

The Age and Growth Program Database*

Jonathan A. Short and Delsa M. Anderl

Resource Ecology and Fisheries Management Division
Alaska Fisheries Science Center
National Marine Fisheries Service
National Oceanic and Atmospheric Administration
7600 Sand Point Way NE
Seattle, WA 98115

Email address for contact author: jon.short@noaa.gov

* *The information in this document has not been subjected to formal peer review. Please use footnotes and the following format when citing this document:*

Short, J.A., and D. M. Anderl. 2012. The Age and Growth Program Database. Resource Ecology and Fisheries Management Division, Alaska Fisheries Science Center, NMFS, NOAA, 7600 Sand Point Way NE, Seattle WA 98115. [Available from: <http://www.afsc.noaa.gov/REFM/Age/default.htm>.]

Age and Growth Program database design and software applications

Age data generated by the Alaska Fisheries Science Center (AFSC) Age and Growth Program are entered into a Microsoft Access™ database called AGEDATA, which is a repository of all fish ages generated during the age determination process. The AGEDATA database uses form-driven menus and custom reports for users to interface with the database to perform functions such as entering age estimates, printing age-related forms, and tracking the physical location of specimens. Each database task is accomplished with Visual Basic for Applications (VBA) macros to minimize redundant tasks.

Database structure and design evolution

The database was originally created using RBASE 2.5 in April 1990 to move age data from a Burroughs system to a more accessible application. In 1993, the database was converted from RBASE to Access 1.0 (Microsoft Jet Engine 1.0) because of cost and projected long-term support. The original design had separate data tables for each data collection. This approach was chosen to provide structural integrity in a shared network environment so that users were not making data updates on the same table simultaneously. Despite this measure, database corruptions occurred 10 to 15 times a

year. The database software was updated to current versions of Access every 3 to 4 years, the latest being the conversion to Access 2007 in May 2011.

The 2002 version of Access and the underlying Jet 4.0 engine appeared to be considerably more stable than the original database engine, so a prototype was created to test database integrity in a consolidated data structure. The new structure held up remarkably well to a series of tests involving multiple users updating records and forms simultaneously, and the new structure was implemented in production mode in May 2006. Significant improvement was made to the database structure, and over 200 tables were consolidated into eight relational data tables. The new relational structure provided more flexibility to implement data constraints, create database summaries, and minimize orphaned data.

Most current database designs are relational with normalization to the third normal form. Relational databases are collections of tables linked together by primary and foreign keys. Normalization is a sequence of up to eight database standards or forms. A database that meets the first three standards, or third normal form, has eliminated duplicate records and repeating groups and contains only similar types of data in each table, with each field corresponding to a single primary key.

Normalization to third normal form was a goal in the restructuring of AGEDATA that was not fully met partly due to Microsoft Access restrictions. A primary restriction was that Access does not allow manual updates to records in a query that contains linked tables. The requirements of normalization would separate descriptive, but repetitive, fields from frequently edited fields into a second data table. Because it is not possible to create an updateable linked query joining tables together, data updates typed directly into the table or query would have to be made without useful identifying information.

Data types and transfer syntax

Within the database, data access is driven by dynamically created queries, Data Access Object (DAO) recordsets and ActiveX Data Object (ADO) recordsets. Some functions recreate queries each time they are triggered. Although it is considered an outdated method, this technique is attractive for debugging and design of data entry forms compared to recordsets because the query is persistent after the form is closed. Both DAO 3.6 and ADO 2.8 recordsets are utilized in this database. While ADO is the newer method, it does not provide some features, such as subform requerying, that are available with DAO recordsets.

Importing data into AGEDATA is done by querying a linked Oracle database or by importing text and Excel files. AFSC Oracle databases (currently Oracle 10G) are accessed by building Access queries of linked tables or with pass-through queries. Pass-through queries perform considerably faster than standard Access queries but are created in the text-only designer using PL/SQL syntax. On the other hand, standard Access queries on linked Oracle tables can be designed in the graphical query builder.

