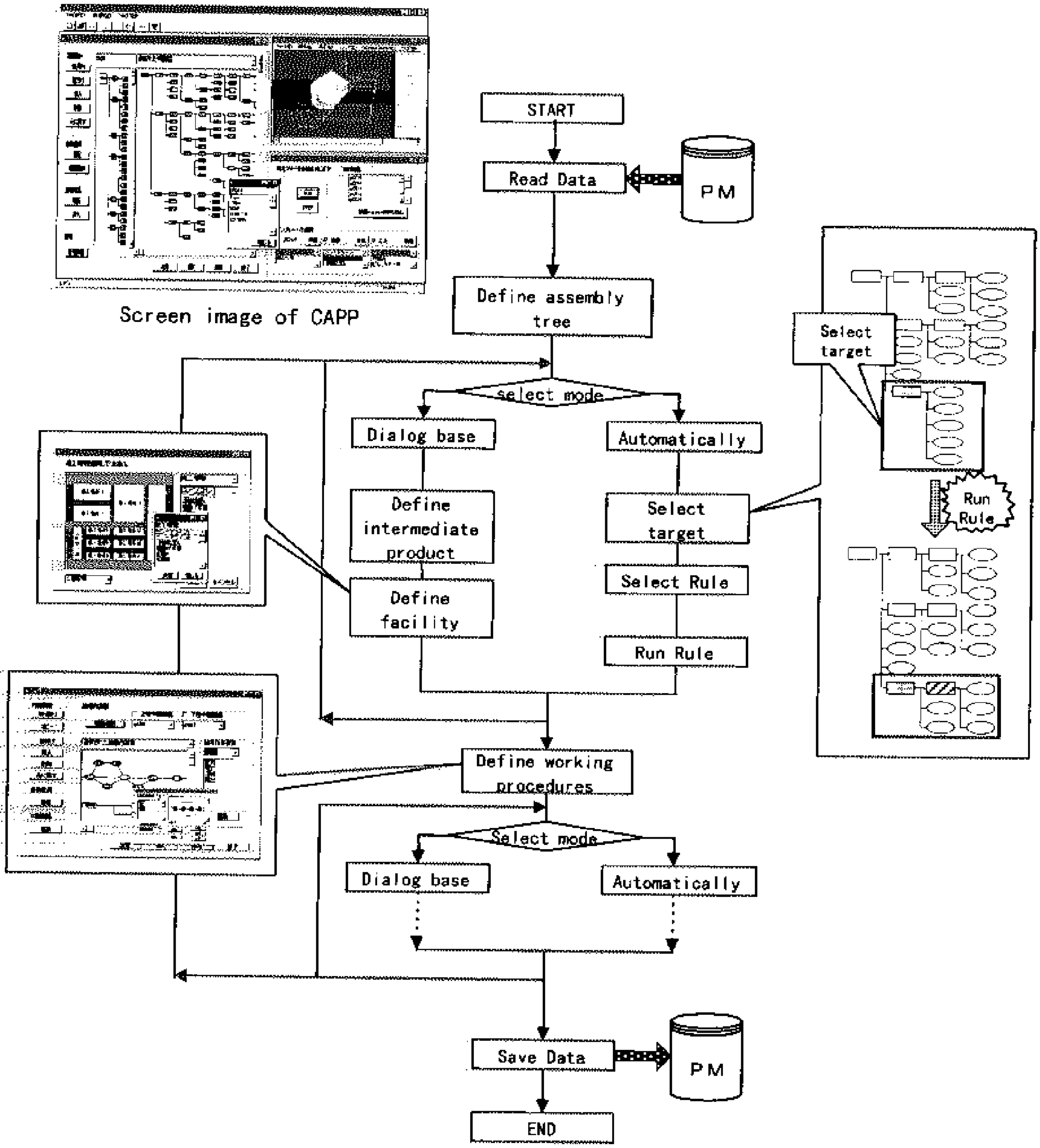


Figure 14. Flow Chart of ACIM CAPP

3. Expression of the Knowledge

To express the knowledge of process planning, "Frame knowledge" and "Rule knowledge" are defined in ACIM CAPP, based on the methodology of ordinary knowledge-base systems. "Frame knowledge" expresses the information of the subject product itself, such as number of the parts, position and direction of the parts, etc. "Rule knowledge" expresses the knowledge which helps to determine the assembling order of parts, such as "A support for pipe should be fitted to hull structure before the corresponding pipe".

Process planning knowledge of structural parts:

Two types of logic are defined; one is "Facility oriented design" and the other is "Product oriented design". The former defines the process by facility and fabrication shop, and the latter does in consideration of product's own characteristic.

Process planning knowledge of outfitting parts:

For the automatic design of outfitting parts, following 3 rules are defined.

- Rule1 : Define the procedure of the outfitting parts in consideration of structural parts.
- Rule2 : Define the procedure by the kinds of the outfitting parts.
- Rule3 : Define the detail procedure in each kind of outfitting parts.

As examples of Rule 3, Fig. 15 shows an expression of the rules and Fig. 16 shows a result of applying the rules.

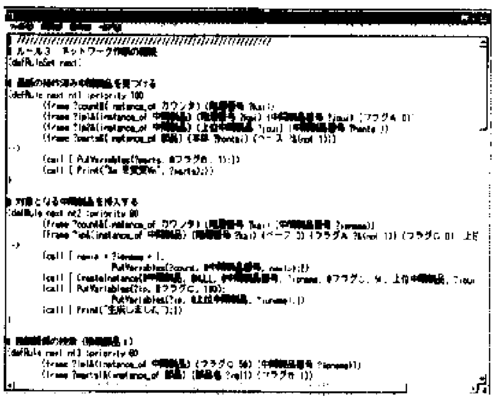


Figure 15. Process Planning Rules for Outfitting

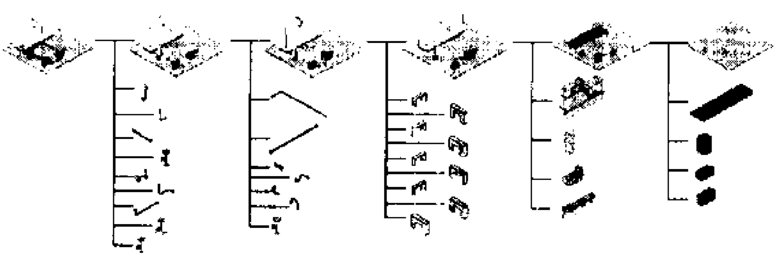


Figure 16. Example Result of Applying the Process Planning Rules for Outfitting

4. Verification of Expanded Frame Library by ACIM CAPP

Following functions of ACIM EFL/S have been verified by ACIM CAPP with satisfactory results.

- Functions of ACIM Structural FL and Outfitting FL in general
- Process planning for structural parts and outfitting parts concurrently
- Keeping consistency between the structural design and the process planning

Conclusions

1. Expansion of FL

ACIM project has been expanding FL from two points of view. One is to add intelligent functions or improve the existing functions, and the other is to improve the capability of expressing actual ship data. From these two viewpoints, the task of ACIM project has been decided with functional requirements. It is in progress just on schedule.

2. Development of 2 application systems

The practical ACIM/IPD and ACIM/CAPP have been developed to meet the latest user requirement. As a result, following two objects have been carried out. One is to extract requirements to a process planning application, and the other is to verify the expanded FL.

3. Verification of ACIM reference architecture

ACIM/IPD and ACIM/CAPP have been developed based on ACIM reference architecture, and their practical use with the expanded FL and the actual data has been proved. Thus, the validity of ACIM reference architecture has been confirmed.

Acknowledgements

This project has been carried out under the sponsorship of the Nippon Foundation, and the authors would like to express their cordial thanks to them. They would also like to express their gratitude for the guidance and encouragement received from Prof. Takeo Koyama, University of Tokyo and Prof. Yan Jin, University of Southern California.

References

1. Koga, K., CIM System and Product Model for Shipbuilding, ICCAS'94
2. Nagase, Y. et al., GPME Modeling Methodology and Applications, ICCAS'97
3. Ship & Ocean Foundation, 1997 Report on Research and Development of Advanced CIM for Shipbuilding based on Knowledge Sharing.
4. Ship & Ocean Foundation, 1998 Report on Research and Development of Advanced CIM for Shipbuilding based on Knowledge Sharing.

PROTOTYPE STEP DATA EXCHANGES IN SHIP INITIAL DESIGN AND THE PROVISION OF AN APPLICATIONS PROGRAMMER INTERFACE TO "TRIBON".

Don Catley, Kockums Computer Systems (UK) Ltd.

Abstract

This paper discusses the background requirements for a modern shipbuilding system, and highlights some of the advancements which are taking place concerning data exchanges. The paper is based both on the author's experience as Chairman of the Design Methods Committee of the International Ship and Offshore Structures Congress, who have collated a wealth of relevant information in successive state-of-the-art Reports, and in his involvement with three research projects, viz: Calypso and SEASPRITE in Europe and MariSTEP in the United States. These projects are each concerned with data exchanges using STEP protocols.

For any modern-thinking shipyard, a computer-based solution is now essential. Implementation of computer integrated manufacturing systems provides unique challenges to all shipbuilders. An applicable solution must be dedicated to the special demands of the complex process and its effective application requires proper formal training of all appropriate staff and a proper integration of the management, computer and production functions. The implied specialised knowledge of the industry and the requirements of its production practices means that it is beneficial to use special purpose software.

Efficiency and reliability of data storage and data communication to cover the lifecycle of a ship is now of paramount importance to a modern shipyard and their partners. Typical usage scenarios require co-operation between designers, owners, shipyards, model basins, approval authorities, sub-contractors and machinery and component suppliers. Increasing standardisation of the data and its presentation are focusing on the need for a Product Information Model which is flexible to the special purpose needs of the shipbuilding industry. STEP compliance of data structures will be of assistance, especially for the long term storage of ship data.

As part of their involvement in each of the above three projects, KCS have taken part in several prototype STEP data exchanges and are continually assessing the merits of alternative, evolving related technologies. This paper focuses on work concerned with the AP216 ship moulded forms schema and also AP215: compartmentation. Data files conforming to either SEASPRITE or MariSTEP schemas can now be imported to and exported from the TRIBON Initial Design Surface module. Furthermore, an Applications Programmer Interface (API) has been developed during the Calypso project to enable CFD vendors to interrogate the geometry of TRIBON Initial Design Surface or, via STEP exchange, more general surface models.

Colleagues are currently leading KCS involvement with related developments concerning AP218: structures and AP217: piping, see for example Reference 1.

Background to the Requirements of the Modern Design Process

The demands on the competitive design, production and maintenance of a ship are reflected nowadays in the following efforts:

- Improvements/refinements to the design and production processes,
- Seeking the optimum design/production process based on effective and extended use of computerised methods,
- Increased concern for and checking of life cycle aspects.

The initial design phase for a new ship is one of the most crucial phases in the lifecycle for the shipyard. Extra time spent in hull design delays the ordering of the steel; this in turn has a direct impact on lead-time. Thus there is pressure to shorten the design phase. Transfers of data between shipyard and model basin are a time-consuming part of this process and often still paper-based. There is a strong need for improved communications which avoid the redefinition of data and for analysis tools which reduce the design work and are convenient to use.

Innovative ship designs and more rigorous international regulations for ship safety and protection of the environment have increased the complexity of the design and production process. For example, the introduction of a double hull increases the cost of production by some 15-20% and then the maintenance costs are much higher. Shipbuilding is a 'one of a kind' industry with short lead times, and also a tendency for late design changes which must be accommodated without recourse to extensive reworking of the design. Unlike the situation in mechanical engineering, no prototype is built to test the design; a partial solution is for designers to become more involved in the design and optimisation requirements.

Advancements in the ship design process are such that almost all design activities are now supported by the use of computers. Computing power has never been so cheap. Hardware, which would previously have been well beyond the means of small- to medium-sized shipyards, is now within the budget scope of a one man consultancy. Portability, standardisation and integration of good marine software has been significantly improved recently but has only partially achieved the status of international standardisation.

From design, the computer revolution has moved to automated manufacture, planning and material control The correct level of investment in and educated use of computer-based systems is imperative for a yard or design agency to secure or maintain an all-important market edge against increasing competition. Shipbuilders are improving their facilities and operational procedures in a re-engineering of their processes and the organisation of the work carried out in their yards. Increasingly, the information required by the shipbuilding processes is being handled electronically. The challenge is to broaden the extended automatic transfer of data, and its integration and flexibility of access to those involved, so as to secure benefits at every stage.

The shipbuilding process is typically top-down and single discipline based, i.e. the structure is designed for the total ship and then divided into assemblies and piece parts. There is then a bottom-up design process to combine the piece parts into interim products of sub-assemblies, assemblies, units and blocks. The simultaneous requirements to also model the distributed systems (cabling, piping, ventilation and accommodation) imply a high demand for the application of well organised, concurrent engineering solutions and management systems. In the modern process it is the ability to generate, organise and manage the vast information flow which is important. The challenge is to utilise a continuum of information in the data flow through sales, design, production and maintenance such that work can be accomplished more effectively and designs can be more thoroughly optimized for production. Much of the current effort focuses on the key requirement to have an efficient Product Information Model for the project. Progress has been monitored in a series of ISSC reports, cited as References 2-5.

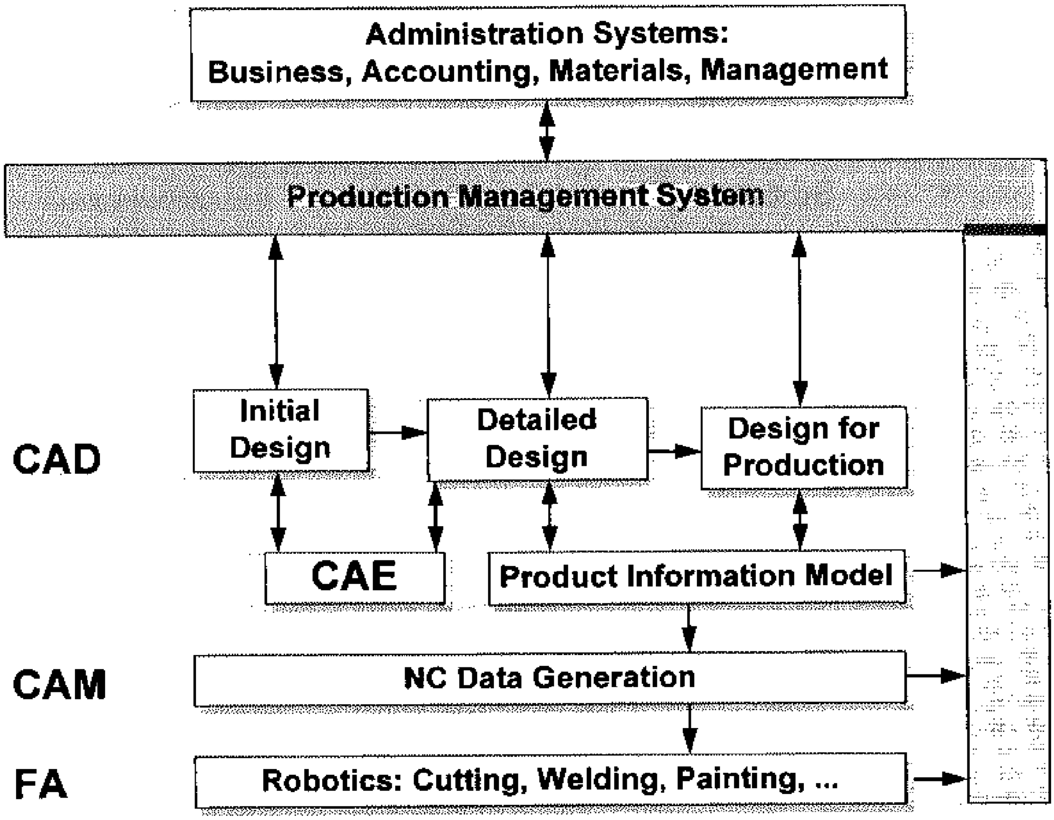
Status of Shipbuilding Solutions

For any modern-thinking shipyard, a computer-based solution is now essential. Implementation of computer integrated manufacturing systems provides unique challenges to all shipbuilders. These challenges and potential solutions (within budget) are not only different for each shipyard but also differ significantly between large and small shipyards. Software packages which are to be applied should be dedicated to the special demands of the complex processes involved and should enable the build strategy to be decided at an early stage. The implied specialised knowledge of the industry and the requirements of its production practices mean that it is beneficial to use special purpose software, as compared with general purpose CAD/CAM engineering software.

Before acquiring a computer-based system it is vital for a shipyard to consider carefully the benefits involved. The aim should not necessarily be for total automation in one step but for that degree of automation which results in the most efficient computer/human symbiosis possible. This will vary per yard and will depend on the constraints of the hardware, software and personnel. Thus it is strongly recommended that a yard should first consult closely with an experienced vendor who is fully conversant with the particular demands which the shipbuilding process will place on the system to be supplied.

Effective use of a computer-based system requires proper formal training of all appropriate staff and shop floor workers and a proper integration of the management, computer and production functions. Set up costs are reducing but extend beyond the initial cost of the software; however, the medium to long term benefits can be immense. Software vendors can assist the transition process by providing suitable training courses and materials which are specially adapted to the needs of the industry. Major shipyards world-wide are already benefiting from the advances which have been made to available ship design and production software over the past decade or so. Case studies which emphasise the benefits are available from KCS; see also Reference 6.

Figure 1 gives a general conceptual overview of a total shipbuilding system. The key function which is most likely not to be in place in a yard is the production management system. The **initial design** of a ship is heavily dependent on human designers having a rich experience. This process generally proceeds through three stages: concept design (parametric studies), preliminary design (precise vessel definition to meet the requirements and constraints) and the contract design (plans and specifications). Design for production is an important aspect which should be considered no later than the preliminary design. The know-how of a design team and a database of previous designs are important elements. The computer system can be used to encapsulate the human expertise and create automated systems. The environment is changing with the introduction of new technology but the concept of the design process is not. Expert systems are currently only in limited use. Series building and design standardisation offer the opportunity for important gains in time and cost.



CAD = computer-aided design
CAE = computer-aided engineering
CAM = computer-aided manufacturing
FA = factory automation
NC = numerical control

Figure 1: General Overview of a Total Shipbuilding System

During **detailed design**, (i.e. post contract design) the structure and outfitting systems are planned in detail in order to satisfy strength requirements and then achieve a cost and time effective building cycle. The detailed design stage is influenced by the build strategy (i.e. the block breakdown) and considerations of producibility and maintainability. The production activities use the information created in the detailed design stage to build the ship. The production can be divided into three steps, viz: parts manufacture, assembly, tests and trials.

The designer is continually checking the proposals against the requirements and constraints but has no way of assessing how closely the optimum design is being approached. Without tools it is not possible to explore enough cases to be certain that all the changes have been made in the best possible way. After probably many iterations, the designer ensures that the ship will meet statutory appropriate operational, strength and safety legislation and rules.

The detailed design of a ship includes many different engineering disciplines. Until recently a ship was created by various disciplines working separately; only after completion of individual efforts were the results integrated into one cohesive design package. This approach was inefficient and uneconomical because it generated an overlapping and duplication of effort as well as conflicts and interferences throughout the design. Implementation of the concept of concurrent engineering and the use of an appropriate CAD/CAE/CAM system offers the potential to resolve many of these inefficiencies. Moreover, the introduction of open, distributed systems will be needed to support the engineering tasks. Today most major design calculations are carried out by computer.

To meet the demand of reducing the building costs by improving productivity and reducing the time for design, the designer needs efficient exploratory tools. The use of computer applications specifically developed or customised for shipbuilding is recognised as the most effective approach. The requirement is suitability for initial design, detailed design and production. The available computer applications in most common use for initial design are mainly for preliminary design, drafting and engineering of the structure and the outfitting.

The high complexity of the final product implies an intensive design and planning process where many tasks have to be performed in parallel. The need for the highest quality at lowest prices and shorter times from order to delivery means that the number of overlapping activities must increase; **management of the information flow** necessarily becomes more complex. This situation can be described by the philosophy of concurrent engineering, also referred to as horizontal integration.

The processes involved generate a huge amount of data, typically up to one million related objects have to be connected in multi-level assemblies, and require an efficient management and control of the information flow There is a general trend towards closer integration of technical production data and information concerned with planning, estimating and materials systems (procurement, stock control, materials traceability, materials allocation and cost control). In future, this will not be separated information but will be automatically produced for central "viewing" by the various shipyard functions.

Due to severe competition throughout the world, during the past decade especially, and the changes in the shipbuilding business brought about by market forces and advancing technologies, much effort has been made by shipyards to reduce shipbuilding costs by improving productivity and by reducing construction time. Specific examples relate to reinvestment in different areas such as: modern equipment (robots), new facilities and use of computer technology at each stage of design through to the production. Particularly in Europe, the less competitive shipyards have been hardest hit by the continuing severe climate and increased competition from other nations expanding their activities and/or modernising.

A distinction must be made between large and small shipyards concerning the use of computer applications. The large shipyards have clearly understood the benefits of using a computer but this is not the case in most small shipyards. In large yards the use of computers in the different stages of design is a common objective and some yards have developed their own applications for at least part of the total system. A typical current requirement of a large shipyard is to update these applications by using new technologies and associated developments. Of concern is the avoidance of re-input or re-definition of data since this is tedious and inefficient, thus expensive, and can lead to errors and inconsistencies.

In parallel to the technical activities, important organisational and financial activities must be performed, viz: planning, materials control and financial considerations, in a general information flow of increasing volume. The non-technical activities will generally become better integrated in the next phase of developments. Emphasis is now on product definition with automatic document generation rather than the former process of drawings creation. Previous document- and drawing-based activities are increasingly being replaced by integrated computer based systems. Working practices must change in order for the maximum potential benefit to be realised. This involves a re-education and thorough training by specialists of all personnel involved in the activities of the shipyard.

Maintenance and repair considerations of the vessel impose design constraints such as demands on the structural arrangements and the availability of a suitable model with which to conveniently monitor the effect of deterioration.

The main objective of the use of a computer system at the design stage is to have a model of the complete ship, i.e. the **Product Information Model** (7, 8) from which any data can then be extracted at any stage in whatever convenient format is required. The information related to the Product Information Model is stored in the product data base which is the backbone of a CAD/CAE/CAM system (see Fig. 1), as well as the common and only source of information for all disciplines participating in ship design and production. Because of the magnitude of the complex characteristics associated with naval ship design, the application of a CAD system is then even more necessary. The benefits are discussed in detail in Ref. (6). The comprehensive description of the Product Information Model requires a vast amount of data to be provided by **concurrent engineering activities** (9,10). These data are often introduced by the users through a series of repetitive input operations.

To meet the requirements of design for production, the computer system should contain all the data and associated information necessary to support the complete life cycle of the ship, i.e. all aspects of the marketing, design, production, operation and maintenance. This is precisely the Product Information Model. The CAD/CAM systems produce information for individual parts manufacturing but often much less information for assembly, the most costly activity in the yard. The combination of CAD/CAM systems, functions for work preparation and MRP-II (manufacturing resource planning) forms a shipbuilding CIM (integrated) system.

Work preparation starts with a definition of the build strategy which is defined in parallel with the initial design. The detailed assembly definition is added later. Using this structured information the assembly production information and part lists are extracted from the Product Information Model. The information required by the workshops can also be stored in the model. A workshop needs to be able to request the information when it is needed; the trend is for the information to be transferred electronically, i.e. no paper delivery to be made to the various shops. In cases where a total integrated system is not implemented, standardised data transfer is increasingly being demanded.

Use of computers in **ship production** has already resulted in major cost and man-hours savings in scheduling, material tracking and computer-aided drafting (5). Reliable data are required to ensure a correct topology of components, correct structural detailing and assembly information for NC control of robots and other machinery. Accurate data for efficient piece-part cutting and fitting are demanded for efficient production. However, the dimensional accuracy of the steel profiles from the mills is still a potential problem for automatic manufacturing.

Relevant existing or emerging technologies for computer integrated manufacturing are:
- Artificial Intelligence/Knowledge-based Systems,
- Just In Time Practices,

- Vendor Relationships/Electronic Data Interchange (EDI),
- Concurrent Engineering (CE),
- CAD/CAM Systems,
- Rapid Prototyping Systems,
- Flexible Manufacturing Systems,
- Virtual Reality.

Summary of Benefits Expected from a Modern Shipbuilding System

Typical benefits to the users of an effective shipbuilding system can be summarised as:
- A quicker response to requests for quotations and tenders; shorter lead times,
- Increased accuracy, detail and quality of information and presentable documents for administration and production systems,
- Availability of a database of reference information, for example a hydrodynamics database of model test results,
- Availability of a Product Information Model for concurrent engineering activities, viz: outfitting, structural analysis, hydrodynamics, noise, vibration,
- Better designs through rigorous evaluation,
- Flexibility for design modifications,
- A more controlled environment for the purposes of standardisation,
- Improved cost control,
- Elimination of tedious manual and repetitive calculations
- Less rework,
- Greater material throughput,
- Less skilled labour needs to be employed although there is still a requirement to monitor the work, especially for design and lofting,
- Storage of lifecycle data for the ship.

Recent TRIBON Developments

Improved accuracy in design and production is well illustrated by application of the **TRIBON DOTORI** function, see Figure 2. This term is from the Japanese meaning quality and close fitting of parts (similar terms are niku biki and koshi). The function improves the accuracy of parts manufacturing and construction and reduces hours in the assembly and erection processes. It enables more productive use of automatic welding and welding robots. The latest NC cutting machines for both plate and stiffener piece-parts have the capability of controlling and changing the angle of the cutting heads. Within TRIBON the angle of the cutting heads can now be varied according to rules established by the customer. Benefits are: improved structural integrity and acceptability, less material waste and a quality finished product.

TRIBON has a unique approach to the management and communication of design and production information in the shipbuilding industry. This approach focuses on the underlying information, i.e. not just the graphical presentation of the design in drawings or 3D graphical models. The database has an object-oriented approach so that all data are stored in the form of "objects" such as systems, assemblies, pipes, equipment, brackets, cables, stiffeners, etc. Once the design of the main hull envelope is finalised in TRIBON Initial Design, the building of the Product Information Model is a refining process of the information registered in the system. The principles of concurrent engineering are well catered for and, in later stages of the design, it is easy to refine the already registered information. Furthermore, the database promotes the integration of data so that new information is instantly available to all users in each discipline. The database structure has been designed so that information can be extracted across a whole project by specifying from a wide variety of selection criteria. Clearly the value of this information can further be improved by making use of it in new application areas and by exploiting the available data in even more detail.

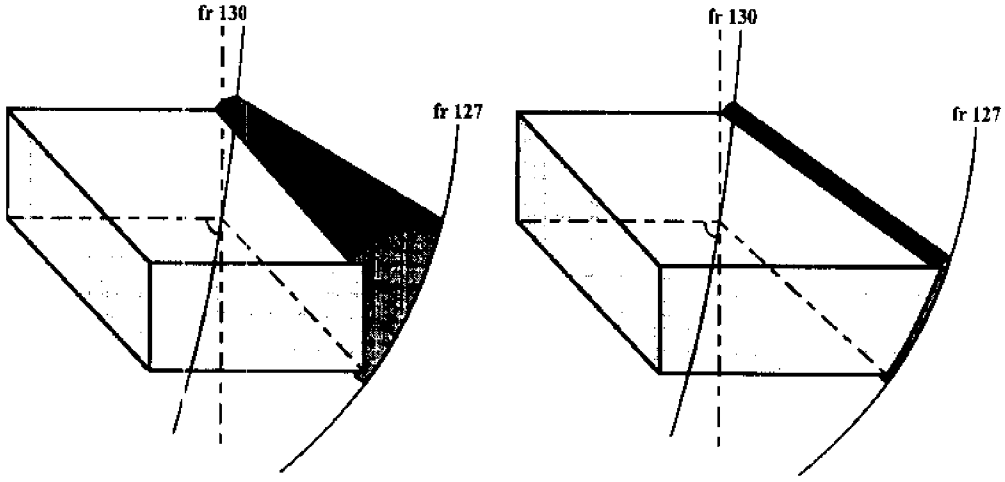


Figure 2: Illustrated Advantages of the Dotori Function in TRIBON

A third generation of design software is available from KCS. The design and production stages within the TRIBON system have been integrated and the production process can now directly access and refine the Product Information Model which is established during the initial design. KCS have adopted the underlying technology of ACIS (supplied by Spatial Technology Inc.) for the design of the main hull surface patches and their topology. This is an object-oriented approach using carefully defined entity types building upon the fundamental, standard building blocks supplied by ACIS. The **SURFACE** module of **TRIBON Initial Design** is now well defined as the basis for a Product Information Model to be used in the TRIBON production software suite. A module for the interactive definition of compartments has been developed as a natural extension of SURFACE; **COMPARTMENT** gives an object-oriented database of the connected major surfaces of a ship and their interrelationships.

Together with certain key partner shipyards and under a high priority development project, KCS are further developing **TRIBON ISD** (for initial structural design) as a concept design tool to assist in the detailed design stage. The user is able to define the main internal structure of a ship hull without explicitly creating this detail in the sense of traditional CAD; the structural elements are defined using a symbolic representation. The work at the detailed design stage is expedited by providing a model which can be used for rapid prototyping and is convenient for the design process. By referencing libraries of standard parts and representing them symbolically rather than explicitly, standard drawings for approval purposes and calculations such as the shell expansion, weights, etc. can be obtained. Broader aspects of the design process, including the initial definition of work content and duration can be covered and also, by linking object data to finite element calculation processes, strength calculations will be facilitated. Subsequent to successful analysis, product data are immediately available to the production modules of TRIBON, i.e. avoiding redefinition of the structural components. The information flow will provide data in a format convenient for the redesign or approval process.

TRIBON ISD is able to deal with structural items on different levels of complexity and allows for the definition of individual items and groups of items, including surfaces, plates, blocks and zones. ISD is a soft, open rule-based system where the attributes of structural entities are calculated from algorithms or from rules which the user can define. The interactive generation of objects using computer graphics and the symbolic representation of the structural items on the graphics screen is possible. This enables the user to view part of the model with a limited amount of associated information prior to its explicit definition. Sample output is given in Fig. 3 which shows an arbitrary block breakdown of the centre cargo holds of a bulk carrier.

In December 1998 KCS embarked on a software development programme that will effectively promote the implementation of STEP AP218 translation technology within ISD. This new initiative involves the implementation of an Object Interface framework designed to handle the necessary interactions that exist between ISD, the **TRIBON Hull** data modelling engine, configuration management functions and data retrieval and storage processes (Fig. 4). The proposed framework architecture will comprise a number of software 'components'. Collectively, these will provide the necessary degree of independence between each TRIBON application, the data modelling kernel, translation interfaces and the chosen database implementation. When completed, ISD will be able to directly utilise the AP218 STEP interface as currently implemented in TRIBON Hull. Thus a STEP interface will become available for data exchanges between ISD and Classification Society approval software in the early phases of the hull structure design process.

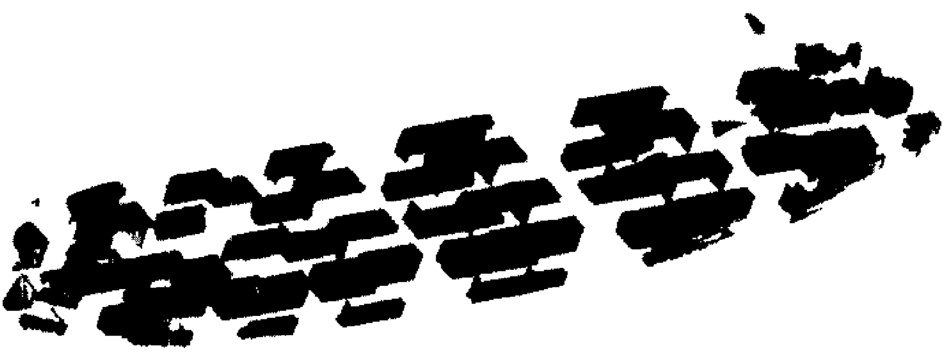


Figure 3: Example Block Breakdown Output from TRIBON Initial Structural Design

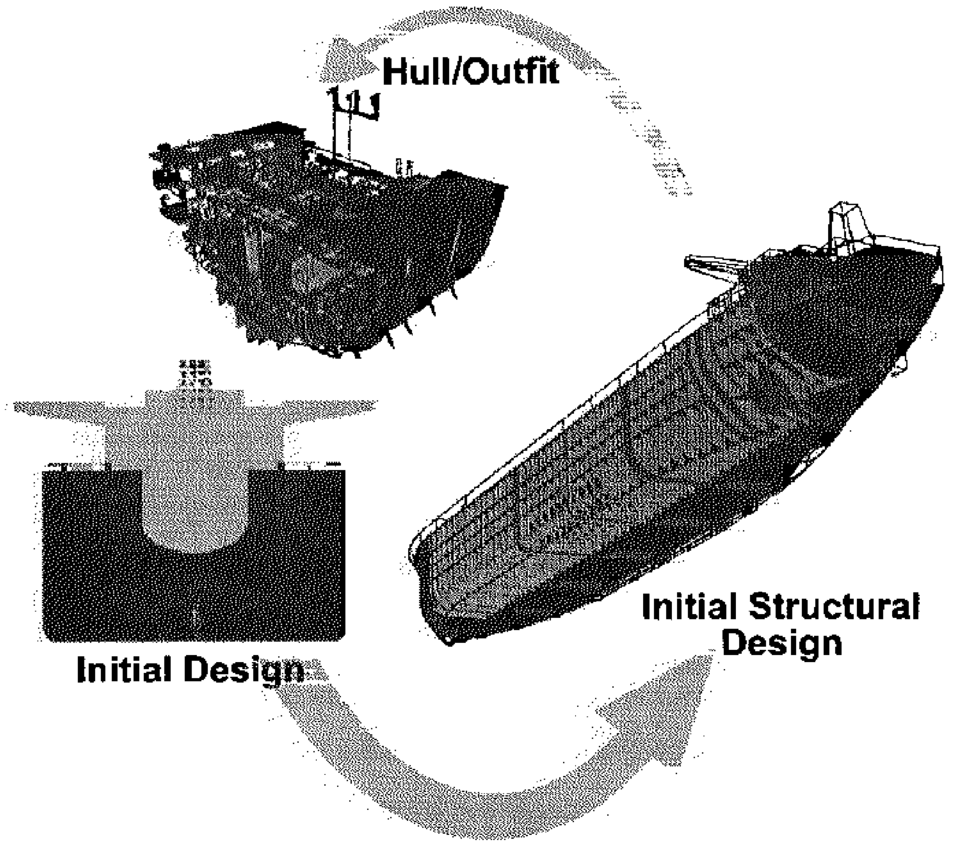


Figure 4: Integration of TRIBON Initial Structural Design

With the release of TRIBON Version 4, the openness of the software is improved through a new, open application programming interface, named **TRIBON VITESSE**. This facility is based on the interpreted language Python and it is now included for hull modelling. Release 4 of TRIBON also includes initial design surface geometry modelling based on ACIS and with isophotes for assessment of surface quality, extended naval architectural functionality, a consolidated hydrodynamics module, a new preliminary structural design module, upgraded work preparation module, colour shaded modelling (using OpenGL) and a new **Production Data Interface** with assembly browser, initially covering pipe and assembly data but extending progressively to equipment, hull, cable and structure. Other example, on-going key developments are: support for knuckled panels, shrinkage handling, "Quick Nesting" and robot and theodolite control interfaces.

Data Exchange Standards

As outlined in the preceding sections, for a modern shipyard, efficiency and reliability of data storage and data communication to cover the lifecycle of a ship is now of paramount importance, especially for usage scenarios in design through to production. Typical scenarios require co-operation between designers, owners, shipyards, model basins, approval authorities, sub-contractors and machinery and component suppliers. Increasing standardisation of the data and its presentation are focusing on the need for a Product Information Model which is flexible to the special purpose needs of the shipbuilding industry. STEP compliance of data structures will be of assistance, especially for the long term storage of ship data (some 30 years of operation is demanded); it can be expected that increasingly more use will be made of a Product Information Model for lifecycle aspects.

Commercial ship owners, navies and shipyards have the responsibility for design, acquisition and life cycle support for a vessel. During its life cycle, design agents, shipyards, equipment suppliers, the operators and classification societies have to fulfil this responsibility. The wide variety of functions carried out by each of these interested partners, often distributed over different countries world-wide and using various types of computer-based systems, emphasises the need for international standards within the maritime industry.

Archiving and retrieval of product information involves accessing a growing mountain of related data items as they are created and modified during the evolving design. The increased use of CAD has led to a data explosion and potential significant data management problems for all shipyards which have invested in the technology. Today the complete Product Information Model data for a commercial vessel can be expected to be in the range of two to ten Gigabytes, depending on the complexity of the vessel and the coverage of the digital model. As systems become even more powerful this figure will increase considerably. For use of product data beyond the design and production part of the life cycle archiving is also of interest. Well organised recording of the design versions of the data is essential.

The use of an **integrated system** is recognised as the most efficient base for operations within a shipyard or design agency but the increasing use of specialist sub-contractors and partner initiatives implies that the need for openness must be satisfied. In response to increasing demand for standardisation and openness of modern computer-based systems, protocols are now well advanced for the definition and handling of convenient data structures. It is anticipated that, as a result of international co-operation projects such as, previously, ITiS, MARITIME and ShipSTEP (11) and, currently, SEASPRITE, MariSTEP and Calypso, designers, model tanks and shipyards will, in the near future, have common access to improved data models for the convenient exchange of ship product information.

The previous collaborating research projects indicated the potential of the STEP methodology to integrate different systems used in the design and production process chain. This work has now been much extended under the current projects with more extensive prototype exchanges being demonstrated and formally tested. Major progress has been made; implementation schemas for the main shipbuilding applications protocols (AP's) are now available as a result of the above activities, see Figure 5. Dedicated work from those attending the regular ISO meetings and international co-operation between the major groups is encouraging, although there remains some sympathy for a viewpoint taken from outside of the STEP community that it may still be difficult to appreciate the real progress being made.

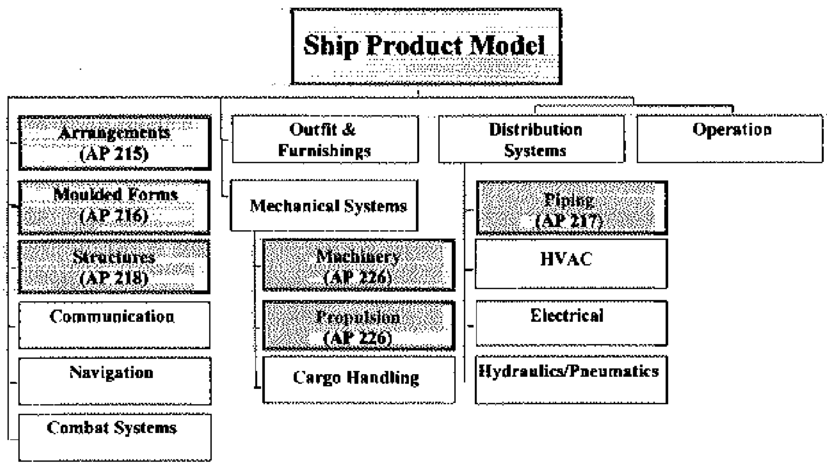


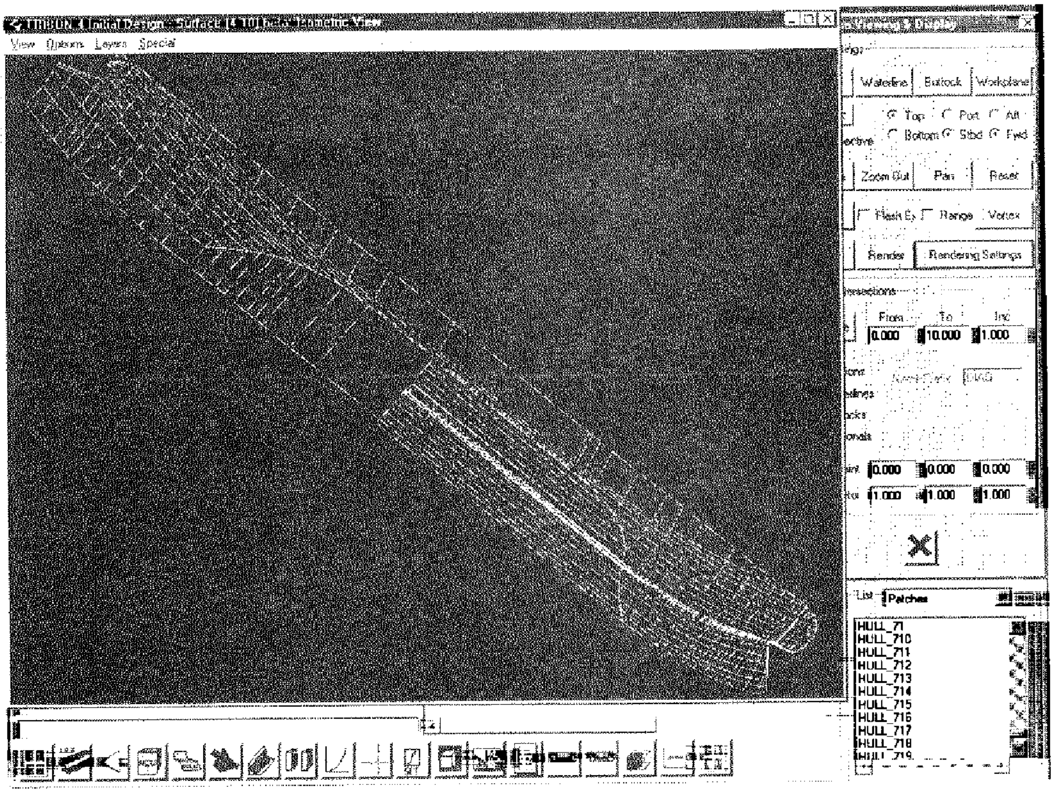
Figure 5: The STEP Ship Product Model Applications Protocols

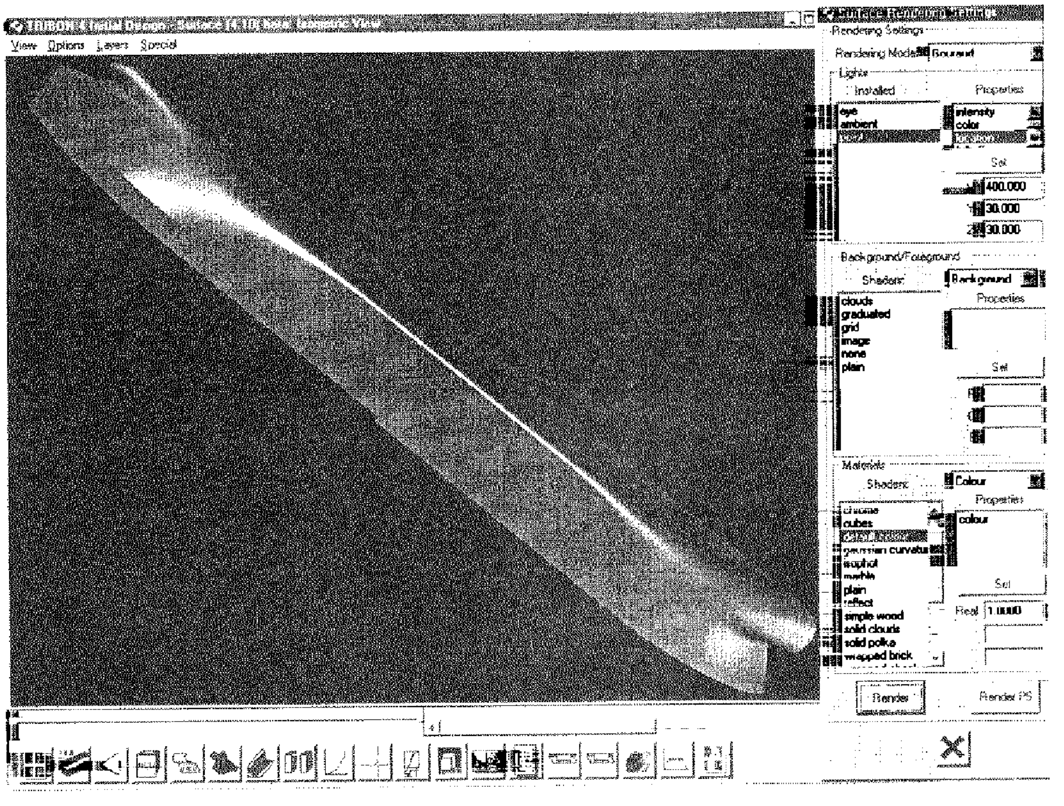
Generalisations used in the building block approach of the Ship Common Model are aiding standardisation across the various AP's (1). In a practical implementation of STEP, keeping track of the inevitable design changes and their management with version control is most important. Efficient representation of a ship structure requires a model which is easy to change and maintain. Key aspects are parametric and topological modelling.

In **SEASPRITE** (Software Architectures for Ship Product Data Integration and Exchange), prototype moulded forms exchanges between TRIBON Initial Design, NAPA, GMS (from Marin) and Seasafe (from LR) have been demonstrated. One of the business case scenarios has focused on minimising human intervention in the exchange of data between shipyards and model basins. The approach ensures that, following a typical series of data exchanges, the design is more consistent than that produced by human operators. The adopted format is based on a subset of the now fairly stable AP216 Applications Reference Model (ARM). Results of the test exchanges will validate the Shipbuilding ARMs and provide input into the further development of the STEP APs; (see Refs. 12,13 for AP215-6).

Figure 6 illustrates surface data which have been exported from the NAPA system and imported into TRIBON. The upper part of the figure shows the boundaries of the imported 719 simple, bi-cubic B-spline patches which is the NAPA surface representation of the semi-hull of the example VLCC based on their connected wire-frame model. The lower part of the figure is a rendered view of the imported surface in TRIBON using multiple light sources but without any further processing of the surface data.

Fig. 6Typical AP216 Moulded Form Data Exchange





For this part of the project, pre- and post-processors have been written by KCS using a late binding approach. The EXPRESS-X mapping facility (EDM STEP toolkit) has been applied to define and utilise a convenient, customised intermediate EXPRESS schema. For the AP218 related part of the project, on the other hand, (a first successful exchange of ship structural data under SEASPRITE was reported in January 1998), KCS have adopted an early binding approach (ProSTEP/GIDA toolkit) and are developing a customised facility for efficient mapping between C++ data structures. Thus the merits of alternative technologies are being assessed by KCS within the SEASPRITE project.