When the age determination process is completed and a sample has been checked for errors, the age data and related information are exported and archived. Data export of the single final age estimate is accomplished by updating linked Oracle tables or through automatically generated comma-separated value (CSV) text files. Access update queries can update Oracle data tables as long as the DSN user account has

adequate permissions. Data exports to text files are accomplished using the 'DoCmd.TransferText' syntax. This method requires an export profile that is generated by manually exporting data to a text file with the same data fields and properties. Age data are archived into a single table on a SQL Server 2005™ database and into a sample-specific text file, and are then deleted from the Access data file. The SQL Server table contains age data from 1982 to the present and currently has data for over 780,000 specimens. Whenever possible, historic data have been converted to current formats to maintain consistency.

Age-related statistics are calculated with the AGREE and RANGES programs, which are compiled Visual Basic .NET executables. These programs connect to the database using ADO recordsets, make necessary calculations in Visual Basic, and output to a PDF file using the iTextSharp library.

Data redundancy and backup

The database instability we experienced in the past has led to an extensive backup schedule. Active data are backed up with a stand-alone application that was written in Visual Basic .NET to create sequential copies of the database hourly and monthly. The hourly copy protects against data loss due to database corruption, and the monthly copy protects against inadvertent data deletions that are not initially detected. The Access data file resides on a Windows network server that is mirrored to multiple hard drives and is backed up regularly to a tape system. This is added protection against database corruptions and catastrophic power surges that could destroy server and backup hard drives. Archived age data reside on a SQL Server 2005 database that is on a 2-week tape backup cycle and a monthly "frozen image" that is stored off-site.

Database web manual

The AGEDATA database has an accompanying web-based manual. This application is an indexed collection of HTML-based pages describing each of the AGEDATA functions and methods. Help buttons throughout the Access database are linked to corresponding pages of the

help application, so users can quickly locate useful information. The help application was built with Macromedia's DreamWeaver 8 and can be edited with any text editor. A table of contents and site index was created with DreamWeaver functionality.

Data recorded in the Age and Growth Program database

Age data generated by the AFSC Age and Growth Program are entered into the AGEDATA database described above. Age readers use a number of different codes to define the qualities of fish age estimates and aging structures. Codes describing aging method, readability, edge type, and physical characteristics of the aging structure are relayed to age data users together with age estimates.

Following age determination, the age reader enters the following information into the database fields below:

Age The age estimate for a given aging structure. If the specimen is "unageable" (i.e., an age cannot be determined), the assigned value in this field is "-1". If an age reader is only comfortable assigning an age range, the minimum possible age is recorded in this field.

Method The method used to arrive at each age estimate. Method codes are listed on page 4.

Read Code "Readability" code, or the degree of difficulty of interpreting an otolith pattern or an indication of problems associated with a specimen. Readability codes are listed on page 5.

Edge Type A characterization of the growth observed at the edge of an aging structure. Filling in this field is optional but highly recommended, especially for young otoliths. Edge type codes are listed on page 5.

Max. Age For difficult specimens that can only be assigned an age range, a maximum age is estimated. (The minimum age is recorded in the Age field, as described above.) When an age range is provided, the Read code is "3".

Comments Code letters used to document characteristics of the aging structure or growth pattern. Comment codes are listed on page 5.

Types of age estimates

To understand the age determination process, it is necessary to understand the variety of age estimates generated by age readers:

Raw Age The age reader's original age estimate.

Test Ages Specimens randomly selected and aged by a second age reader (i.e., a tester). Typically 20% of the total sample.

Updated Age A second age estimate that differs from the raw age and that takes precedence over the raw age. The assignment of an updated age is most often the product of the discrepancy resolution process necessitated by differences between reader and tester age estimates. Sometimes a second age estimate is generated by the age reader for selected otoliths outside the test sample because of apparent problems in age interpretation discovered during the resolution process or because examination of age-length outliers suggested a problem with the raw age.

Final Age The final age estimate for a specimen. Final ages are generated by AGEDATA and are a combination of raw and updated ages. When an updated age differs from the raw age, only the updated age is carried to the final

age column. A final age of “-1” means that the specimen was “unageable”, due to reasons such as 1) otoliths have patterns too ambiguous for a reader to assign a specific age, 2) no otoliths, or more than two otoliths, are in a vial, 3) the two otoliths in the vial belong to two different fish, 4) otoliths are crystallized or are impossible to age due to other physical characteristics, 5) the reported fish length is not appropriate with respect to otolith size.