The partners of the further EU project Calypso (Computational Fluid Dynamics in the Ship Design Process) decided to adopt the SEASPRITE AP216 model as a standard for the exchange of hull surface and appendage geometry data which may be accessed by alternative flow analysis packages. This project has further extended the practical application of SEASPRITE. The CFD tools require information about the hull shape to be translated into a proprietary format for further processing. Marin are using STEP to communicate hull moulded form data. On the other hand, Flowtech have adopted an Applications Programmer Interface which has been written by KCS as a further level (on top of STEP) to enable the interrogation of the hull geometry from TRIBON or an alternative geometric modeller when generating surface grids to be used subsequently in their volume grids. Figure 7 gives a schematic view of the data flow which is planned to be completed and then utilised with plug and play facilities via a Designer's Workbench before project completion in summer 1999. With valuable input from Frank Stolte (14), the basis of a hydromechanics model was also written as an EXPRESS schema for Calypso and this has been adopted for consistency across the data structures of the Workbench interface and the database.

Primarily, the Designer's Workbench is a workflow system supporting the process of designing a ship hull, especially with respect to integration of CFD tools. The approach, however, is to integrate ALL relevant systems into one single environment, and enabling unified parameter generation and passing to generalised categories of tools, i.e. the geometric modellers, the mesh generators and the CFD solvers.

Direct Interfaces from Tribon Initial Design Geometry modellers to the CFD codes have also been developed These provide a quick and easy method of transferring geometry data (offsets and patch distributions respectively) to the CFD codes. Where more sophisticated geometry or patch distributions are required, the model may be accessed via the API. As an important part of the project, validation work has been carried out to provide information from which the user of CFD tools can judge the applicability and accuracy of the CFD tools. This aspect is indispensable for a successful introduction of the CFD tools at shipyards. The information will be available in a so-called 'cookbook'. A validation archive will be set-up, containing detailed reference and CFD information. Users of the CFD tools intend to continuously update the validation archive, after the Calypso project. The validation archive may also be used as a starting point for future shipyard design work.

MariSTEP (Development of STEP Ship Product Model Database and Translators for Data Exchange Between U.S. Shipyards) is a collaboration between U.S. shipyards, software vendors, research institutions and the "virtual" shipyard represented by the U.S. Navy and the NAVSEA CAD-2 program. A System Requirements Document (SRD) specifies the STEP translator and product information model requirements covering the Data Model, Function, User Interface and Diagnostics. MariSTEP schema are defined for each of the APs 215-8.

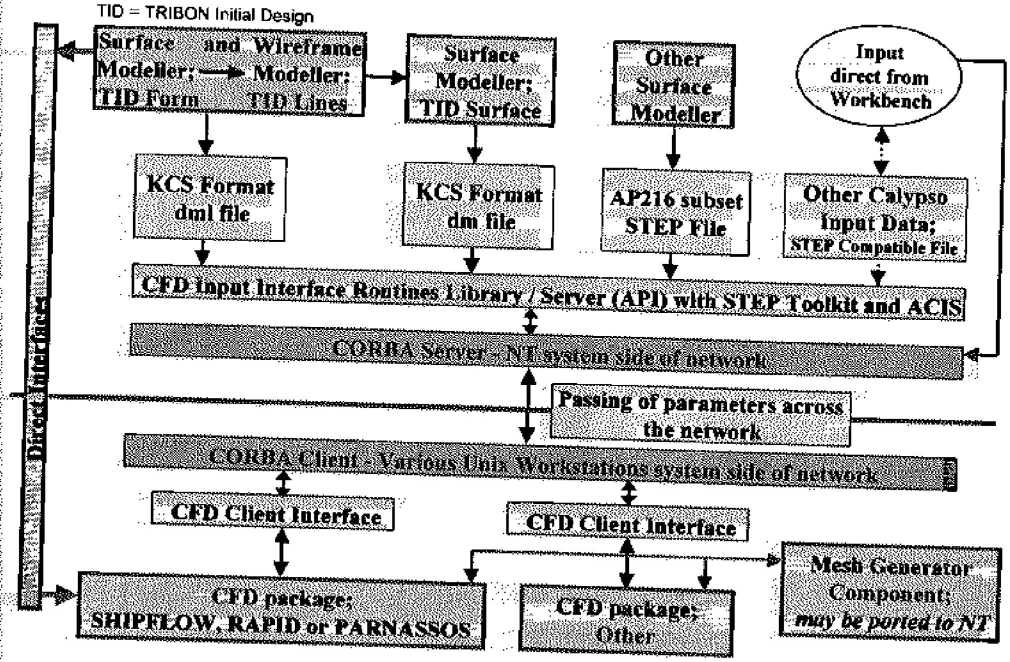


Figure 7: Calypso Project; Data Communications

During the project any changes which proved to be necessary were made to the SRD rather than the schemas. Thus, following project completion in June 1999, it will be necessary to update the schemas to reflect on-going changes in the Ship Common Model. The MariSTEP consortium thus proposes to continue its efforts to assist in standardisation of the product models needed for the shipbuilding industry and to implement translators supporting the exchange of this product model data.

KCS translator work associated with AP216 and AP215 within MariSTEP has utilised the EXPRESS-X intermediate mapping facility to minimise the impact on the database interface access coding of the differences between the SEASPRITE and MariSTEP AP216 schemas. The translators have been greatly extended for the more complete implementation of AP216, internal structures, compartments and zones. Figure 8 illustrates an example model in TRIBON created by importing a set of STEP test data files (AP216 and AP215) after export from the Newport News Shipbuilder's VIVID system. Data files have also been exchanged with Intergraph Corporation and are planned with Electric Boat Corporation and Ingalls Shipbuilding Inc. in the second track phase.

Future Perspective

Progress which has been made with shipbuilding solutions is now such that the commercial success of a shipyard is strongly influenced by the availability of an effective Product Information Model. This is the singly most important concept and applies not only to the large shipyards but also to the medium- and even small-sized shipyards.

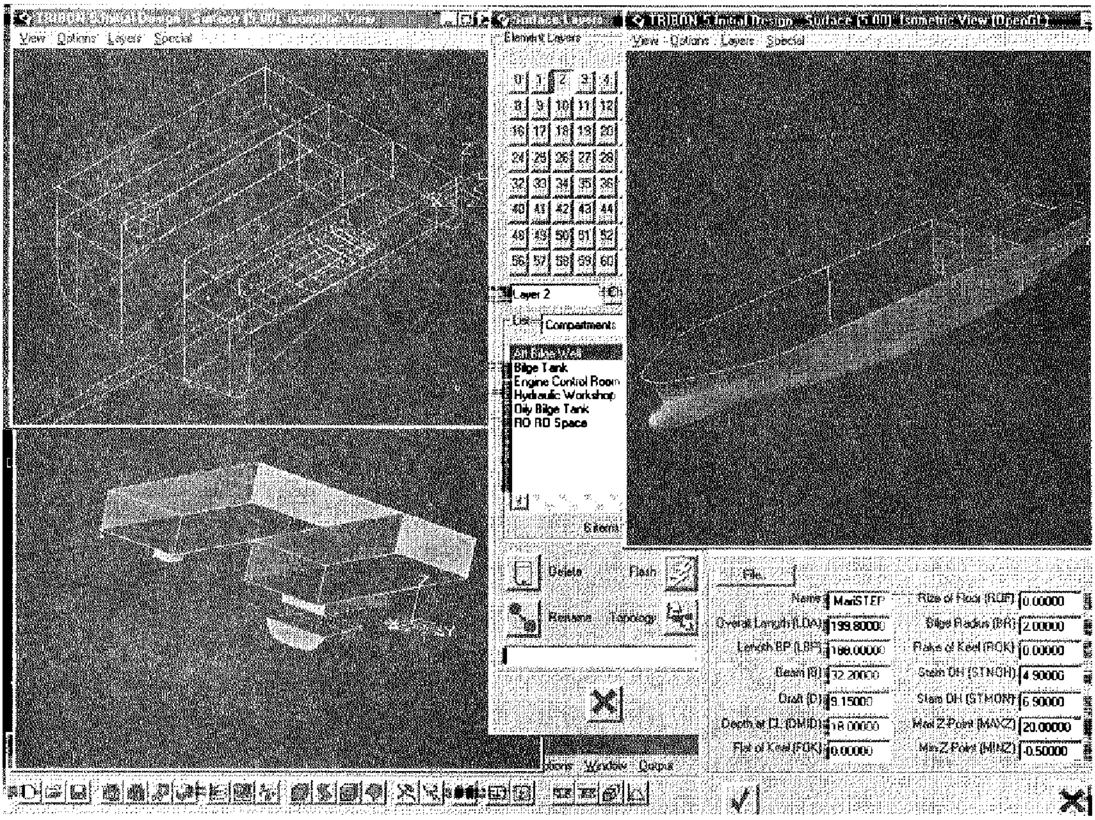


Figure 8: Import into TRIBON of STEP MariSTEP AP215 Data

Ship owners and operators will begin to take advantage from the availability of Product Information Model data for reference in maintenance and repair. STEP compliance of data structures will be of assistance, especially for the long-term storage of ship data. Prototype data exchanges have already occurred; exchanges on a far wider basis are now anticipated.

In the near future simulation-based design, e.g. structural design based on FEA, is expected to have a significant impact on the design process. The use of robotics in automatic production will continue to find new cost-effective applications. *Producibility of the product,* considering the manufacturing appliances and manpower of each production stage, will be assessed by virtual production, i.e. simulation prior to production. Virtual reality will be put into common use, especially for crew and maintenance training in hazardous environments.

Acknowledgement

The author would like to thank KCS for permission to publish this paper. Acknowledgement is also due to colleagues at KCS, ISSC committee members and the many co-workers in the research projects who have supported the work.

References

1. Grau, M. and Koch, T. (1999), "Applying STEP to Shipbuilding",10th Intl. Conf. on Computer Applications in Shipbuilding, ICCAS '99, MIT, June 7-11th, Pubs. Omnipress.
2. Catley, D. *et al.* (1988), "Design Optimization", *Report of V.5 Committee to ISSC.* Lyngby, Denmark. Also (1989): "Exec. Summary", *Jl. Marine Structures*, Elsevier Press.
3. Catley, D. *et al.* (1991), "Computer-Aided Design Processes", *Report of V.5 Committee to ISSC*, Wuxi, China. Publishers: Elsevier Press, Applied Science Series.
4. Hughes, O.F., *et al.* (1994), "Applied Computer Aided Design", *Report V.5 Committee to ISSC*, Institute of Marine Dynamics, Newfoundland, Canada.
5. Catley, D. *et al.* (1997), "Design Methods", *Report of IV.2 Committee to ISSC*, Trondheim, Norway, Elsevier Press, Applied Science Series, Eds. Moan, T. and Berge, S.
6. Shin, J-S. (1996), "Advancements made in Naval Ship Design at Daewoo with the Application of Computer-Aided Design", *Journal Society Naval Architects Korea.* Also translated from Korean by Shin, J-S. (1997) for *KCS Newsletter* No. 30, January.
7. Johansson, K. (1996), "The Product Model as a Central Information Source in a Shipbuilding Environment", *Journal of Ship Production*, **12:2**, 99-106.
8. Méthot, C. and Lovdahl, R. (1996), "From CAD to Product Information Model", *Mari-Tech, '96*, Ville de Quebéc.
9. Bennett, J.G. and Lamb, T. (1996), "Concurrent Engineering: Application and Implementation for U.S. shipbuilding", *Journal of Ship Production*, **12:2**, 107-125, May.
10. Catley, D. and Koch, T. (1995), "The Impact of New Technologies on Computer-Aided Ship Design", *PRADS'95*, **2**:1324-1336.
11. Goodwin, J.J. (1996), "ShipSTEP Project Final Report", *Lloyd's Register Project CTPA 40044, Systems Development Report* **96/51**, 10th December.
12. ISO Part 215 - "Application Protocol: Ship Arrangements", Working Document of ISOTC184/SC4/WG3 N531 (t12); Oct. 4, 1996.
13. ISO Part 216 - "Application Protocol: Ship Moulded Forms", Working Document of ISOTC184/SC4/WG3 N538 (t12); Aug.30, 1996.
14. Stolte, F. (to appear), "A Data Model for Ship Hydromechanics", PhD Thesis to the Institute of Marine Systems & Fluid Engineering, University of Rostock, April 1998.

OBJECT-ORIENTED WELDING INFORMATION SYSTEM FOR SHIPBUILDING

Ju-Yong Park, Korea Maritime University, Pusan, Korea
Byung Yoon Kang, Research Institute of Small and Medium Shipbuilding, Pusan, Korea

Abstract

Welding is the major manufacturing process in shipbuilding industry and has a great influence on the productivity and quality of ship construction. Shipbuilding process needs a lot of information and welding information is the important part of it. Welding information is required throughout entire shipbuilding process, from design stage to the final erection stage.

This paper aimed to develop a welding information system to provide the suitable information necessary for each stage of shipbuilding in an easy and rapid way. For this work, the properties of the welding information used at each stage of shipbuilding and the information flow were analyzed systematically in the viewpoint of the object-oriented paradigm. Welding information distributed in various sources such as classification society rules, practices and standards of a shipyard was integrated into the object-oriented information model. To get the high effectiveness and extensibility of the information system, database system was combined with the information model. Estimation of the quantity of welding material consumption, joint design standard, guidance for welding technique and welding procedure specification are the main output of the information system. Event-driven programming technique and graphical user interface were used to give the richer interactive user environment and the flexibility of the application.

Introduction

Shipbuilding is a highly integrated technology and belongs to fabrication industry. A ship consists of several hundred thousand of members. They are made by many kinds of material processing method and fabricated each other by welding. Shipbuilding process goes with a great information flow. A lot of information is required at each stage of shipbuilding process and they are closely related to each other. Therefore, an integrated information system is a very important tool to improve the productivity and quality in shipbuilding. Welding information is a major part of fabrication information. It is used over almost all stages from design to erection.

This research aims to develop an object-oriented welding information system for shipbuilding. For this work, the types and the characteristics of the welding information for shipbuilding were analyzed from the viewpoint of object-oriented method. The flow of welding information and its relationship to other information were also investigated. To describe and process this welding information, a suitable object-oriented data model combined with relational database was proposed from these works. The proposed data model consists of two base objects, the member object and the joint object, and the innumerable derived objects from them. The data which each object contains are saved in the database. The various combinations and operations of these objects according to the user's requirements or classification rules give the useful welding information such as welding material consumption, welding procedure data etc by calculations or data searching.

The welding information which is distributed in the different stages of design and production is integrated into this object-oriented data model. This model can be easily connected to the other information of the design and other fabrication stages. It can be used as a part of CIMS(Computer Integrated Manufacturing System) for shipbuilding.

Characteristics of Welding Information in Shipbuilding Process

Most of welding information in shipbuilding is given in the design stage although it is used in the fabricating stages after the ship design stage. The ship design cycle can be subdivided into four steps: concept design, basic design, detail design and production design. Each design step is directly related to stages of the shipbuilding process. The drawings and documents at each step are made by the correspondent design group such as hull form design group, structural design group, outfitting design group, etc..

Providing welding information is a part of the work of the structural design group. Welding information is prepared in line with ship structural design process, and inserted in the structural drawings step by step.

In general, a shipyard has several standards or guideline booklets for design and construction to get a lot of welding information more accurately and more comfortably. These publications are referred throughout the entire shipbuilding process.

Flow of Welding Information in the Shipbuilding Process

The complete ship structural design process and the corresponding welding information is shown in Fig. 1. Each design step has a different task and requires a different welding information. Concept design deals with the topology and overall geometry of the structure. Overall structural layout and principal dimensions are determined by general requirements such as beam and draft limitations, cargo capacity and others rather than by structural considerations. At this step, the quantity of welding material consumption is estimated and the information for special welding joints of the nonstandard structural members is also considered. Structural engineers communicated with hull construction production engineers concerning requirements for nonstandard structural joints.

The main works at the basic design step are the scantlings of all main structural members, drawing up the key plans and some naval architectural calculations. Size and scantlings of hull structural member is determined on the basis of rules and regulations of classification societies. The welding information decided at this step includes selection of appropriate steel grade, location of erection butts and seams and detailed joint shape for the major structural joints.

Detail design is concerned with standard building block, which should be heavily dependent on the shipyard practice and facilities. The information generated at the basic design step is refined at the detail design step, and the amount of detail information is greatly expanded. The leg length of fillet joints, joining location of each component and welding procedure specification code number or symbols are given at this step.

Production design is to complete the detail design and to prepare shop drawings used for the actual construction of the ship. All the joints which are not yet defined in the previous steps are supplemented. The instructions for parts fabrication and assembly are inserted in the shop drawings. At this step concrete welding procedure data which include welding sequence, welding conditions and inspection method should be provided.

Documents for the Welding Information for Shipbuilding

Welding information for shipbuilding is distributed in the various documents. The classification rule is the most important reference for the ship design. It does not, however, cover all design aspects, therefore, other materials besides the classification rule and various categories of information are necessary for the structural design.

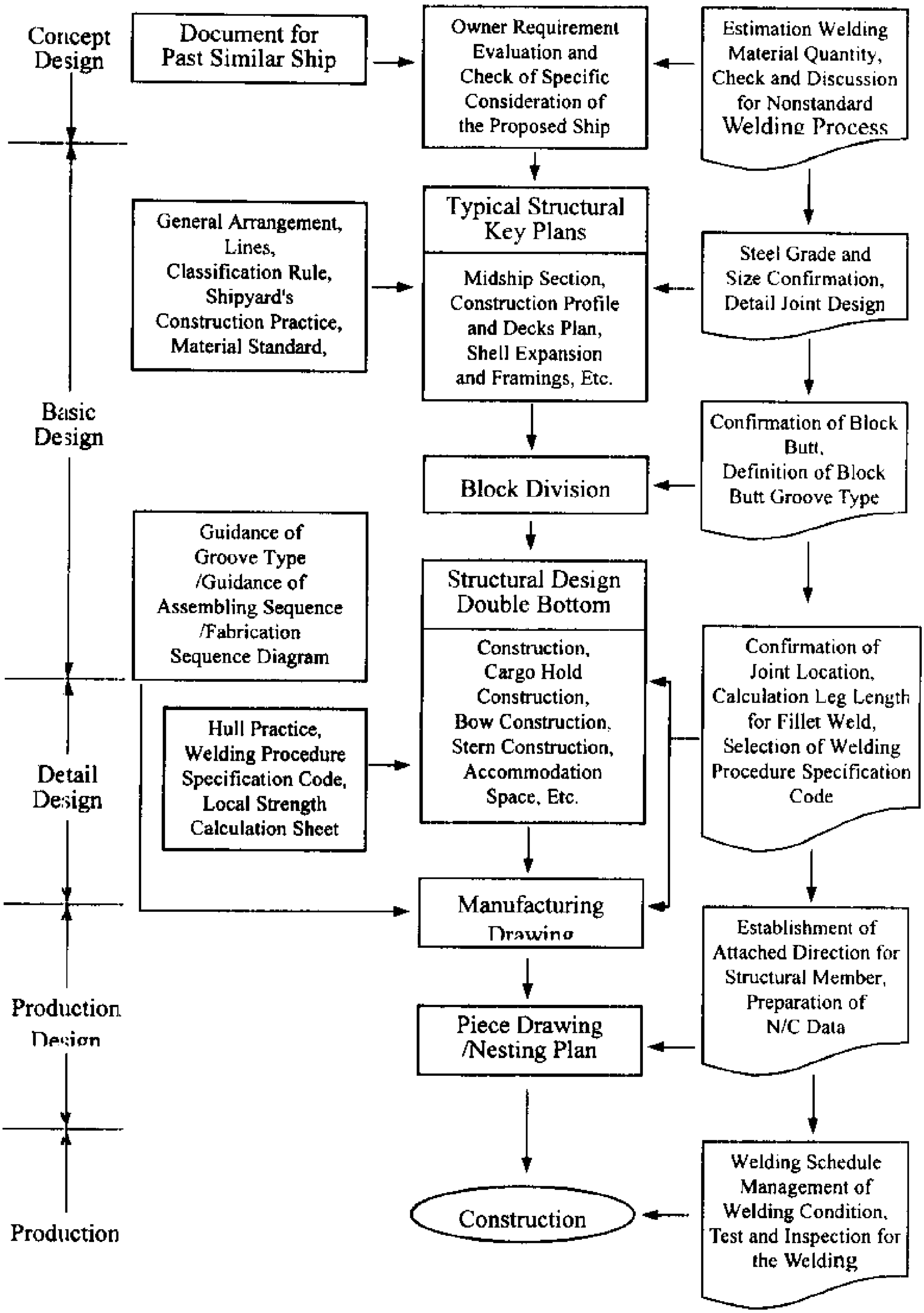


Fig. 1 Design step of ship structure and flow of welding information.

The goal of design and engineering is reduction of the production cost and it can be realized through standardization of design and construction practice which comes from the accumulated experiences and data. The standards are developed to accomplish the high design performance and quality. They can be used as a tool to convey the design information to the shop in concise manner. Selection of appropriate material, detail design of the joint and matching of quality assurance requirement are the main part of them. It is possible to design more reliable hull structure in a simple and efficient way by use of the standards.

Technical criteria and design guidelines concerning the welding information are formulated in the classification rule, the shipyard's hull construction practice and the standard welding procedure specification booklets.

1. Rules for regulations of Classification Societies

The classification society rules prescribe some requirements to ensure the hull strength, that is, steel grade and scantlings of structural member, rough guidelines of structural details, suitable welding materials, kind of fillet weld and fillet weld size. These rules are referred as a part of standards for design and construction throughout the entire shipbuilding process. Technical considerations and useful information for welding in the classification rules are usually incorporated in shipyard's practice to simplify design procedure and to eliminate time-consuming works

2. Shipyard's hull construction practice

This document is made in order to overcome an undesirable situation in shipbuilding procedure, such as repetition of the same work, errors in fabrication. It contains the valuable information and guidelines for the overall ship design and construction process, such as various symbols, abbreviations, joint details. Using this practice design process can be simplified and the design productivity can be greatly improved. For instance, it is very simple to choose the right size and shape of joint details for sound structural quality. Most of shipyard's hull practices include the welding practices which are described commonly in the form of welding symbol on structural drawings.

3. Standard Welding Procedure Specification booklets

Standard welding procedure specification is a document that includes the correspondent certification code number, and the detailed welding information, that is, groove shape, welding process, welding material, welding sequence, welding conditions and heat treatment. It serves also as one of the quality standards for hull structures. All of the welding techniques and data included in this document must be verified through the performance qualification. Welding procedure specification and performance qualification should be certified also by the classification society.

Modeling of Welding Information for Shipbuilding

Target and Scope of Modeling

For modeling of welding information the above-mentioned properties of welding information for shipbuilding must be considered and its target and scope should be decided to be able to realize the practical welding information system. The target and scope of modeling in this research are as follows:

(a) The information model of welding should be consistent with the flow of ship design information.

(b) The applicable scope of the pilot information system is confined to two unit blocks for a specific class, a specific ship type and size.

(c) Database is built to store the standardized welding information.

(d) It is focused to generate and to retrieve the welding information itself rather than the information which belongs to the ship design or construction

(e) The welding information served by this system is joint shape and dimension, welding process, welding sequence and techniques, code no and data of welding procedure specification.

Object-Oriented Information Model for Welding Information

The welding information model for shipbuilding consists of two basic objects and numerous derived objects from them. The two basic objects are 'member' and 'joint' object. The member object is a basic unit composing a hull, while the joint object is a basic unit fabricating two or more pieces. The member object is divided into two object, 'plate' and 'stiffener' as its derived objects according to its geometric shape. Hundred thousand of hull pieces are derived from one of them. They have the common attributes inherited from plate or stiffener object and their own information additionally given by their location in the ship, class requirements and so on. The joint object means a weld joint. According to AWS(American Welding Society), weld joints are classified into 5 types, butt, T, lap, corner and edge joint. Roughly speaking in a viewpoint of shipbuilding, however, the weld joints can be composed of two types, 'butt' and 'fillet'. Each type of joint objects contains at least two member objects and a lot of information related to welding such as joint shape, weld process, weld size and length etc. Each object possesses a lot of operation functions which determine the attributes of the object from various sources. They transmit a useful information to other related objects also. Fig. 2 shows the inheritance hierarchy of the objects and their aggregation relationship which are represented using OMT(Object Modeling Technique) methodology suggested by Rumbaugh.

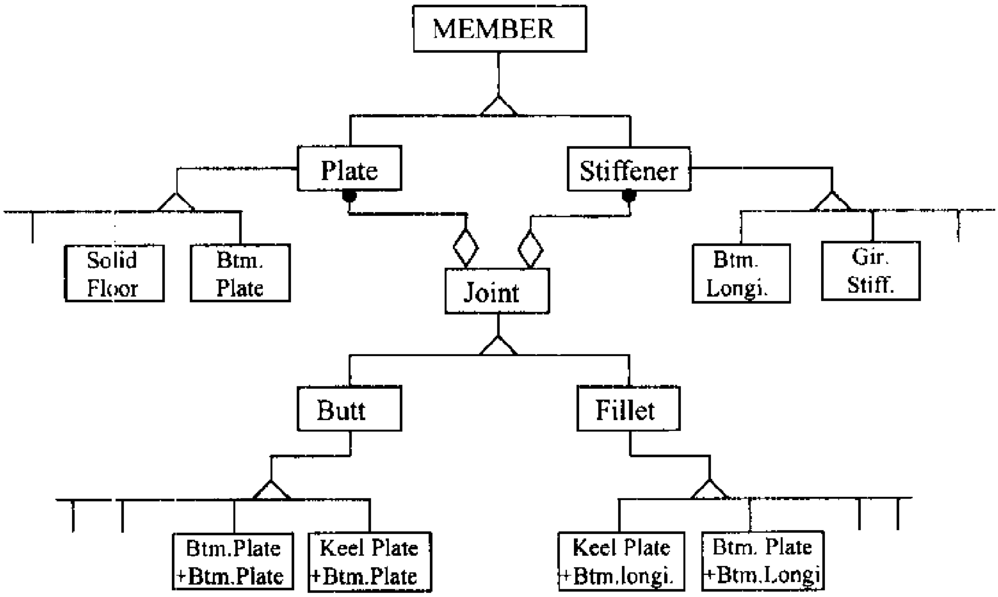


Fig.2 Hiearchy of objects and their aggregatin relationship

An object is created from a class which is a template for the object. The definitions of class for member and joint object and the derived ones from them are shown in Fig. 3.

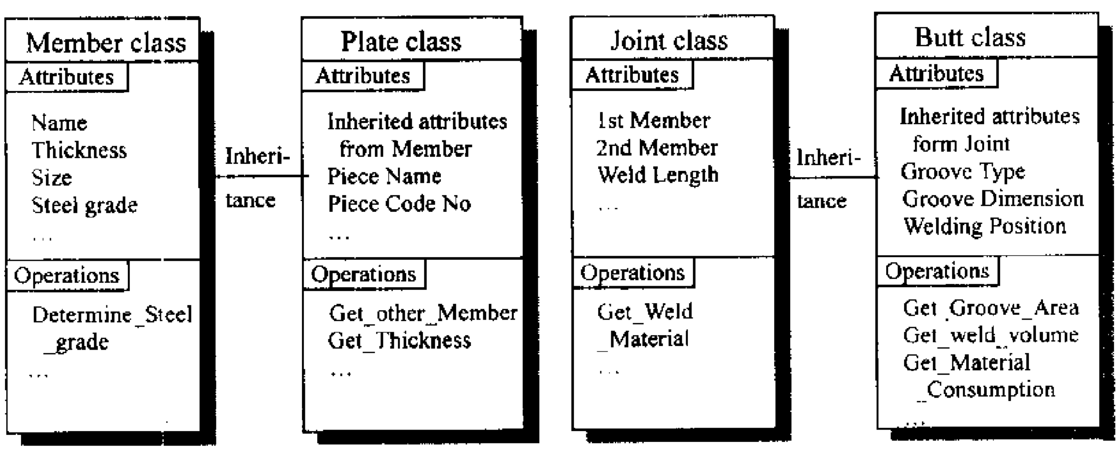


Fig.3 Definitions of classes for member and joint object

System Description for Modeling

The final result of this system is the welding information which contains concrete welding process, kind and size of welding material, welding procedure specification, the quantity of welding material consumption and other welding data. They are distributed in the several databases. They are searched from the databases and gathered into objects. They are presented as the final results through the modification and calculation processes according to the requirements of user. The components of the system is illustrated in Fig. 4.

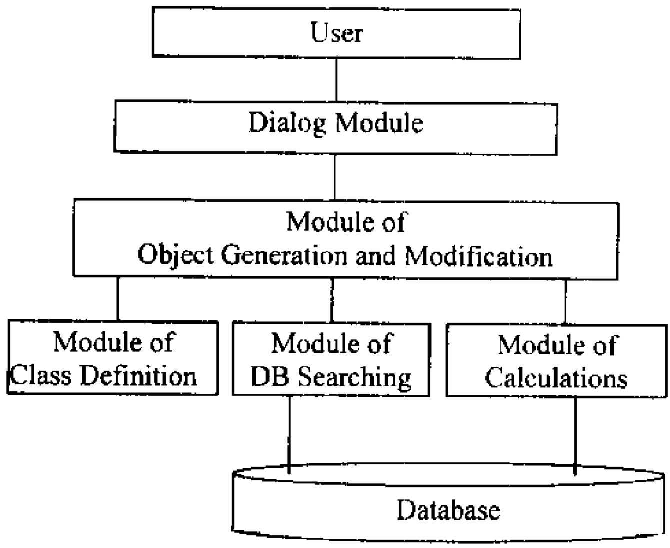


Fig.4 System components

1. Selection of an unit block of ship structure

As the first step, an unit block of ship structure is selected to get the welding information for the selected block. At this step, load state, available welding processes and correspondent rule requirements are determined from the rule requirement database.

2. Creation of joint and member objects

Joint objects are created by selecting the specific joints in the selected structure. At creating a joint object the member objects owned by the joint object are also created. The attributes of member objects such as member type, material type, thickness and size are determined from the values stored in the standard member database. These values can be changed according to the current ship type, size and owner's requirements. The standard values of the joint object attributes such as joint geometry, welding process and position, welding material and weld length are obtained from the standard joint database. They can be also edited by the same reason as for the change of member attribute values. This creation process is repeated until all the joints in the selected block of ship structure are created.

3. Creation of welding information

The final step is creation of welding information for the selected joints. This information comes from the various sources. Weld size and welding method of fillet and butt joint is given in the requirement of rule and regulation and hull construction standards. Some queries can be sometimes introduced to determine a definite solution of available choices. Welding procedure data such as welding condition, pre- and postheat treatment, back cleaning and so on are stored in WPS and PQR database. The estimation of the quantity of welding material consumption is calculated using the appropriate equation and the correspondent coefficients stored in the special database for this purpose.

Application Test of the System

The developed system was tested to verify the rightness and the usefulness of the output result. This system can be used double bottom and side hopper tank block in the midship section because the database of this system has only data of those blocks up to the present.

Development Tool and System Environment

The welding information system was developed using the following tools and system environment.

- Hardware: IBM PC compatible/Pentium
- OS: Windows NT
- Languages: Visual C++ and Visual FoxPro Language inclusive SQL
- Database: Visual FoxPro

Test of the System Step by Step

1. Selection of an unit block

Fig. 5 shows selecting an unit block of 5 unit blocks in the midship section of hull. When an unit block is selected, the correspondent rule requirements are searched from the database and the next detail drawing of the selected unit block is prepared.

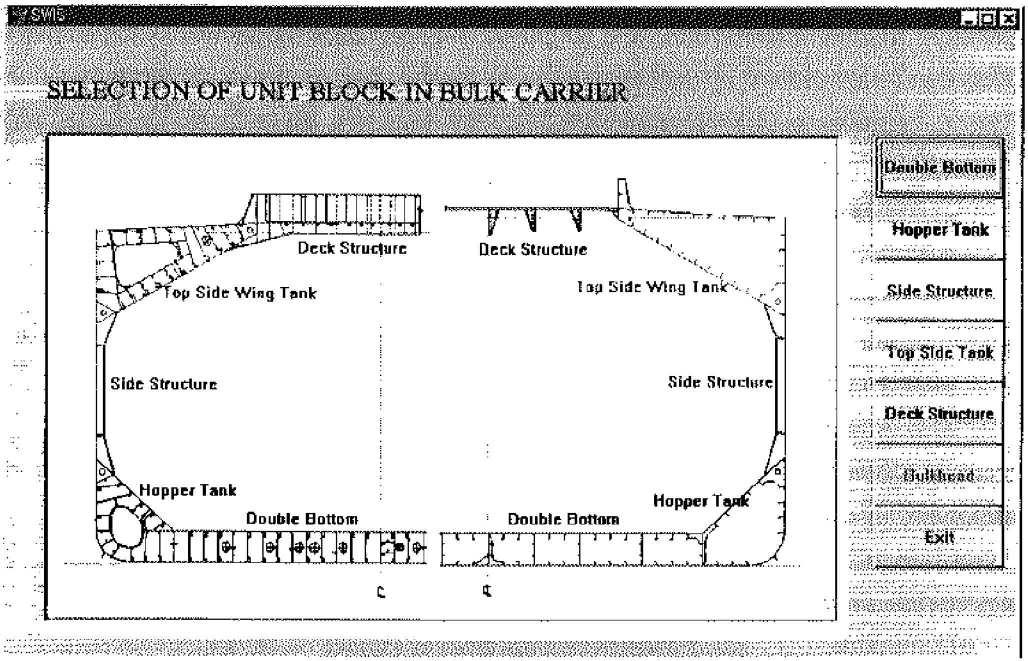


Fig. 5 Selection of an unit block of ship structures

2. Creation of member objects

Member objects are created when a joint object is created. In this system all the member objects are created before the creation of joint objects for higher efficiency. The creating process of member object is shown in Fig. 6. The number on the drawing corresponds to a hull piece, that is, a member object. Clicking on a number creates a member object and enrolls it in the grid in the right side of the figure which contains the already created members. Steel grade or thickness of the created member objects may be changed as shown in Fig. 6. The non-standard members such as special brackets, collar plates and other auxiliary pieces, which are not presented in the drawing, can be also created.

3. Creation of joint objects

The creation of a joint object by clicking on the number on the drawing is illustrated in Fig. 7. The type of member objects of a joint and the type of the joint are already known by its location in the block. The detail information is therefore entered only by determining a combination of the available members for the selected joint as shown in Fig. 7.

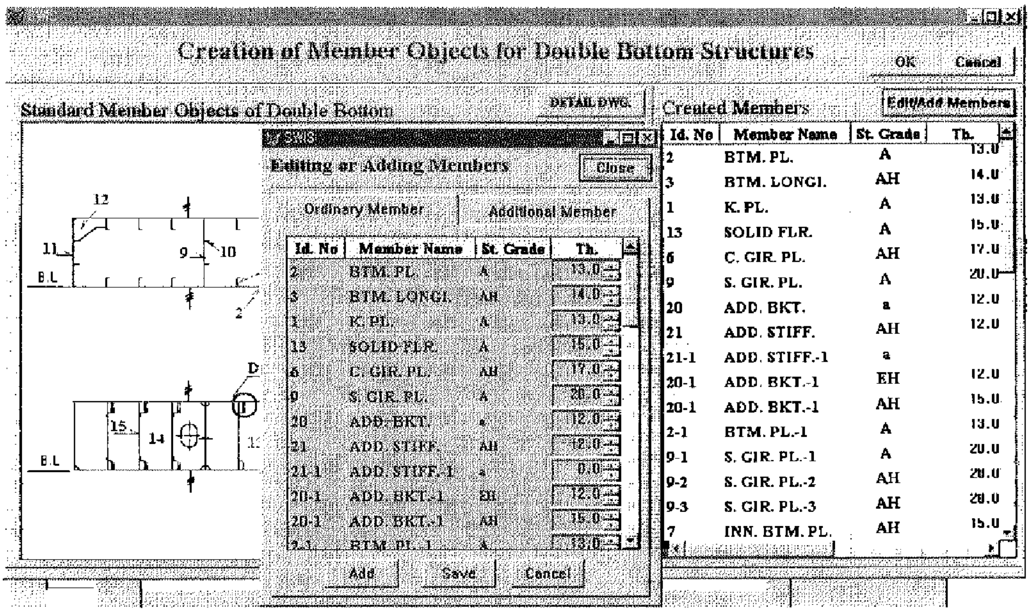


Fig. 6 Creation of member objects

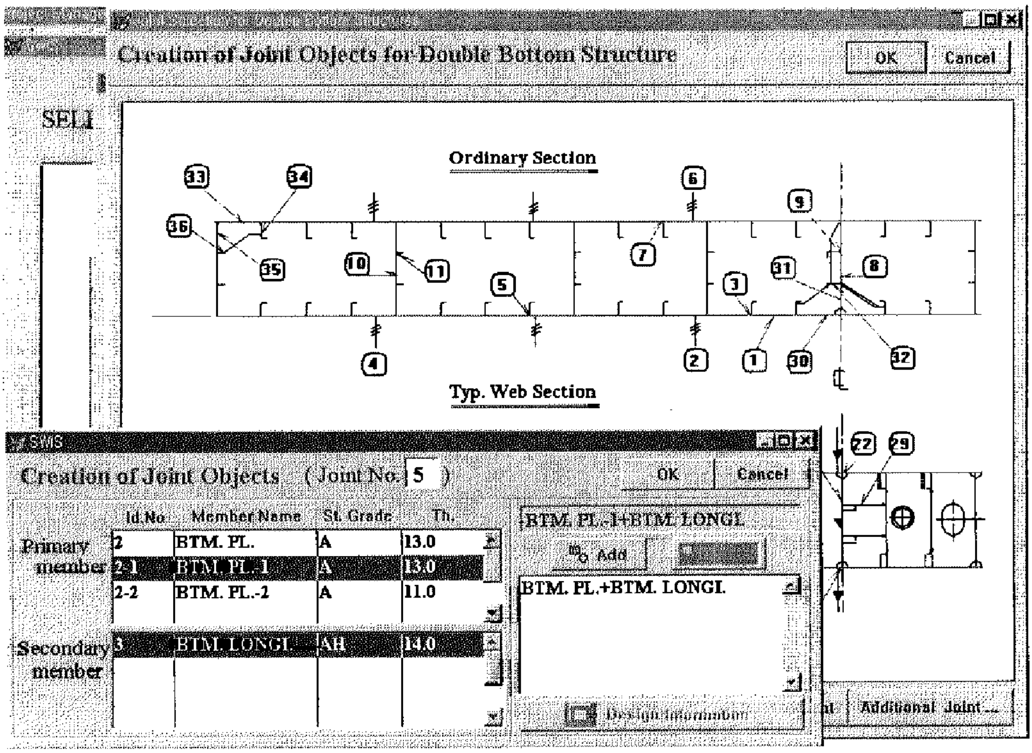


Fig. 7 Creation of joint objects

4. Obtainment of welding information

The next step is to obtain the suitable welding information by searching the database and to assign it to the attribute values of the joint object. To get the detailed welding information, several dialog steps of query and answer are given. Groove shape, fillet weld type and fillet size, welding process, welding position, other welding techniques and compartment requirements are determined through the dialog processes interactively, as illustrated in Fig. 8. On the basis of these data, the concrete welding procedure data are searched from the database and presented as the final result, as shown in Fig. 9.

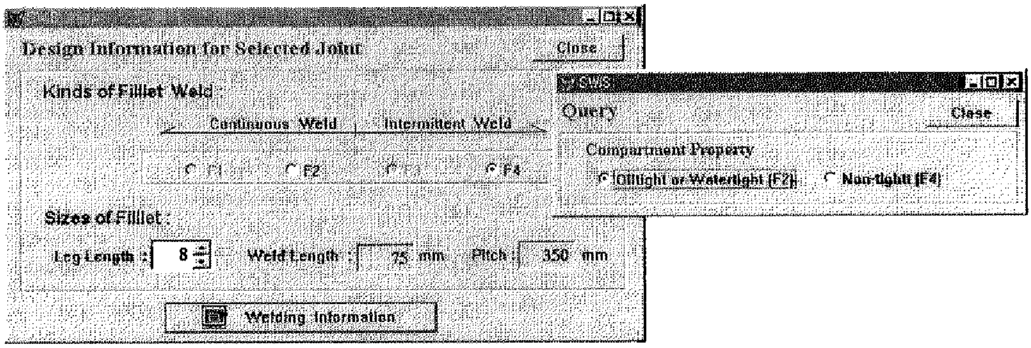


Fig. 8 Dialog for searching the welding procedure data

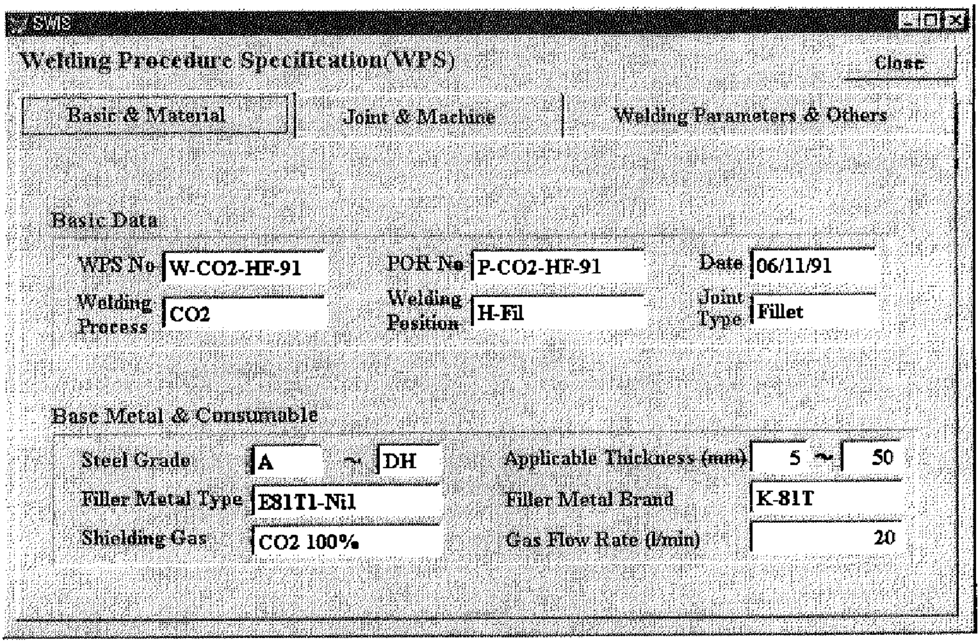


Fig. 9 The welding procedure data as the final result

Conclusion

Welding information is required over all stages of shipbuilding process and should be given at the design stage. Therefore, it is a very important task to develop the effective information system which is able to provide the appropriate welding information and to be easily used by a designer. This paper proposed an object-oriented information model combined with database system for welding which can give various welding information necessary for each step of shipbuilding. This information model consists of two basic objects and numerous derived objects from them, which contains a lot of properties and operations to store and to process the welding information. The welding information distributed in the various sources could be integrated by using this information model. The use of object-oriented technology and the integration lead to effective generation and easy handling of information, consistency of welding information with the information in various stages of for design and construction. Additionally, the database system connected to this welding information system made the storage of the vast welding data for shipbuilding and the easy and prompt search from them possible. In the future, the parametric modeling technique will be introduced into this system and CAD system for shipbuilding will be connected to this system.

Reference

1 Owen F. Hughes (1983), "Ship Structural Design, a relationally-Based, Computer-Aided, Optimization Approach", John Wiley & Sons.

2 Richard Lee Storch, et al (1995), "SHIP PRODUCTION", second edition, Cornell Maritime Press & Inc.

3 Robert Taggart (1980), "Ship Design and Construction", The SNAME.

4 Reonard P. Connor (1987), "WELDING HANDBOOK", Eighth Edition, Volume 1, American Welding Society.

5 Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., and Lorensen, W.,(1991) "Object-Oriented Modeling and Design", Prentice Hall International Editions.

6 Donald. Keith Burleson (1995), "Practical Application of Object-Oriented Techniques to Relational Database", John Wiley & Son Inc.

Emerging Information Technologies

AN IMPLEMENTATION OF LARGE-SCALE PRODUCT MODEL VISUALIZATION IN SHIPBUILDING

Bryan Marz, Newport News Shipbuilding, Newport News, VA
Rob Lisle, Newport News Shipbuilding, Newport News, VA

Abstract

Newport News Shipbuilding (NNS) has investigated the latest emerging visualization technologies for interfacing with large product model datasets. From this investigative research, a solution has been reached for implementation at NNS to address the multi-operational needs of shipbuilding, including but not limited to operational analysis, preliminary and detailed design review, manufacturing, and training. The solution was designed for collaboration efforts, between local and off-site participants, and includes immersive capabilities. This paper will provide an overview of the visualization efforts undertaken at NNS and how visualization is being implemented within the shipbuilding process.

Disclaimer

Reference herein to any specific commercial products, process, or service by trade name, trademark manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by Newport News Shipbuilding. Statements and opinions expressed in this paper are those of the co-authors and do not necessarily state or reflect those of Newport News Shipbuilding.

Introduction

In the area of computer science, designers and engineers have always been faced with the problem of inefficient or lack of computational capabilities to help in design and analysis. Rapid changes in computer hardware technology are now allowing software engineers to greatly improve the functionality and usability of applications on a daily basis. One area in particular that is beginning to show great strides in improvement is the graphical visualization of simulation and analysis projects. In the past, NNS has used VIVID® for performing large scale visualization. VIVID® is a proprietary product modeling system developed at NNS specifically to handle the requirements of solid modeling for ship design and manufacturing. During the design of the US Navy's new fast attack submarine, SEAWOLF, VIVID® was successfully used as a virtual prototype to replace the traditional full-scale physical mock-up. Because VIVID® was specifically designed to facilitate concurrent system development in a solid modeling environment, it enables the designer to load and visualize an entire ship's compartment. This capability is fundamental for an efficient product model centric design process. Many products that compete with VIVID are now using third party visualization products to bring together an entire view of the composite design. This is extremely inefficient for product modeling on a day to day basis since the designer must continuously switch between applications. VIVID® offers a much better solution for ship compositing since the designer can always see the entire design. However, VIVID® is essentially a static environment. As the utilization of product modeling has grown, so also have the requirements to have the product model more representative of all aspects of the virtual ship. For this reason, NNS has undertaken an implementation of third party visualization applications to be used in the shipbuilding enterprise. The objective in this implementation effort is to augment the existing legacy systems with a dynamic environment. This dynamic environment will allow everyone in the enterprise to interact with or within the synthetic shipbuilding environment. This paper will discuss the process that was used to determine the appropriate technologies to achieve these objectives.

Approach

To begin the effort, a business model was developed to associated industry efforts, NNS user requirements and areas with the greatest potential return on investment. From this document, NNS developed a high-level business case for visualization implementation. During the industry research phase the following areas were studied:

- Business or user requirements
- Display Devices
- Software Options
- Hardware Options
- Infrastructure Requirements
- Security Issues
- Implementation Strategies

Business Requirements and Implementation Strategies

The top-level requirements for visualization can be easily stated as the technology that can enable products to be developed better, faster and cheaper. In order to understand how these technologies can be implemented one must first have a good understanding of your business processes and where visualization can enable these efforts. Essentially visualization technology is best deployed as a means of fundamental communication. Most people will agree that their primary sense of communication is visual. Consequently, NNS feels that the greatest return on investment (ROI) is in implementing visualization technology to facilitate your organizations processes for communicating product data information. The NNS business model is comprised of three primary tiers as shown in figure 1.

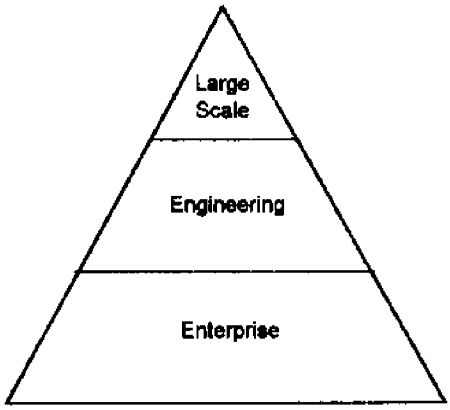


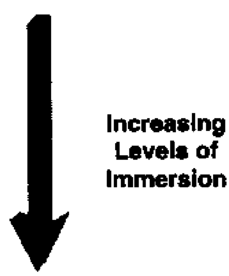
Figure 1. Visualization Tiers

The highest tier, which we call large-scale visualization, presents the biggest challenge. Large-scale visualization in this model is defined as the interaction of large volumes of ship product model data for large design and design review teams. Design review teams may range typically from around 12 to 60 people. Shipbuilding presents a scale problem like no other industry. A typical ship's machinery space may contain 10,000 parts equating to upwards of 10 million polygons. NNS has estimated that to visualize a complete NIMITIZ class aircraft carrier the technology would have to support models of greater than 10 million parts and files on the order of 200 gigabytes in size. Navigation speed is also an issue when working with large teams. If the system's performance is slow and there is significant time spent waiting for the display, teams tend to lose their focus and side meetings begin to occur. Therefore we can derive several requirements for the large-scale tier; accommodate very large models and large design teams. These requirements then decompose into specific requirements for facilities, display devices, hardware performance and software functionality. The middle tier, which we call the engineering tier, represents the personnel who are doing the day to day design tasks working in a product-modeling environment. Model sizes will be smaller and collaborative efforts may usually involve only 2 to 12 people. Consequently there is a different set of requirements for display technologies, hardware performance and software functionality. In the lowest tier, which we call the enterprise tier, our goal is to provide some collaborative visualization technology to anyone operating in the business enterprise.

Display Devices

There are a number of possible display mechanisms for visualization solutions. The options run from personal computer monitors to fully immersive virtual reality headgear. The following list provides a range of potential options:

- Workstation Desktop Viewing
- Large Flat Screen Viewing
- Large Desk Projection Systems
- Large Wrapped Screen or Immersive Viewing
- Stereoscopic Viewing (desktop, flat or wrapped screen)
- CAVES
- Helmet Mounted Display Systems



The most pressing business need for new visualization technology at NNS will be in the area of large volume visualization for electronic mockups with our customers. This particular area will most likely drive the performance requirements for both hardware and software. Very few people in industry are trying to visualize datasets as large as what is required for shipbuilding. The industry leaders in this area are the aerospace and auto industries. Our datasets will most likely be 1000X + the size of their typical datasets. NNS has considerable business experience in performing electronic mockup reviews. This approach was pioneered by NNS on the Seawolf program during the mid to late 1980's. During the Seawolf program, there were typically two levels of design reviews preformed: large audience space reviews and individual system reviews. Consequently, our primary driving visualization display requirement is to be capable of large audience viewing (12-60 people). Additionally, immersion technology provides a significant advantage in that it allows the viewers to be more enveloped in the visualization environment; and, thus, the effect or sense of spatial realism is increased.

For NNS, we have selected to implement large wrapped screen centers as well as flat screen centers depending on the need for immersion. All sites will be capable of stereoscopic viewing. We have chosen the Panoram System (http://www.panoramtech.com/) as the typical implementation. Figure 2 depicts a typical center.

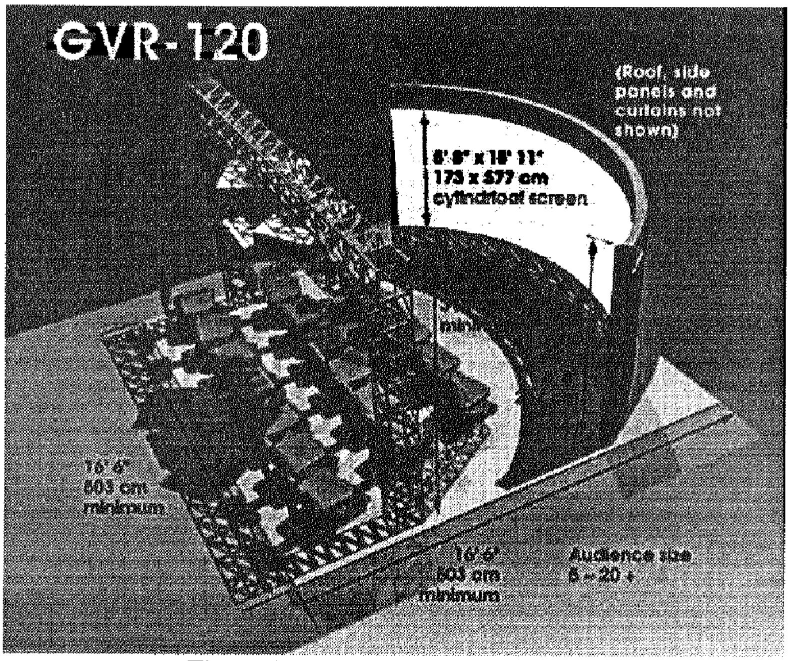


Figure 2. Panoram Display System

For the engineering tier we are implementing smaller flat screen single projection systems. Personnel compositing in the product model will also have workstations (both NT and Unix) that will be able to display large volumes of geometry. The enterprise tier will utilize common network PCs for displaying web based visualization applications. There are also a host of other display technologies that may be implemented; however, the intent of this paper is to focus on our core shipbuilding requirements.

through the ship interactively. This is needed in case anyone within the review wants to take a closer look at a compartment or piece of equipment. The benchmark that we are striving for is a minimum continuous refresh rate of 10 Hz.

Collaborative Review

A collaborative review is the ability of multiple users in multiple locations to review and manipulate a common model. This is becoming increasingly important with travel budgets dwindling and the desire to keep design team members informed with regularly scheduled meetings. Ideally, these reviews should require a minimal amount of preparation and special software, and should perform adequately on a mid-range engineering workstation. The objective is to be able to perform interactive distributed design reviews within a central model in conjunction with video teleconferencing technology. Requirements address areas such as cost, platform, ease of use for casual users, avatar technology, telecommunications technology, ability to coordinate effort, ability to create and collect comments as well as many others.

Equipment Removal

An important consideration of ship design is ensuring equipment can be removed with minimal impact to other ship components. Creating, maintaining, and presenting these equipment removal paths should be easy and time effective. The paths created should be able to be saved and used in conjunction with collision detection. The software should be able to simulate the removal process of ship's equipment There should also be mechanisms for capturing these simulations in video format for use later in training and life cycle support.

Collision Detection

The ability to perform analysis to determine location and extent of interference and collisions between components describes collision detection. This process should be capable of being performed parallel to other capabilities. Methods for collision detection should be a core feature of the software and contain some method of compiling and retaining collision data.

Manikin Technology

Manikin technology includes the use of manikins to obtain detailed information on human interaction with the product model. This includes items such as adding scale to the product model, logistics functionality (e.g. How effective are the planned routes from general quarters to battle stations?), ergonomics functionality (What can this person see and reach?) and physical stress functionality (Is this task physically possible and/or what are the limitations?). The manikin technology should address the multi-level requirements of different users. For example, still images of static manikins may be needed for technical manuals, simple human interface simulation may be needed by design teams and detailed anthropomorphic ergonomic analysis may be needed for safety and man-machine interface design. Each of these levels requires a different set of functionality. For example, the design teams working on human simulation within the product model do not necessarily need all the overhead of a full-featured ergonomics model. The design team needs quick straightforward answers whereas the ergonomics analysis needs very detailed answers. The two different applications will most likely result in different levels of time performance.

Desktop Visualization

Many hours of preparation are typically needed to design and build a large-scale presentation for use on a high-end computer. This preparation will likely be performed on desktop machines. Thus,

Hardware and Software Options

Initial research indicated that there were two potential options in the market for large scale visualization: SGI's Onyx 2 Infinite Reality Supercomputers and HP's new Pixel Flow Technology Machine being developed in conjunction with the University of North Carolina. However, at the time of our investigation, HP had decided to abandon the Pixel Flow Machine because there was not a sufficient business case. Therefore, the SGI Onyx2 Infinite Reality was chosen for our large-scale tier. There are many different visualization applications on the market today. It should be noted that the performance and functionality requirements for every organization might be different depending on the nuances of the business requirements that need to be filled. In general, NNS looked at a variety of different applications. For our business requirements, we grouped the applications into several categories as shown below;

- Large Scale Mockup and Simulation
- Virtual Human Simulation
- Enterprise Collaboration

Later in the paper, we will discuss detailed requirements for evaluating various software alternatives.

Infrastructure and Security Requirements

For NNS our primary requirements are centered around the design and manufacturing of aircraft carriers and submarines. These involve very large enterprises that must deal with both unclassified and classified information. Again each organization will have its own unique requirements. For this discussion we would like to simply point out that a fundamental piece of implementing visualization for many companies will require research, planning and development in network technologies and their associated security issues. These technologies may include:

- Computer Hardware Requirements
- Secondary Hardware Requirements
- Signal Processing Requirements
- Encryption Devices
- Internal and External Networks (Servers, Hubs, Switches, Fiber, etc.)

Software Requirements and Testing Criterion

In the process of determining visualization software requirements, a quality function deployment process was utilized to drive software requirements from core business needs. The following section will briefly discuss some of the major requirements for NNS. The software was graded based on twelve of NNS' most important visualization requirements. Each category had a weighted value and was given a score of 1-10. Short explanations of these requirements and the criterion on how they were scored are listed below.

High End Visualization (Real Time Fly-through)

The ability to smoothly fly through and manipulate large data sets constitutes a primary goal of visualization. The software must be able to move through large portions of ships in an impressive, real time manner with little or no hesitation. The intent of NNS is to construct several visualization rooms throughout the yard for reviews with integrated product teams. Many of these reviews will be upper-level management meetings where a review of the whole ship will occur. Being able to load a large portion of the ship, if not the whole ship at once, is important. The user should be capable of flying a requirement of desktop visualization is the ability to adequately run the software on a typical desktop machine with a large volume of data loaded.

User Interface

The user interface of the software can either guide the user through his/her options, or act as a maze that forces the user to endlessly search through a labyrinth of functions. It should allow an average computer user (with two to four hours of training) to create a replayable, simple fly or walk through. The controls should be straightforward, and should not require any special motor skills to use. In addition, the interface should have good on-line help documentation as well as a screen with the capability to be customized to suit the user's preferences.

System Integration

System integration is the ability to integrate the software with existing NNS software that will be the data source for visualization. How well will the visualization software fit into 'total visualization solution'? How well will the software fit into future NNS tools? How can the software be utilized to integrate the many different product-modeling applications used in the shipbuilding enterprise?

Data Integration

Data for projects may be supplied from multiple sources. The number of sources will likely increase as we move to a more collaborative environment. Data integration is the ability of the software to import data from these sources and integrate them into one cohesive model. Data Integration scales the entire visualization pyramid. It must be able to interface with multiple CAD systems, multiple PDM systems, and multiple graphics/product model standards such as VRML, STL, SLA, Inventor, DXF, IGES and STEP.

Cost

The cost of the software should be structured such that it supports the many different budgets and requirements of the organizations in the enterprise.

Customer Support

How well does the software manufacturer support the end user after software purchase? This is based on discussions with references supplied by the vendor or obtained through others means.

Business Viability

How well is the company currently positioned in the market? There are a variety of good sources that continually evaluate software companies relative to their technology, market share and balance sheet.

Benchmark Data

For each of the requirements, a benchmark was established as a method for comparing the different applications. This data represents empirical data gathered during the evaluation process. The scores are normalized to 10. The geometry/product model data should be based on an actual test scenario and model developed by the shipyard and supplied to each of the vendors being evaluated. Vendor supplied demos are wonderful for seeing the functionality contained in an application but it is our experience that the application must be stressed with real world data specific to your organization and industries processes. By developing your own bid package you can stress the application and determine not only its strengths but also its weaknesses. Once the applications are validated and put into the weighted evaluation, a composite score can be derived for each application.

Conclusion

The objective of this paper was to provide the user with a high-level example of how a shipyard can go about the process of implementing large-scale visualization. Emphasis has been placed on the front-end evaluation work that is required in order to bring together the best solution for your company. Shipbuilding presents a very big challenge for visualization technology. Ships are the ultimate products when discussing large-scale visualization. No other industry really compares. However, because shipbuilding provides such an enormous challenge, the existing hardware and software solutions on the market today do not satisfy all of the requirements needed in our industry. Nonetheless, visualization technology is only beginning to emerge and NNS believes that this technology along with other distributed network technologies will shape the processes of ship design in the future.

GEOMETRIC MODELLING FOR SIMULATION BASED SHIP DESIGN

John L Martin C Eng. MRINA, Marconi Naval Systems, UK

Introduction

Marconi Marine (VSEL) have implemented a Concurrent Engineering process with supporting information systems that are being applied to all current surface ship and submarine design build contracts. The process is also being used for concept design studies where Simulation Based Design technologies are being developed and evaluated.

A tree based product structure model is currently applied to ship detail design activities. This process has the functionality and data management capabilities for effective use on concept design studies where 3-D geometric models with associated behaviour attributes can be used to visualise the operational aspects of design options at very early stages of the design process.

3-D CAD modelling technologies and methodologies are well established throughout the shipbuilding industry as an aid to detail design, draughting, and manufacturing. Simulation tools and techniques are evolving for use in engineering and marine design to verify that design solutions meet operational or functional requirements. Combining the results of analytical simulation with 3-D geometric representation gives a high quality visual display for assessment of the design and its effectiveness in a range of operational scenarios. Using results of the simulation algorithms, the 3-D CAD model is animated in real or slow time to enhance visual presentation.

To achieve this the 3-D model must be constructed such that geometric objects may be associated with attributes, constraints and behaviour determined from physical limitations or operational specification. Simulation results can then be associated with the object geometry to visually illustrate the design in an operating condition.

Marconi Marine's VSEL site have created 3-D models for use in a simulation based design process applied to studies to assess the operational effectiveness of aircraft stowage, preparation and sortie management on future Aircraft Carrier. The modelling techniques are also used for creation of general arrangements in 3-D to show vehicle stowage loading and disembarking options on Logistics Landing Ship concept designs.

Marconi Marine (VSEL) use the product structure trees and concurrent engineering methodologies for the creation of 3-D ship models to create an environment which is suitable for implementing simulation based design applications on future ship concept design studies.

Simulation

An English dictionary definition of Simulate is to *"make a pretence of, copy, reproduce or replicate the conditions of a particular situation"*. Under this definition, the application of simulation based design tools to the ship design process spans a wide variety of situations

Simulation Overview

The most basic and simplest form of simulation is the creation of a 3-D model of the geometry of an object or set of objects that may be reviewed visually against some criteria such as spatial acceptability, shape, ergonomics, and/or operational feasibility. 3-D digital models have been created in engineering environments for many years as the alternative to a costly and time consuming physical model prototype. Model technologies used have evolved from full-scale wood mockups, through accurate scaled plastic models ($1/5^{th}$ and $1/20^{th}$ scales being common) to digital CAD and Virtual Reality models.

Enhancement of the 3-D models to enable animation of objects and their component parts increases the value of the visual simulation and provides improved verification of the design as selected objects may then be assessed within their envelope of operation. The animation movements range from the simple action of opening a door for maintenance access and reserving 'soft' space for the extraction of internal parts, to complex actions such as proving a route for the withdrawal of an item of equipment from a compartment for replacement.

Behavioral characteristics which are calculated using accurate physics based algorithms further enhance the animation when detailed analysis is required on a chosen design option against original requirement and/or performance constraints. A simulation of the operational aspects of the design under a range of scenarios will give a representation of the physical behaviour of components and verify their suitability.

Real-time simulation in which the object behaviour includes time constraints and characteristics will give a time-based analysis that reflects a true to life operation of the object. When several interdependent objects are operating as a logical group, real-time simulation will show the sequence of events of each object and their effects on other objects in the group. The inclusion of the time behaviour allows interaction between the simulated objects and human operators which may be used in human factors studies.

Discrete event simulation also uses the behaviour of the objects for accurate calculation of operational scenarios, however in this case the interest is in the logical sequence of events and interrelationships between objects and their status rather than the effects of time. Discrete events are analyzed using event rules and operational processes which verify that the design satisfies all of the constraints and conditions which must apply before any action can be carried out on the selected object.

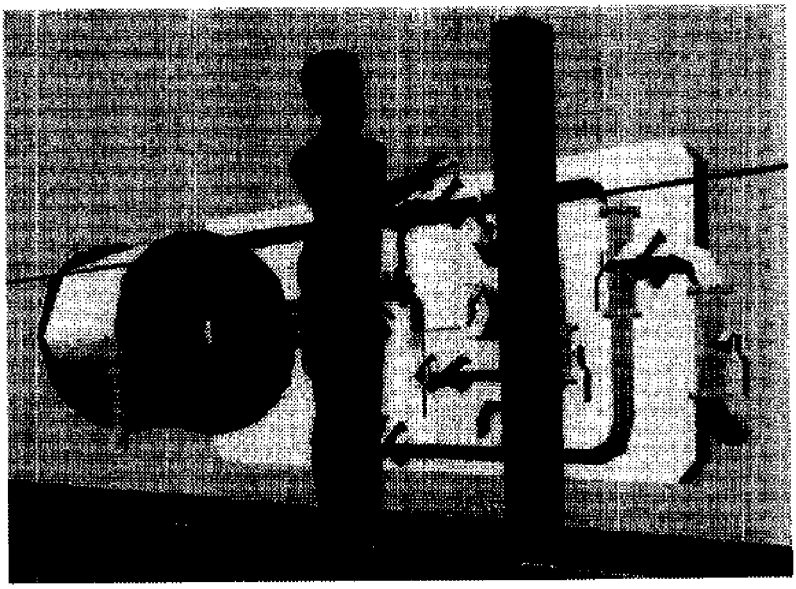
The analysis of human factor aspects of the design may be achieved by introducing a manikin model with behaviour algorithms that closely match human movements. For human operations to be assessed for operational effectiveness using simulation, the geometric models involved must have monitoring (dials, gauges, displays etc.) and control (hand-wheels, levers, switches etc.) components detailed and positioned accurately. Figure 1 illustrates an operating position in which the operator needs access to turn levers from a standing position on a platform.

Simulation tools have been used and proven over many years in engineering design and shipbuilding for study of specific aspects of the design. Merging the simulation results from these tools is a desirable objective, and with use of modern geometric display technologies can provide a visualisation of the operation of the objects being simulated and their interrelationships.

The Challenge

The challenge is to use an Engineering Data Management/Computer Aided Design (EDM/CAD) system, normally used for static detail design modelling to create geometry data suitable for use in dynamic simulation modelling systems.

Figure 1 Manikin Verifying Operating Position



To effectively visualise design simulation studies, a process is required in which the geometric models are constructed to enable attributes and behaviors to be allocated to component parts for use when animating the model during display. This can be achieved if the geometry is decomposed as hierarchical assembly components that are controlled and managed through a product structure tree.

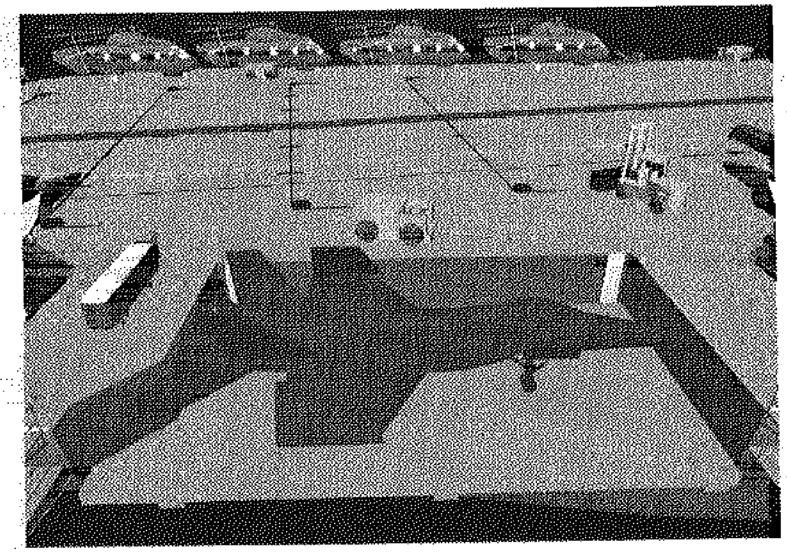
Behaviour Modelling

Ships and submarines involve many component parts that may have a simple or complex behaviour that can be verified by use of simulation tools. Figure 2 illustrates some critical items involved in the operation of an aircraft carrier flight deck which may be considered as items in which simulation can show their effectiveness during operation. Table 1 lists the model considerations required for each item.

Table 1 Models and behaviour considerations

Geometry	***Modelling Requirement for Simulation***
Aircraft folding wing	Aircraft Model split into body plus wingtips
Aircraft lift	Lift modelled as platform with corresponding opening in deck geometry
Wheeled vehicles such as aircraft tow-truck	Truck and tow-bar modeled separately, wheels modeled as component parts of truck
Fork lift truck	Steering wheels, forks modelled as component parts of truck, load is a separate model
Weapon stowage box containing missile.	Box model with separate lid and internal weapon
Aircraft Arrestor wires	Pulleys Modelled with separate geometry for wire
Helicopter rotor blades and tail – folded and operation position	Fuselage, rotors blades and folding tail section modeled as component parts of assembly

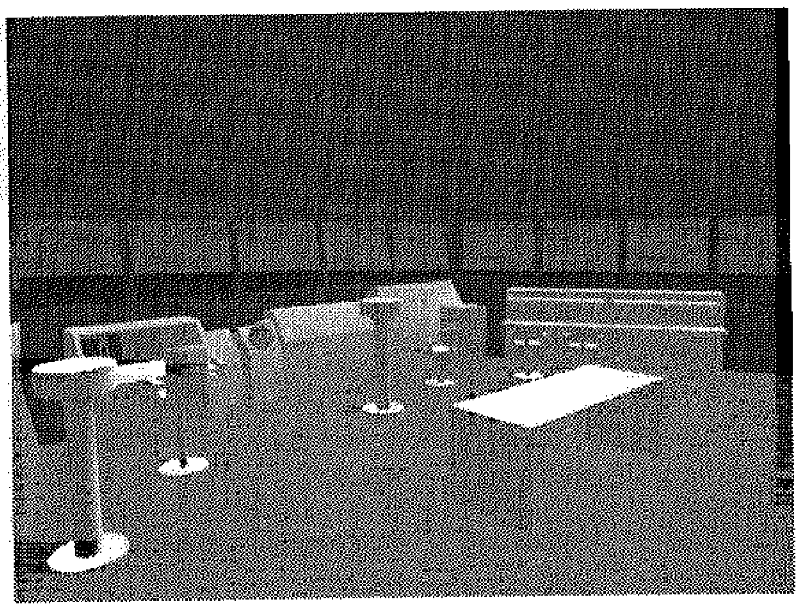
Figure 2 Flight Deck – Operational Equipment



Geometric Modelling

To create a 3-D definition of the layout of a ship design, a product structure tree may be built with selected nodes allocated a geometric reference that points to the appropriate geometry libraries or database to extract the geometry. The product structure also defines the location and orientation of each item of geometry used to give a true representation of the physical design. The geometry consists of the ship structure and steelwork, equipment and components, connecting services and operational items.

Figure 3 Arrangement Layout



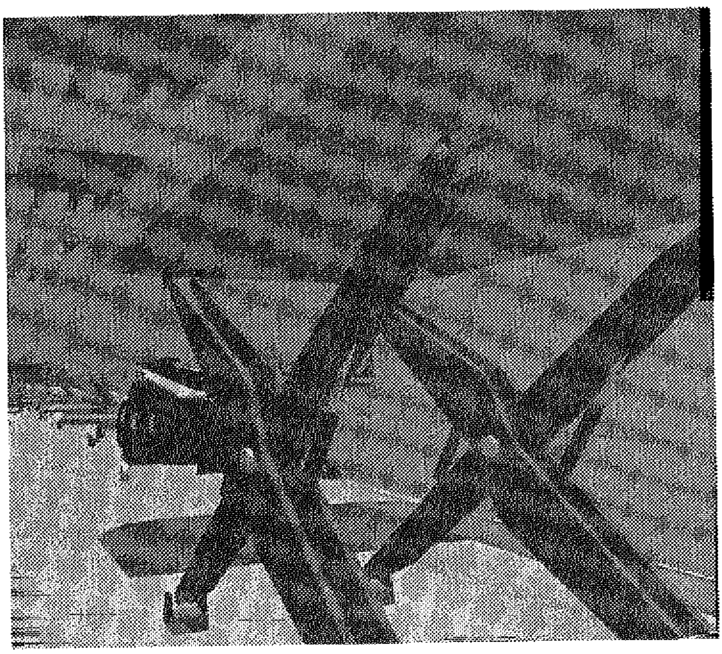
3-D models that are to be used for displaying simulation results must be constructed such that all component part geometry affected by the simulation is independently modelled and allocated a unique identifier. This will ensure that the parts may be selected and manipulated as necessary. The form and content of the component part geometry is dependent on the level of simulation to be undertaken. For simple general arrangement layout it is sufficient to create one-piece geometric components and locate them as required within some boundary as illustrated in the simple layout in figure 3

As a design evolves, several options are considered when trying to achieve stated requirements, the optimum solution will be selected after full analysis of the options against the requirement criteria. Using the product structure tree, each option may be modelled and the appropriate supporting data attributes allocated to tree nodes. The optional solutions can then be reviewed by selecting one or more subsets of the product structure tree and analyzing the results.

Animation

For animated components, the geometry must be further decomposed to model the parts to be animated in the simulation, for example an aircraft lift will be modeled as the platform and the critical component parts of the lift assembly. During simulation the mechanisms will be animated in accordance with the behaviour calculated from the kinematics see figure 4.

Figure 4 Aircraft Lift for Simulation



Design options

When design options are to be considered, a product structure tree is used to define a common origin for the options as a tree node and then define the options as sub-nodes. Each option can then be selected from the list of sub-nodes for display. The design option technique may be used to verify the effects of each of the possible options during design review with simulated assessment of the operational effect of each option analyzed and either the appropriate option selected or several options proven. See figure 5 and figure 6

Figure 5 Small superstructure option

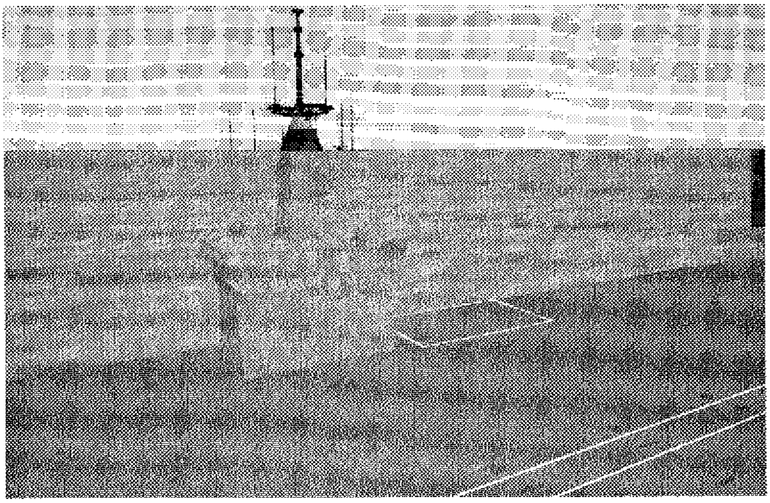
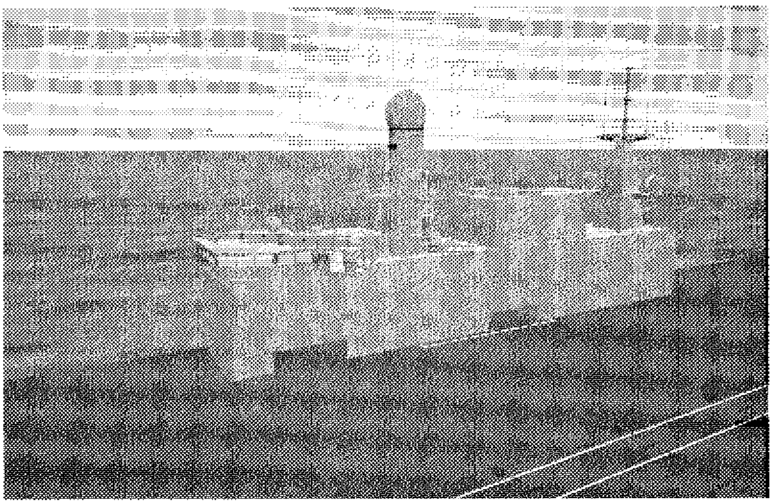


Figure 6 Large superstructure option



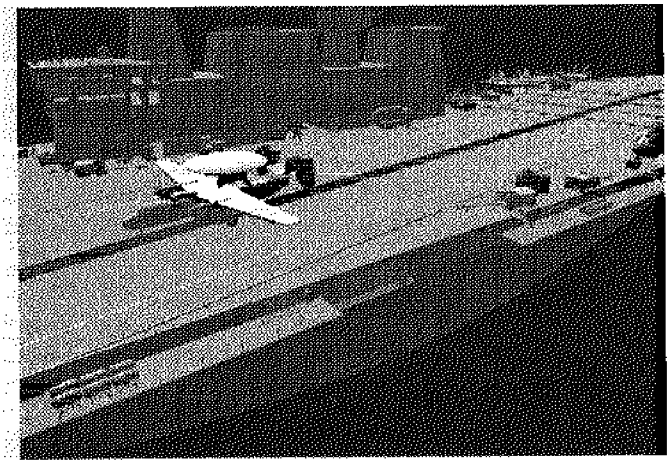
Simulation Considerations - Physical Characteristics

Physical items that form component parts of the ship have associated restraints or constraints, which affect their behaviour in operation. The constraints may be directly associated with partial geometry such as a hinge along one edge of a door or a rotating component on a shaft. They may be associated with movement of an object having one or more degrees of freedom, or may involve a change of form of the object itself.

To simulate the physical behaviour of component parts of an object, the object must be represented by a model constructed such that all geometry affected by the simulation is individually identified and controlled. A product structure tree has the functionality to enables this to be achieved as well as allowing behaviour attributes to be allocated to all parts.

For complex simulations in real time, the component geometry being simulated will have location and orientation provided at specific time intervals from the simulation calculation to convey the impression of movement within the model. To maximize performance, the component geometry in this case will be optimized in level of detail (for example, the simulation of an aircraft on an Aircraft Carrier will use an aircraft geometry which minimizes level of detail whilst providing a reasonable representation of the aircraft see figure 7).

Figure 7 Aircraft on a Aircraft Carrier



To manage the geometry models for components and their individual parts requires a structured database with individual controls for each part and a hierarchical relationship between the parts. This can be achieved using a product structure model as provided in modern EDM/CAD systems. Furthermore, if the simulation involves mixed discipline geometry of large complex data sets such as ships and submarines then well defined database technologies and concurrent engineering philosophies are desirable to guarantee the data capture and access required.

The Product Model

Concurrent Engineering

The implementation of concurrent engineering is well documented as being a complex task involving a combination of people, processes and technology issues. The concept and ideas that enable concurrent multi-discipline working demand the flexibility and functionality to decompose component parts of a design into a logical hierarchy for controlled data management, access and sharing. This gives the option of attaching attribute data and conditional status to individual, logically grouped or physically grouped object representations.

The ability to allocate non-geometric attribute data to the nodes of a product structure tree and control the access and extraction of selective information from the hierarchical structure during design development allows object behaviours to be recorded for each component part of the design.

In the concurrent engineering environment customer and suppliers contribute to the operational aspects of the design requirements for the vessel and its component parts using their knowledge and experience to specify the expected behaviours of systems, individual equipment and the ship itself under stated operational conditions. This information can be managed through a product Model.

Single Source Data

The requirement for single source data is critical to successful implementation of concurrent engineering. All data users to ensure consistency, accuracy and effectiveness share the single source. The data capture, management and control must be arranged such that individual or selective groups of data are owned by a pre-determined creators whom have full write and edit access and accountability for the quality and content. The data will then be extracted and used by project team members with read only access authority.

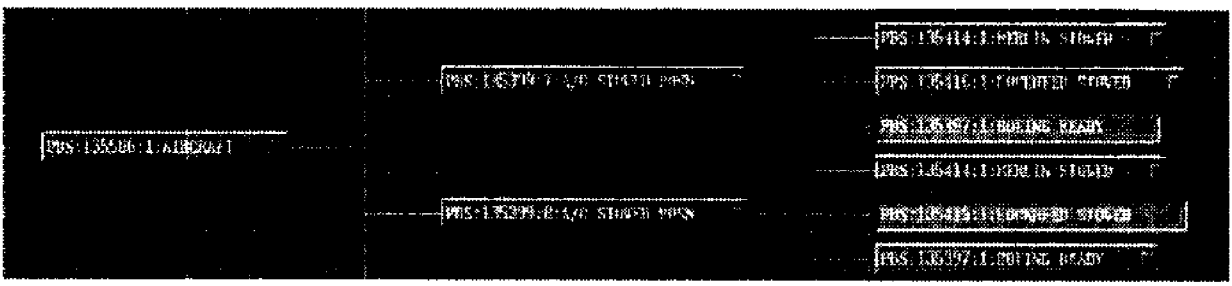
In practice the management and control of a single source of product data may be achieved using the product structure concept. The product structure also allows geometric data and associated behaviour attributes to be stored and managed in a logical manner making it a good tool for dissecting geometry into component parts, ideal for simulation.

Product Structure Tree

The product structure tree manages the configuration of the model, its operational characteristics and design options and specifies the relationship, location, orientation and function of all items on the ship. Optional design details may then selected for review, comparison and manipulation or an advanced visualisation (VR) display technologies.

Data review, approval and release of all information that defines a product can be achieved through the product structure tree with the status of individual or grouped subsets of the data recorded as attributes on the tree nodes. Using a product structure tree, the design configuration can be maintained from concept design, through production and in-service life of the product see figure 8

Figure 8 Extract from Product Structure Tree



Attribute Data

The definition of a ship involves a very high volume of data in a wide range of formats and media types, all of which contribute to the final configuration of the product design. The data will include requirements, specifications, design calculation, systems definitions, equipment lists, cost estimates, applied standards, build criteria and the geometric layout of the vessel. In concept design, feasibility studies and design class evolution, the definition may have several optional design solutions from which the optimum is selected for a particular product. The product structure tree can manage the configuration and all of the attribute data associated with the design giving a single well-controlled specification of the product, its component parts and their status at any time during the design cycle.

A product model based on a hierarchical tree structure offers the functionality to capture, and manage design attributes and options as component parts, as required for input to simulation tools.

Project VITESSE – A Practical Example

Overview

The Ministry of Defence (UK) department of Naval Architecture and Future Projects have a vision to adopt Simulation Based Design in the procurement process for future warships. The process will merge the simulation results for specific detail of concept designs to verify the operational effectiveness of the options studied. To achieve the vision, the project, named VITESSE chose the CVF UK future aircraft carrier concept design options to illustrate the application, accuracy, and potential use of simulation and interface technologies to prove the operational effectiveness of the ship.

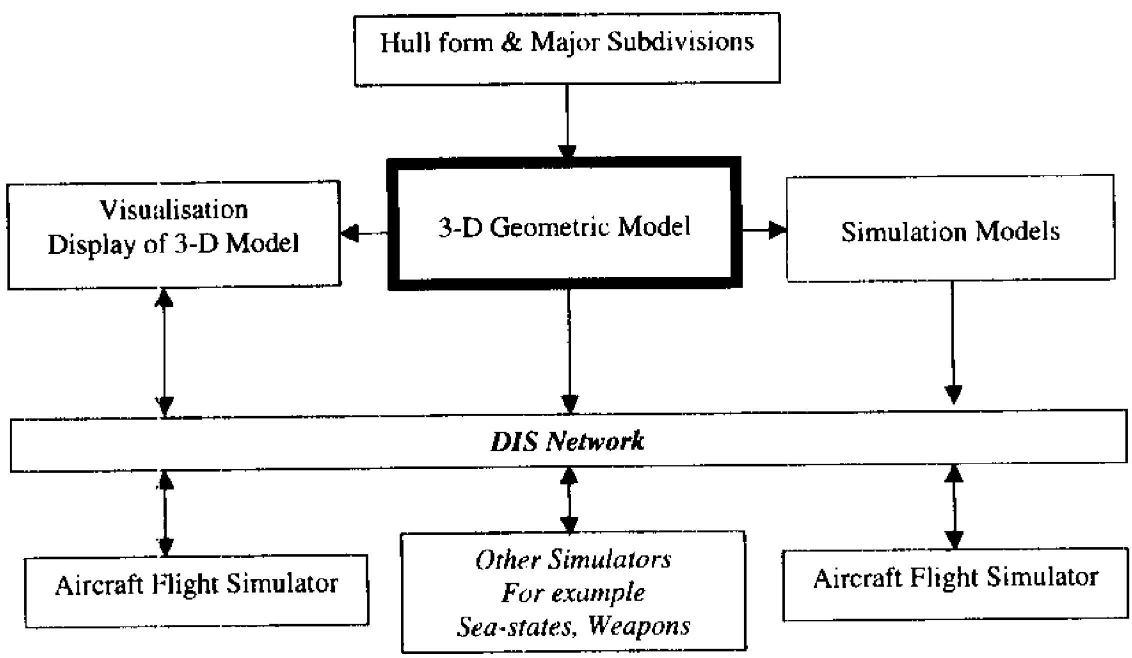
The objective of the project was to create an environment using 3-D geometric modelling with the capability to display concurrently, and/or in real time, the results of several simulation studies involving aircraft sortie rates. Simulators used for flying the aircraft, the behaviour of the ship in various sea states, and human factors in preparing aircraft for flight were remotely located and merged over a distributed network using DIS technology with visual display of the simulations at each site.

Project Architecture

To achieve the objective of the project, several industries were invited to contribute their skills and expertise to the project. The principle tasks were to develop the ship hull form and major subdivisions, create a 3-D CAD model of the ship. To calculate the physical behaviors of the ship and its components, link the ship model and its simulated behaviors to aircraft flight simulators and present a combined display showing the aircraft landing on the ship, refueling and weapons loading and take off operations.

Contributions from industry involved companies working closely to develop data models then convert them and use them in a range of simulator tools, the resultant architecture as shown in fig 9

Figure 9 Project Architecture



Data Conversion

Geometric representations of ship designs are created using computer aided design (CAD) systems. Modern CAD systems capable of handling large complex data volumes use hierarchical structures to manage the design geometry. Assembly Modelling and/or Product Structure Trees are used to manage the relationship between the component parts of the design.

When converting the design from the source CAD system to a visualisation/simulation environment, the assembly/product structure relationships are critical in defining the design and must be converted accurately along with the geometry files for the components. The hierarchical structure of the assembly or product structure may contain the location and orientation data essential to create the assembled model, failing to convert this data results in all component parts of the geometry being located at the model origin.

A conversion process from an assembly based CAD system involves two steps, first the assembly model or product structure tree is converted to the target system, then all of the component parts used in the assembly are identified, extracted from the CAD library and converted.

In the VITESSE project, three visualisation technologies were evaluated and the master CAD data converted for input to each. One converter, provided by a system vendor and used by Marconi Marine for design review visualisation, considered the product model structure and delivered a usable converted model for visualisation/simulation with minimum effort and time.

Two other converters supplied from specialist third party companies did not consider the product structure with the result that all part geometry was converted to the target system format and located at the model origin. Manual correction was required to redefine the location and orientation of each geometry item on the ship in accordance with the product structure data.

Scope of Modelling Tasks

Marconi Marine (VSEL) was allocated the task of generating the 3-D geometry, which would be used to evaluate several simulation and visualisation systems during the VITESSE project. The scope of the work was to deliver models for three concept ship designs that could be used to study operational aspects of the aircraft and their sortie rates. To do this, the models addressed exclusively the flight deck, aircraft hangar and magazines. The task involved input of a hull form and creating product structures and geometry models for areas to be studied.

The main input to the process was an IGES format wire frame model of hull forms. 3-D solids/surface models were then created using Marconi Marine (VSEL) Concurrent Engineering processes based on 'Parametric Technology' Optegra/CADDS5 systems with visualisation using 'Division Ltd.' dVise VR products. Output to third party companies for simulation studies involved data formats including CADDS5 (with product Structure tree), dVise, Open Flight and IGES files.

The Flight Deck details with major markings, superstructure, operational items for aircraft launch and recovery, aircraft lifts, maneuvering and other service vehicles, and other operational equipment were modelled in the CAD system with a component breakdown to suit the simulation behaviour criteria.

The aircraft hangar and magazines were detailed in the CAD system. Magazine weapons stowage was represented by simple block models of groups of weapons cases with one case of each type modelled as box, lid and weapon to enable weapons handling simulation and animation.

In order to effectively visualise the model using standard display technologies, the geometry representing moving components such as doors, blast screens, and lifts were modelled in open and closed condition, selected via the product structure tree, with one or the other switched on for viewing at any time. This enabled the CAD data to be displayed in optional positions/conditions without the need for simulation software or advanced visual tools.

Spatial Layout

Spatial layouts of ship arrangements are have been created on CAD systems for many years and the techniques and processes to achieve an acceptable arrangement are well proven. However, at the concept design stage several options need to be considered and their effects taken into account on the overall requirements for the ship. To address the issue of a mix of aircraft types on the ship, and the requirement that each will need to be parked, moved, stored, maintained and serviced, the model needs the flexibility to show each of the potential aircraft at all possible operating spots.

This is achieved in the EDM/CAD system by using the product tree to identify the location of the aircraft with a reference node and define all possible aircraft as tree nodes that are created as 'children' of the reference node. Any one of the aircraft options can then be switched on for display, and the spatial requirement for all possible aircraft can be considered. Using this principle figure 10 and figure 11 show the effect of selective aircraft display showing two of the many options.

Figure 10 Selective Aircraft Display I

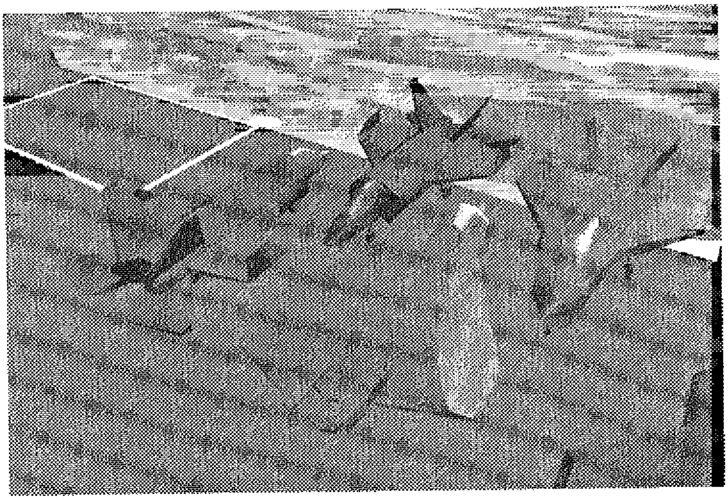
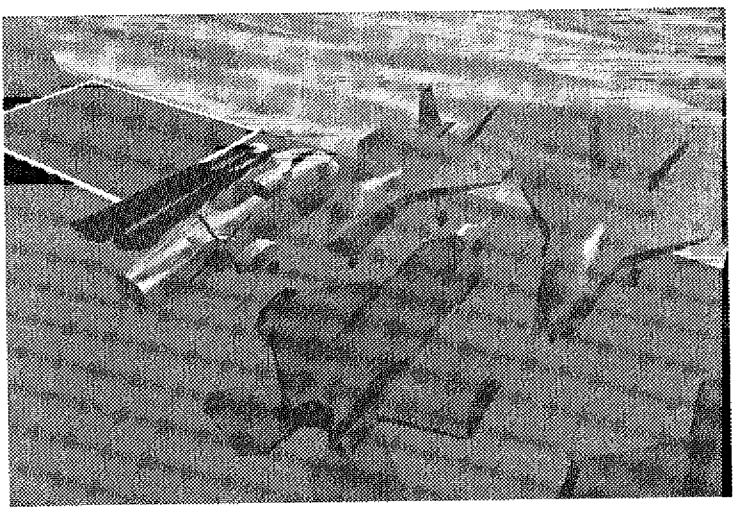


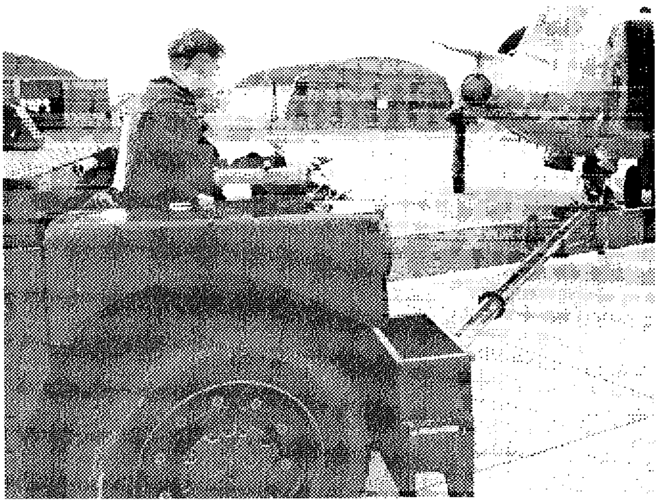
Figure 11 Selective Aircraft Display II



Kinematic Simulation

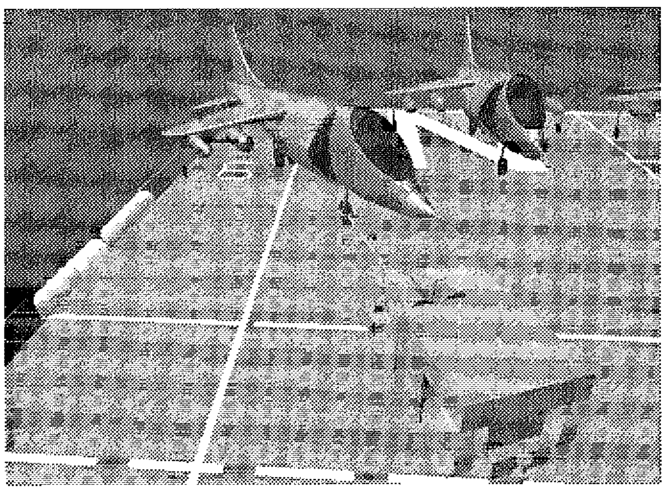
The operational process for aircraft movements on the flight deck of an aircraft carrier requires complex kinematic calculations involving a four wheel steer truck as shown in figure 12, a linkage bar with hinged connections at both ends, an aircraft turning wheel, and the turning behaviour of the aircraft itself.

Figure 12 Tow-truck Photograph (showing linkages)



To visualise the results of a simulation of the above process, behaviour attributes must be associated with the geometric model components affected by the calculation. For example, the tow-truck wheels, the tow-bar linkages at each end, the aircraft wheel, and the truck and aircraft bodies will each have turning and maneuvering characteristics. The 3-D model of the truck is thus created to ensure that moving components can be manipulated independently of the main body geometry. See figure 13

Figure 13 CAD Tow-truck with linkages



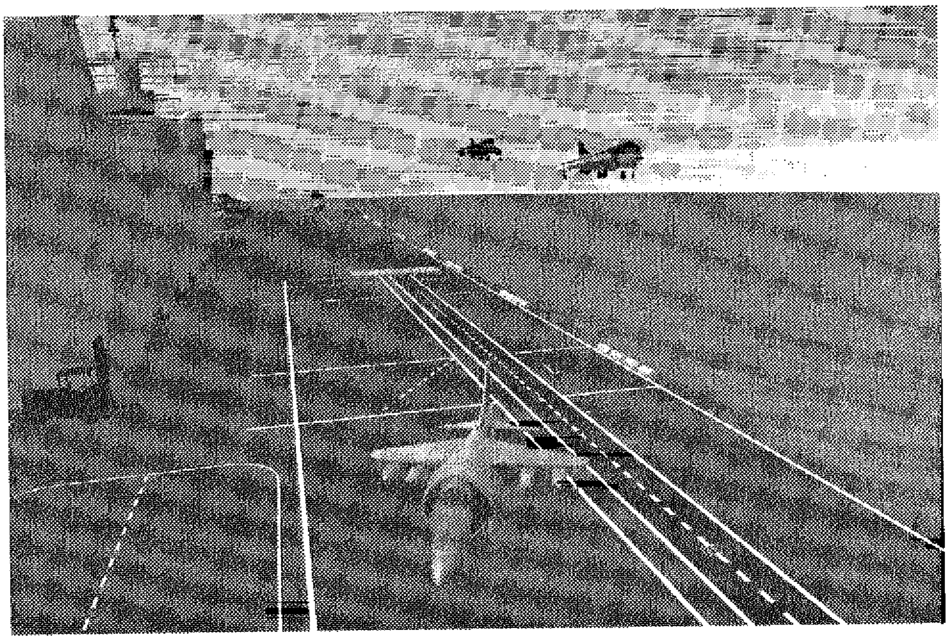
Project Results

The VITESSE project completed successfully in March 1999 with a demonstration to senior MOD staff of aircraft carrier models for two concept ships and the in-service ship HMS Invincible. The demonstration showed the ships in various sea states, the maneuvering of aircraft to service points, aircraft fueling and weapons preparation. Qualified pilots in a remote flight simulator carried out landing and take-off operations on the carriers. The ship geometry being displayed in the flight simulator and the position of the aircraft displayed in the ship simulator, controlled by data messages sent along a network using DIS (distributed interactive simulation) technology.

The objectives of building CAD models as 'master data' for a simulation exercise were achieved, with the functionality of the product structure tree playing a major role in the flexibility of the geometric data. This gave the ability to model and manages several options for selectively review and consideration from very early stages of ship design. The CAD data was converted to a variety of visualisation/simulation technologies with variable levels of success

Simulation of the kinematics of the tow-trucks and aircraft was also achieved, the simulation engineer suggesting that models be decomposed into appropriate component parts prior to simulating. The real time exercises involving flight simulators were achieved with some difficulties in response caused by the complexity and size of the CAD models. Although the models were created with considerations for minimizing levels of detail, they were compromised to give reasonable visual representations.

Figure 14 Aircraft Approaching for Vertical Landing



The Future

The success of the VITESSE project in developing models, simulation applications and a distributed simulation network demonstrated the technology as feasible. VITESSE will be followed by an in depth study over several years on application and use of the technology as an aid to warship procurement.

Conclusions

General

Creation of the CAD 3-D models for the VITESSE project involved some consideration of the requirements of simulators to define component parts of assemblies and allocate behaviour to them as appropriate. When creating EDM/CAD models for the layout of ship compartments the principle consideration is spatial arrangement of equipment and services, which requires representative geometry rather than high definition manufacturing detail. Use of a product structure tree representation of a ship is also acceptable to shipbuilders who often work with structured work breakdown definitions such as weight groups.

The only aspect of concern when preparing CAD geometry for simulation is to identify the component parts to be considered during the simulation and break down the product structure to a level to suit this requirement. The process is thus a minor deviation from normal shipbuilding practice, the costs in resource being to add additional nodes on the product structure tree and decompose the CAD model into component parts (we used to do this often with CAD layering technology!).

In terms of real-time simulation, such as the flight simulators feeding positional information through DIS, the major consideration is for the animated geometry (in this case the aircraft) to be modelled with the lowest level of detail that gives acceptable visualisation. This will maximize visualisation performance.

Specific

The use of simulation based design tools and technologies is evolving as a stable environment which may be applied with minimum effort over and above the current work of preparing and executing independent simulation studies on critical aspects of the design.

The development of EDM/CAD systems with structured data management and attributes associated with geometry offers the controlling mechanism for defining databases which can be used to detail component parts of assemblies with relative ease.

The high performance visualisation systems offered by CAD and Virtual Reality vendors can display large data models and have the capability and performance for real-time visual display.

Combining the above statements suggests that simulation based design is achievable, the technologies involved have been proven and are further developing (DIS network is being replaced with more powerful HLA technology for example). In addition the process required to prepare the data does not deviate from normal ship design practice other than to add more detail to selected components.

Concept Design

The VITESSE project vision is to develop a simulation based design environment suitable for use in ship procurement. This implies that the simulation based design activities will start at the very early concept stages of the design process. The technology is available to achieve this, as illustrated by the VITESSE project. Naval Architects and Ship Designers, however, need to adopt the use of product modelling and CAD 3-D arrangements to replace traditional 2-D drawings in order to gain advantage from the process and technologies.

Developing concept designs using 3-D arrangements in a modern product structured EDM/CAD environment has the advantage that geometry created will be stored in the database and may be selected for use in future designs. Adding appropriate generic attributes to this geometry will develop a library of re-usable 'blocks' which may be selected and used in future design considerations and adopted in a block based design philosophy to create quick, accurate and flexible design options at concept stage. Verifying this with simulation results gives a high confidence of the expected performance of the design options studied and a means of selecting the optimum design. If the 'block' attributes include cost models and build considerations, selecting an optimum design will not only consider design expected performance against requirements, it will also consider costs against performance.

Summary

Simulation Based Design is an evolving technology that may be used from the early stages of concept design to select from several potential options

The underlying EDM/CAD processes do not deviate from normal shipbuilding practice and may be brought forward to the concept design stage. If the selected concept design becomes a contract, then the product structure tree can be extended through the design build cycle.

The VITESSE project has shown the technology to be feasible.

USING PROCESS MODELS AND INTELLIGENT AGENTS TO SUPPORT COLLABORATIVE ENGINEERING IN SHIPBUILDING

Masahiro Sonda, Ship & Ocean Foundation, Tokyo, Japan
Yutaka Takemoto , Ship & Ocean Foundation, Tokyo, Japan
Hiroyuki Okabe, Ship & Ocean Foundation, Tokyo, Japan
Yan Jin, University of Southern California, Los Angeles, USA

Abstract

In this paper we discuss collaboration support for engineering activities in shipbuilding. Ship design is a complex process and involves multiple engineers to work concurrently on different parts of an overall design. The demand of short lead time has created a highly concurrent engineering practice in the real world.

We take a process-based approach to achieve coordinate engineering activities. In this approach, engineering activities are prescribed by a set of pre-definable and dynamically updateable active process models. An agent-based approach has been taken to implement the active process models. Each agent is associated with an engineer. It helps the engineer to get information from process servers, and report status of, and the exceptions generated from, the engineer to the process servers.

We will present active process models and the framework of process-based collaboration support. After that, we will show a collaboration support case example that uses the proposed framework for collaborative engineering support to prove effectiveness of our approach.

This research is one of the main works in on-going ACIM(Advanced CIM) project, which is funded by Ship and Ocean Foundation with participating 7 major ship-building companies in Japan[1, 2].

Introduction

Large scale engineering problems, such as ship design, require multiple engineers to work on different parts of an overall problem. At any time, different engineers may work on interrelated tasks—i.e., the results or decisions of one designer may have significant impacts on those of other designers. To prevent inconsistency in results and reduce redundant activities, engineers must collaborate effectively so that information flows correctly and timely and engineering activities are well coordinated.

Sharing engineering information through a common product model has been proven an effective way for engineering team members to identify inconsistencies and generate timely information flows. By accessing a logically centralized product model database, an engineer can retrieve the latest engineering information generated by other engineers and pass his or her task results to others through the database. While shared product model can be used to facilitate information flows among engineers, it contains only the results generated by engineers. The information of the process through which the results were generated is not part of a product model. Furthermore, the control information that defines how engineering activities should be carried out and how they related to each other is also missing from most product models.

In this paper, we propose a framework based on a process-driven and agent-supported approach to collaborative engineering support called 'Active Process.' Active Process has a capability to describe 'how engineering tasks should be done' in order of precedence. It is powerful but still not sufficient because engineering task procedures will be often changed according to situation. To meet this problem, agent-supported approach is introduced to Active Process[3, 4]. Each participants in the system has its own agent who is interested in negotiation with others to acquire information. Agent has intelligence to modify pre-defined process based on negotiation and can offer participants useful information to ask them for instruction if the situation is too difficult to judge.

The basic ideas behind this approach are:

1. Engineering activities can be prescribed by a set of pre-definable and dynamically updateable process models;
2. These process models can be used as points of reference as well as points of control for engineers to coordinate their engineering activities; and
3. Intelligent agents can be developed to support the "referencing" and "controlling" based on the predefined process models.

The overall goal of our research is to develop a process-driven and agent-supported framework collaborative engineering support and test framework through prototyping and applying it to example scenarios. The specific goals of the research include:

1. To clarify requirements of process-driven collaborative engineering support.
2. To identify process elements and develop a process model for engineering support
3. To define a process-driven work paradigm based a set of engineering task scenarios.
4. To develop intelligent agents and an agent-network for monitoring and controlling the engineering processes based on the process models.
5. To test the concepts and the framework by developing prototype systems and testing scenarios.

To achieve our research goals described above, we are taking a modeling-prototyping-case study approach.

Process-Driven Work Paradigm

To support engineering collaboration, we propose a process-driven work paradigm. There are two basic research issues that must be addressed in order to realize the proposed work paradigm. One is how to model the process information, and the other is how to apply and manipulate the process information for engineering work and collaboration support.

The Need of Process Model

While knowledge about engineering products captured by product models is important for engineering support and needs to be shared among distributed engineers and systems, another important aspect of engineering knowledge is about engineering processes. Engineering tasks are not carried out in a random way. Rather, they are well planned and managed based on the planned processes. An engineering process is composed of a set of interrelated activities that collectively realize certain engineering objectives. A process model is a representation of a engineering process in a form which supports automated manipulation or enactment by a process management system. The process definition consists of a network of activities and their relationships, criteria to indicate the start and termination of the process, and information about the individual activities such as participants associated IT applications and data, etc.

From an engineering point of view, process models are needed to support engineering process design and planning, process knowledge sharing, and engineering collaboration. The goal of engineering process design is to generate a document that prescribe a set of tasks and the ways in which the tasks should be carried out by whom using what tools. A well defined process model will help process designers and managers identify key activities, required participants (resources) and define their relationships. Since process models can be made machine readable, it makes it easy to use process management systems to manage process execution. Furthermore, the process information in a process model can help coordination among engineers. Engineers can share the process model so that they are aware of the whole engineering process, in terms of who is responsible for which activity, who is busy, who is advanced and who is behind. This information about current situation can help engineers determine whom to ask for information, to whom to route a certain task, and who needs help.

Process-Driven Engineering Task Management

As described above, a process model is a representation of a business or engineering process in a form that supports automated manipulation, such as modeling, or enactment by some process management systems or programs. Figure 1 illustrates how process information represented as process models can be used to support engineering work and collaboration.

Process design: Engineering process design starts from putting together process ideas (e.g., relevant processes, activities) about a specific engineering task. The ideas may come from experience or from a process repository that contains all known process ideas about an engineering domain. As shown in Figure 1, the process ideas are applied to define a specific process model.

Execution, Work and Collaborate: Once a process model is determined for a given task, then the model is used to guide the execution of the process by human engineers, project managers as well as computer controlled systems. Based on the task specification described in the process model, each engineer carries out his or her task and coordinates with others when needed. Interdependencies predefined and dynamically generated are explicitly represented in the process model for collaboration support.

Monitor, Report, Evaluate: The execution process information can be monitored by certain systems, e.g., intelligent agents, or reported directly by engineers. The execution information can be used to evaluate how well the engineering team is following the process plan. For example, by comparing the progress information with the original process model, it will become clear whether the project team is ahead of the schedule or behind.

Control and modify: One output of the evaluation is the instructions or control of the project team to improve their performance. If the evaluation result shows that the process model does not fit the current project team well, then the other output of "evaluation" will be to modify the original process model to make it more achievable by the project team. This loop, middle loop in Figure 1, of process model application is quite similar to a feedback control system in which the process model is used as a reference or input to the system and the monitored information is used as the feedback for generating control instructions.

Improve and Store: Another way to use the monitored process information is for "long term" process knowledge improvement through learning, as shown in Figure 1. The accumulation of the monitored process execution information can provide important insights on how effective and efficient the original process model is. By analyzing the information through benchmarking the executions of different process models, one can acquire new process knowledge and improve understanding of the known process models. The new knowledge can then be stored into the process repository.

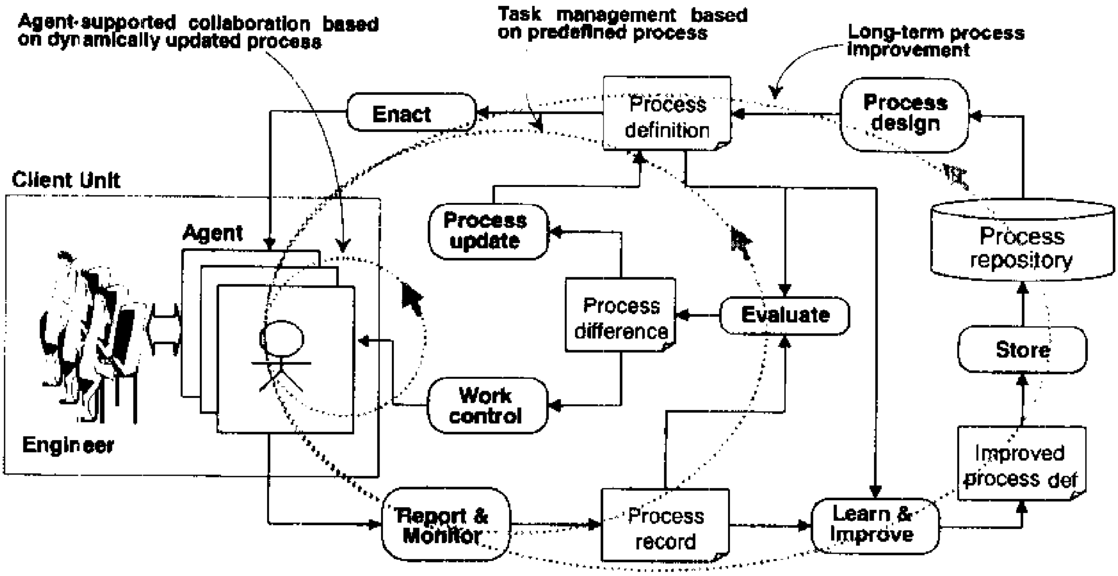


Figure 1. Process Model and its Application for Engineering Support

Those 2 feedback loops described above are based on process model, which are defined in advance. But it is very difficult to pre-define all the processes because of the complexity in shipbuilding. So it is preferable to get dependencies in processes dynamically during operation. Using intelligent agents, we introduce this function which is represented as the inmost feedback loop in Figure 1.

Agent-Based Work Support

To make the task management framework a real engineering work support system, the issues of systems design and implementation must be addressed.

Figure 2 illustrates the conceptual system image of Active Process System(APS).

We have identified three important system components that will be explored through this research. The components are: Intelligent Agents, Process Server and Enterprise Resource Server.

Intelligent-Agents: Most important function that intelligent-agents will do is to extract dependencies in processes from monitored results dynamically. It can compensate uncertainty or complexity of shipbuilding engineering processes. Intelligent Agent can facilitate appropriate local operations service for its engineer as well. This may include local application management, operation bookkeeping and communication management. And the agent can function as a systems interface between Process Server and human engineers. This function makes it possible to make the Process

Server more general and leave engineer specifics to Intelligent Agents to manage.

Process Servers: The functions of Process Servers include capturing process models and managing process execution based on the process models. Process models must represent "planned (or intended)" processes, "simulated" processes, and "executed (or real)" processes. For process execution management, Process Servers must be able to generate activation signals, control the real process by reasoning based on the given rules and the data of "planned", "simulated" and "real" processes. Process Servers may be organized in a hierarchical way so that the higher level servers can take care of more "global" issues of the process while the lower level ones can manage activities at more engineering or task specific levels.

Enterprise Resource Server: To manage engineering processes, resource information is very important. Enterprise resource server centralizes resource information logically. It has also process templates and journal information. Process templates are typical pattern of process combination. Those will be enhanced through the outmost feedback loop in Figure 1. Journal information means the results of engineer's operation or communication. Dynamic dependencies may be extracted from journal information.

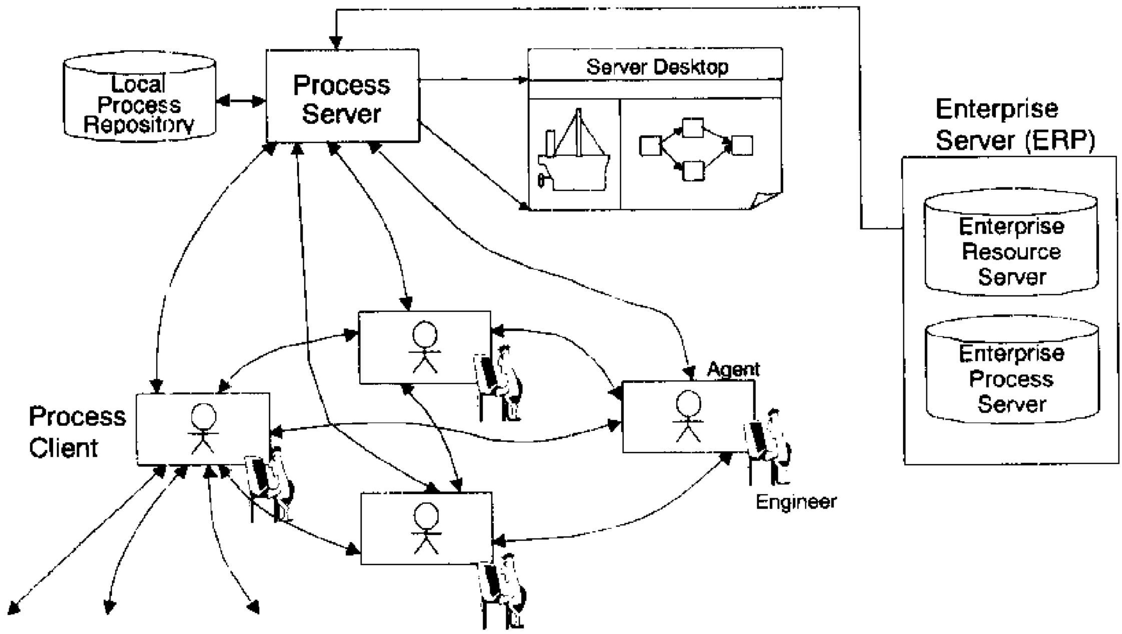


Figure 2. Active Process System

Process Modeling

The purpose of process modeling is to develop an ontology, i.e., a set of concepts and their interrelationships, that can be used to capture process knowledge and information and to manipulate (i.e., design, monitor, control) the process information. In this section we show the process description language in Active Process.

Active Process is a dynamic work process model. Unlike existing process or workflow models, Active Process is intended to be able to meet not only pre-definable process, i.e. activities(defined later) and dependencies among them, but also unforeseen, finer work unit dependencies. Our model has description capability in compliance with granularity of tasks.

One feature of Active Process is that processes will be linked to product model data. Product model represents component data which comprise a product and their inter-dependencies. Execution of process results in the evolution of product model data.

Active Process Model

Following components are addressed in the Active Process model.

1. Process: A process is composed out of a set of activities.
2. Activity: An activity is a unit of behavior of a process and used to describe a process at most abstract level for process design. Detailed dependency relationships between the activities should be investigated for collaboration support.
3. Resource: A resource is an object that is used or produced by one or multiple activities. Product is a specific resource.
4. Dependency: A dependency relation defines relationship between activities, or an activity and a resource, or resources.

What is Process?

The definitions of process can be given according to the following two view points. As a description model view point, a process is defined as a network of interrelated activities. As a control mechanism view point, a process is defined as a set of continuous or discrete behaviors that can be observed and controlled. As general features of a process are as follows. For description and documentation, a process expresses concepts, relations, requirements and properties. For enactment, a process has the information related to the behavior, performance, environment and feedback (learning).

General Characterization of Process

The process is generally characterized as follows;

$$\mathbf{P}(t) = \{\mathbf{A}, \mathbf{U}, \mathbf{D}, \mathbf{R}, \mathbf{Q}; t\}$$

Where,

A: activity,	$\mathbf{A} = \{a_1, a_2, \ldots a_J\}$
U: resource,	$\mathbf{U} = \{u_1, u_2, \ldots u_K\}$
D: product,	$\mathbf{D} = \{d_1, d_2, \ldots d_L\}$
R: relation,	$\mathbf{R} = \{\mathbf{R}^a, \mathbf{R}^u, \mathbf{R}^d, \mathbf{R}^{au}, \mathbf{R}^{ad}, \mathbf{R}^{ud}\}$
Q: requirement,	$\mathbf{Q} = \{q_1, q_2, \cdots q_N\}$

$\mathbf{R}^a$: activity relation,	$\mathbf{R}^a = \{r^a_1, r^a_2, \ldots, r^a_F\}$
$\mathbf{R}^u$: resource relation,	$\mathbf{R}^u = \{r^u_1, r^u_2, \ldots, r^u_G\}$
$\mathbf{R}^d$: product relation,	$\mathbf{R}^d = \{r^d_1, r^d_2, \ldots, r^d_H\}$

$\mathbf{R}^{au}$: resource assignment relation

$\mathbf{R}^{ad}$: product assignment relation

$\mathbf{R}^{ud}$: relation between resource and product

Implications of Process

We have to notice the implications which process has. $P(t = t_0)$ is a "complete" description of the process state at $t = t_0$ can be described as a static process description. The analytical properties of a process can be addressed. From a P(t)'s point of view, the unit of analysis is Activity. On the other hand, process is a continuous behavior changing from $P(t = t_0)$ to P(t). The above mentioned characters of process, ***A, U, D, R, Q***, all evolve and/or change over time. This means that addressing P(t) requires behavior models of ***A,U,D,R,Q***. Also, this need to move unit of analysis from Activity to components of Activity.

Another important implication of process concerns about **R**, relations. The complexity of a process **P** is largely determined by $\boldsymbol{R} = \{\boldsymbol{R}^a, \boldsymbol{R}^u, \boldsymbol{R}^d, \boldsymbol{R}^{au}, \boldsymbol{R}^{ad}, \boldsymbol{R}^{ad}\}$. Also, the uncertainty of process comes from **R**, relations. Practically, a complete information of $\boldsymbol{R} = \{\boldsymbol{R}^a, \boldsymbol{R}^u, \boldsymbol{R}^d, \boldsymbol{R}^{au}, \boldsymbol{R}^{ud}, \boldsymbol{R}^{ad}\}$ is not available both in $t = t_0$ and t. To acquire more information about ***R*** requires interaction and also costs time and money. Furthermore, ***R*** changes over time. This means that better or lower level model of ***R*** is needed.

Activity

Activity is a unit of a behavior of a process. The definition of an activity is as follows:

$a = \{o, d, u, q; B\}$, where o = action, B = Behavior

The function of an activity can be described as follows:

Function = $\{o, d\}$, e.g., {*design, hull*}

Action o is resource relevant concept.

Behavior B means manipulation and completion of activity components. Behavior B can be described by following method:

complete behavior: differential equation based

computational behavior: discrete event simulation

Proximate human behavior

Those are the formal definition of Activity. This model is intended to be applied to ship-building tasks. From the viewpoint of application of the model to tasks in ship-building, we assume Activity to be corresponding with the manageable granularity of tasks.

Moreover, we introduce *WorkElement, Event, Exception* and *Performance* in our modeling concepts.

Work Element

The concept of Work Element has been introduced to express the running of Activity or the behavior of Activity. An action o is performed by an actor/designer on a set of components d^e_i $(i = 1, \ldots, S;\ d^e_i \subset d)$ of product d.

A work element $w^e_i = \{o, d^e_i\}$ is defined as a component of an activity a, i.e., $w^e_i \subset a$, $i = 1, \ldots, S$.

The reason why a work Element w^e is not an Activity are as follows;

WE is too simple: All $w^e_i = \{o, d^e_i\}$ are within individual's reach

WE is too complex: Because ***R*** cannot be clearly defined so that teamwork is needed. If all **R** are clearly pre-defined, each designer can proceed his/her task according to the predefined instruction.

The status of Work Element is basically described as one of two conditions, i.e. pre-start and completed. The requirement or possibility to describe intermediate status of Work Element depends on domain & problem. Work element relations composed of intra-relations in a Activity and inter-relations between Activities. That is, a Work Element may have a relation with other one belonging to the same activity, or other activity. Work Element relations are essential model to capture underlying mechanisms of activity relations. The relation is dynamic. It is generated when recognized/identified. And it becomes strong / weak when manifested in a process.

Work element is defined as a unit of work that can be measured or evaluated based on the unit progress of product information(d^e_i, in the definition of w^e_i) being generated from engineering design. Work element is introduced to link product model with process model for work and collaboration support.

The connection of process model and product model is a significant feature of ActiveProcess. Product model has a complete data set for a product while it has not information of process. That is, product model has information on 'what a product is' but has not information on 'how to make it'. Process model offers that information. On the other hand, the progress of process can be represented based on the object state of product model data. Those two models are complementary to each other.

How Do Relations Evolve?

The relations are identified by working together at work element level. As the process proceeds, new relations are detected through the fact that two designers have contact each other. If the relation once identified are manifested again, the relation is strengthened. Figure 3 illustrates this situation schematically.

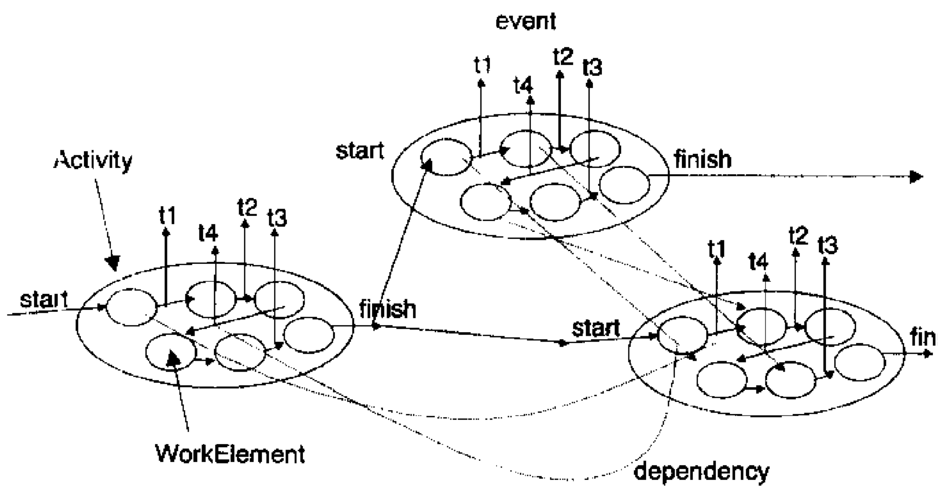


Figure 3. Relation between Work Elements

Events

The concept of event is used for process monitoring. An event e is a notable and noteworthy happening in an activity a and associated with one or more work elements w^e_i. An event compound e^c is a recognizable set of events that can approximate an recognizable action o.

$$e^c = \{e_1, e_2, \ldots, e_Z\};\ o \sim e^c$$

Knowledge is needed to generate e^c and o based on observable events.

Events $\boldsymbol{E}$ represent observable behavior of activities. The underlying mechanism of the $\boldsymbol{E}$ is not assumed, i.e. events can be outputs of a simulation program or can be human actions.

Interpretation of events means to infer "what needs to be done" from e. 1-to-1 mapping based interpretation is useful for fully. On the other hand, the reasoning based interpretation of events may involve uncertainty as shown in Figure 4.

Exceptions

Exceptions are unexpected situations when executing a plan. In Active Process, exceptions are departures from a planned/ideal project process. In Active Process, agent detects/anticipates the exceptions from exception category. While exceptions can occur for many reasons, it is expected that they have a relatively small number of kinds of categories that include. Example of categories may be as follows;

Inquiry:
- Ask for status of work elements
- Prediction
- Communication failure

Change:
- Design status change, change of design process
- Requirement q change

Wrong doing:
- Break commitment

Conflict:
- Critical and non-critical conflicts

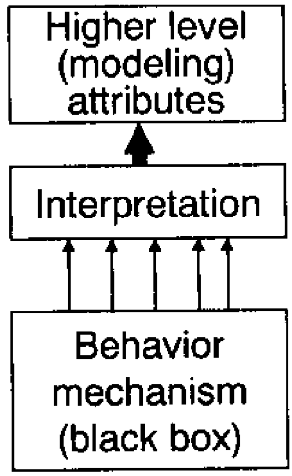


Figure 4. Interpretation of events

Active Process System Architecture

Active Process has two aspects: process modeling capability and management capability for modeled process. As described above, it is very difficult to define all the process precisely in advance. To overcome this difficulty, it is indispensable to collect or modify process dynamically during operation. Therefore we adopted agent-based approach as Active Process system architecture.

To collect dependencies dynamically, our system is modeled as collection of intelligent agents each responsible for one activity or task. Since each agent can work autonomously on behalf of human designer, they are expected to identify both problems and opportunities with the current design process. The initial definition (or design) of the design process will not be followed as a static template. Instead, it will be actively revised and updated by agents who are interested with each other to acquire information and to negotiate revision. We implemented prototype system to examine those ideas.

In this section, we show Active Process system architecture based on agent approach.

Active Process System

Figure5 illustrates the prototype system architecture. Intelligent agent is assigned to each component. Designer + Intelligent Agent is 1 client unit. Communication is performed through agents. All the information which is necessary for collaboration among engineers(i.e., dependencies with activities other engineer does, progress state of other activities, state transition of product model data, etc.) will be captured by an agent instead of an engineer himself. Those information is stored in

Enterprise Resource Server(explained later) as engineering task journal.

Components of the system are detailed below.

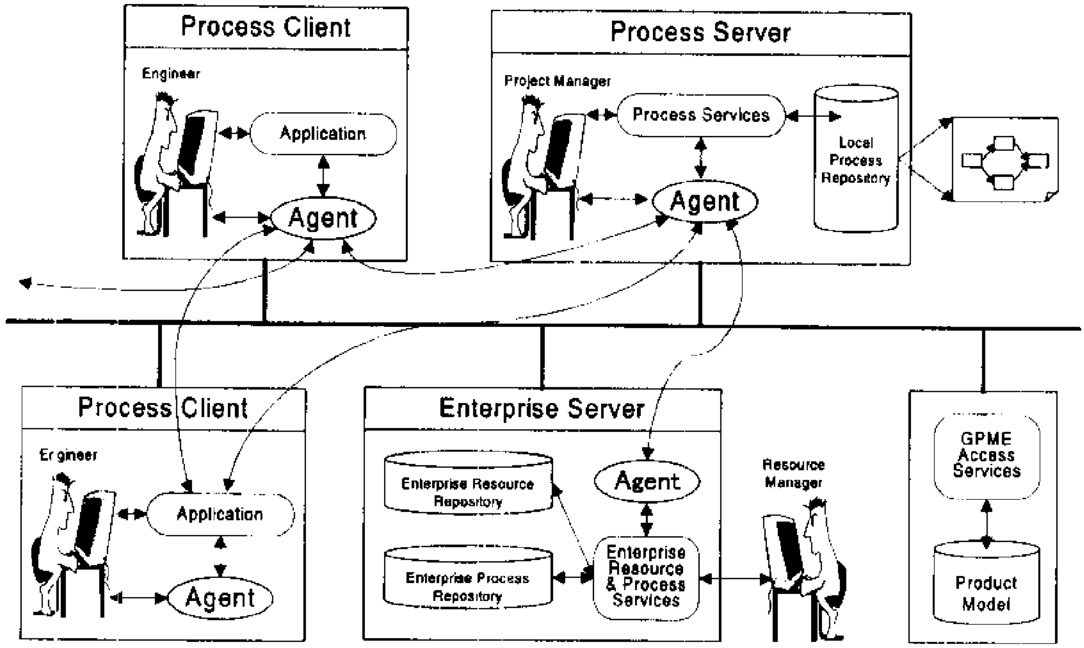


Figure 5. Prototype system architecture

Function of Intelligent Agent

Intelligent Agent is a state-of-the-art IT[5, 6, 7]. Intelligent Agent is regarded as a core technology to realize engineering collaboration support system using process model. One of the feature of Intelligent Agent is to support system operation and participants in the system intelligently and autonomously. Agent's intelligence does not mean all the knowledge that engineers will depend on when they meet engineering tasks. In this architecture, Agent is a kind of facilitator who may provide engineers useful information, i.e., state of other engineer, past examples, alternative options. Those information will help them to make decision. Agent has learning capability as well. It has knowledge that what information should be offered to its engineer depending on cases, and this knowledge will be modified and enhanced through the interaction with its engineer or other participants. As for process model, this learning capability enables to obtain dependencies among WorkElements dynamically.

In Active Process system, Intelligent Agent has four functions as follows:

1. Communication function with other agents.
2. Event Monitoring function.
3. Exception Detection function from events information.
4. Task Execution function in compliance with events/exceptions.

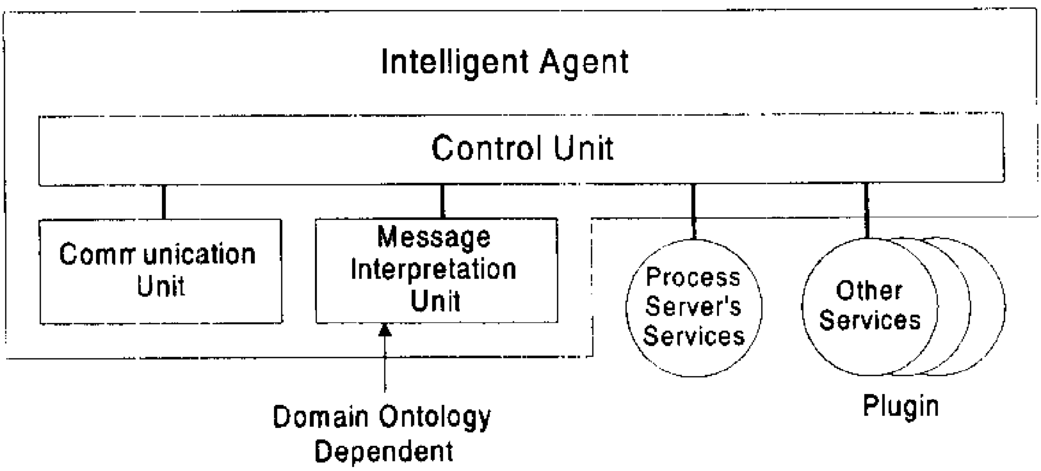


Figure 6. Intelligent Agent Architecture

Intelligent Agent Architecture

Figure 6 shows architecture of Intelligent Agent implemented in the prototype system. It consists of 3 modules: communication function module, message interpretation function module and local service module. Intelligent Agent is implemented with Java.

KQML(Knowledge Query and Manipulation Language) is used as communication protocol. Contents of message are expressed in the form of KIF(Knowledge Interchange Format). Terminology in messages conforms to domain specific ontology. The part which is domain specific is just only ontology. This means that Intelligent Agent architecture is applicable to any domain just only to change ontology.

Process Server and Enterprise Resource Server

Process Server has three functions. It defines a process(activities and inter-relations among them), supports resource assignment and monitors progress state.

(a) Process definition: Activity level process is defined through GUI. It may be generated newly or may be extended based on templates which will be stored in Process Repository. Processes defined in Process Server are able to be separated and spawned to several client participants recursively. That is, one process may be assigned to an engineer and he/she can separate it into pieces and assign them to other engineers. Information defined here are as follows: task contents, schedule, dependencies, responsible workers, etc.

(b) Resource assignment: Available engineers lists and machines lists are available as resource information. Those resource information will be referred during process definition. Processes are optimized during operation if idle resources(we call them slacks) are found. We call this approach slack driven process management.

(c) Progress state monitoring: Process server asks all agents who are responsible for each engineer about the progress of their activity and collects the current status. It will be displayed and broadcasted to all agents. Every engineer can know the whole progress status in real-time through his/her agent.

Enterprise Server integrates information needed for collaboration support, that is, process repository, resource information and journals information. The contents of those information are as follow.

(a) Process repository: This contains typical process templates which will be extracted out of journal information in previous projects.

(b) Resource information: This contains organization information(departments of managers and engineers), personnel information(name, dept., agent name, current project, skills, etc.) and available facility resource information(tools, machines, data, operation schedule, constraints, etc.).

(c) Journal information: This contains communication logs between agents and dependency data generated from communication logs. After the project finishes, more detailed process may become evident with analyzing journal information. This helps to improve engineering process. Moreover, engineering knowledge can be shared and organized.

Running Image of this prototype system

Here we show how Agents work in our prototype system.

(a) After a process is defined, Agent of a project manager notifies agents of each engineer that activities are assigned. If an engineer accepts, communication link between a project manager and an engineer is established and dependencies or progress state information will be exchanged through this channel.

(b) After activities are assigned, each activity is carried out under agents supports. Agents monitor its engineer and provide appropriate information when needed, i.e. how to notify other engineers if a design change is occurred.

(c) During operation, agents accumulate its engineer's work logs as an engineering task journal and agents retrieve dependencies out of the journal. For example, an engineer may have to look for an appropriate engineer to notify his design change for the first time. But after that, his/her agent will notify automatically pertinent agents when a design change on the same part occurs.

(d) On the other hand, project manager can manage the progress status based on the information reported by all the agents. It enables him/her to re-order engineers to catch up with pre-defined schedule.

Case Study

The objective of this case study is to examine the validity of Active Process approach and retrieve extract problems. Building the Blocks game is met here. This scenario is intended to be simple but still cover as many as possible features of process description and enactment that can be seen in typical collaborative engineering problems.

Game Scenario

An 8*8 area(Figure 7) needs to be built with four kinds of blocks which have different colors(Red Block:RB, Blue Block:BB, Green Block:GB, Yellow Block:YB). Three constraints are defined:

1. You can't place a block to a certain area if its under area is vacant.
2. Cost and time consumption are attributed to each block locating
   (RB: $1, 4min., BB: $2, 3min., GB: $3, 2min., YB: $4, 1min.)

Cost must be less than $160 and time consumption must be less than 160 min.

3. All adjoining blocks in horizontal must have different color blocks.

The objective of this game is to fill all the area with the cheapest cost and the least time consumption. The number of participants are not limited but there are two kinds of players: Project manager and workers. Project manager assigns tasks to workers. The lowest cost and time consumption as a whole is desired. In this game, all participants are required to collaborate each other to carry out their assigned tasks under those constraints.

Analogy With Models

Numbered shapes in Figure 7 comprised of several small squares represent Activities in Active Process model. Small square represents a Work Element. Tasks are assigned to workers in the unit of Activity. For example, some worker may have to deal with Activity 1, 8and 11 Planned workflow (right-hand in Figure 7) represents activities dependencies which comes from geometry constraints of replacement. Workers can begin his task in the unit of Work Element, that is, he/she can place a block on a new activity area even if current activity has not finished yet.

Following two points should be noticed:

1. Concurrent Activity Execution: Activity 4 may be processed though Activity 1 has not been finished for example.
2. Necessity of negotiation: Adjoining blocks in Activity 1 and 2 must have different colors so that workers who are responsible for each activity must negotiate each other.

Dependencies of Work Elements will be accumulated during operation by agents. How to support effectively workers and managers based on acquired dependency information is the point of this prototype system.

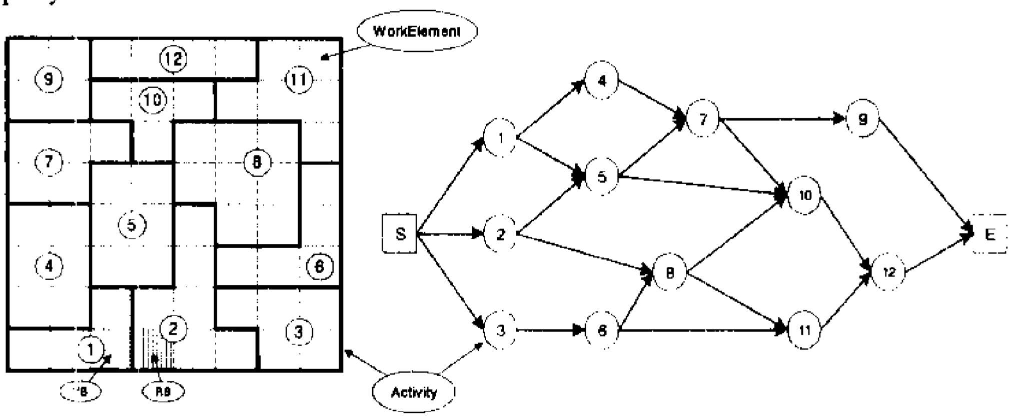


Figure 7. Activities and Planned Workflow

Verification through Prototype System

Active Process has two aspects: process modeling capability and process manipulation to support engineering collaboration. Prototype system was useful to:

Visualize and embody modeling concepts.

Examine process manipulation mechanism (in particular, agents' motion).

The game was played by several people and we gained following insights.

First, to know information needed for a task, neighboring status in this case, contributes significantly toward the increase of task efficiency. Though the case here seems to be simple, it is important to be able to obtain necessary information timely by agents, without knowing whom should be appropriate participants with regard to the information. Agents asks Process Server an appropriate participants' agent and establish communication with the agent to gain desired information. This game proved agents to be useful greatly information collecting.

Secondly, dependency relationships accumulated in real-time enable prompt reaction to changes of circumstances and prevention of inconsistency occurrence because of lack of communication.

Thirdly, the possibility of 'super concurrency' was suggested. In this game, players could grasp progress status, in the unit of not Activity but WorkElement. They could start their tasks(WorkElement) though the previous activity had not finished yet. Actually the manager could grasp the whole progress status in detail so that he/she could give instructions to players timely. This means that the game management becomes smooth and the total playing time will be shortened.

Lastly, players operation journals were recorded, so that strategy planning for high-scoring became easy

As shown above, this prototype system proved that process models and agents mechanisms in Active Process was effective in supporting multiple participants' collaboration and was able to manage the whole process smoothly.

Conclusions

In this paper we discussed collaborative engineering support system using process model and agents approach. To connect process model with product model and enterprise resource data, dependencies of product components are transformed into dependencies of engineering processes and those information are used as workflow data. As for unpredictable dependencies(Work Element dependencies), agents collects dependencies monitoring engineer's operation. Followings are our conclusions.

- We proposed Active Process model which has capability of learning dependencies of finer granuality task level. It is difficult to do for existing workflow tools.
- Active Process can meet dynamic(unpredefinable) process modeling as well as static(predefinable) process modeling.
- Intelligent-Agents approach are adopted as Active Process mechanism.
- Active Process was verified through prototype implementation using a simple scenario(building the blocks).
- Modeling-prototyping-case study approach was so effective that we could gain many insights.

Future research will be focused on refining Active Process framework in terms of modeling(description capability) and mechanism(agent-approach), through examining scenarios in various design phases in shipbuilding.

Acknowledgements

This research has been carried out under the sponsorship of the Nippon Foundation, and the authors would like to express their cordial thanks to them. They would also like to express their gratitude for the guidance and encouragement received from Prof. Takeo Koyama, University of Tokyo.

References

1. Ship & Ocean Foundation, 1997 Report on Research and Development of Advanced CIM for Shipbuilding based on Knowledge Sharing.
2. Ship & Ocean Foundation, 1998 Report on Research and Development of Advanced CIM for Shipbuilding based on Knowledge Sharing.
3. Jin, Y., ASCAD: "Agent-supported collaborative design", In CIRP 1997 International Design Seminar Proceedings: Multimedia Technologies for Collaborative Design and Manufacturing, Los Angels, CA, 1997.
4. Jin, Y. and Stephen C-Y. Lu "An Agent Supported Approach to Collaborative Design", in CIRP Annals, Vol. 47/1, 1998.
5. Kaelbling, L. P., "A situated automata approach to the design of embedded agents", SIGART Bulletin, 2(4,) 1991.
6. Maes, P., "The agent network architecture(ana)", SIGART Bulletin, 2(4), 1991.
7. Maes, P., "Modeling adaptive autonomous agent", In C. Langton, editor, Artificial Life Journal, volume 1. MIT Press, Cambridge, MA, 1994.

Index of Contributors

INDEX OF CONTRIBUTORS

INDEX OF CONTRIBUTORS

Volume 2