The relationship between fish age and fish size can be used to identify three classes of outliers which are examined by age readers prior to releasing age data to end users:

- 1) If a specimen is identified as an outlier from the age-length relationship and the reported fish length is inappropriate with respect to otolith size, the specimen is considered “unageable” and a final age of “-1” is assigned. However, a raw age is generally assigned.
- 2) If a specimen is identified as an outlier from the age-length relationship but the reported fish length appears appropriate with respect to otolith size, the specimen is recorded as a fast-growing or slow-growing fish. The specimen is considered “ageable” and is assigned a final age based on the raw or updated age estimate.
- 3) The fish age-weight relationship is used to identify gross discrepancies. The specimen is considered “ageable” and is assigned a final age based on the raw or updated age estimate.

Should any specimens appear as outliers under these criteria, the readers will document this in a brief memorandum sent to the age data user.

Method codes

There are a variety of preparation methods that can be used to determine age from otoliths and other structures. Two of the most common methods used at the AFSC are surface

examination and the break-and-burn technique. However, certain species require special methods such as thin-sectioning. (Please see Goetz et al., 2012, for a more detailed description of standard AFSC otolith preparation methods.) Age readers are required to record the method used to determine age for each specimen so that this information can be transmitted to age data users. For certain species, a variety of methods may be used within a single collection of specimens. Methods and their corresponding codes are listed below:

[blank]	surface examination
U	untreated otolith cross section
B	break-and-burn
V	break-and-bake
T	thin section
C	stained cross section
G	ground otolith
M	break-and-bake followed by break-and-burn

Readability codes

In addition to age estimates, readers must record readability codes. These codes describe aging structure quality and the ability of the reader to estimate age from the growth pattern.

- 1 Clear growth pattern.
- 2 A single age estimate can be generated with a variable level of confidence.
- 3 Very difficult; a reader can only assign an age range.
- 4 “Unageable” due to problems interpreting the growth pattern (e.g., extremely faint or too many checks). No age range can be assigned, and final age is “-1”.
- 5 “Unageable” due to physical characteristics such as crystallization, chalkiness, or damage. Final age is “-1”.
- 6 “Unageable” due to collector error such as more than two otoliths in a vial, discrepancies exist between otolith size and recorded fish length, etc. Final age is “-1”.

Edge type codes

Documenting growth on the edge of the otolith is important to age determination (Matta and Goetz, 2012). Recording the edge type code for each specimen is not mandatory, but highly recommended. These codes describe the growth state on the edge of aging structures:

- 0** Strong translucent growth zone on edge
- 1** Strong translucent growth zone on edge with a slight halo of opaque growth
- 2** Approximately ¼ year of opaque growth on edge
- 3** Approximately ½ year of opaque growth on edge
- 4** Full year of opaque growth on edge
- 5** Full year of opaque growth on edge with a translucent growth zone beginning to form

For edge types 2 through 4, the width of the opaque growth zone at the edge is graded by comparing it to the previous fully-formed opaque zone.

Comment codes

Comments are largely left up to the discretion of the age reader. These codes are used to document

quality, growth pattern, or other aging structure properties. Any combination of the following letters may be used:

- A** collector error occurred
- B** broken (damaged) aging structure
- C** handwritten comments on age forms
- D** difficult growth pattern
- E** edge is in question
- F** faint annual mark
- G** growth pattern is in question
- H** break-and-burn problems
- I** chalky
- J** checky
- K** crystallized
- L** “unageable”
- M** more than two otoliths in vial
- N** slow-growing fish
- O** fast-growing fish
- P** recorded fish length is too large with respect to otolith size
- Q** recorded fish length is too small with respect to otolith size
- R** special-project specimen
- T** specimen has been photographed
- X** clear (photo-quality) growth pattern

Literature cited

Goetz, B. J., C. E. Piston, C. E. Hutchinson, C. G. Johnston, and M. E. Matta.

2012. Collection and preparation of otoliths for age determination. *In* Age determination manual of the Alaska Fisheries Science Center Age and Growth Program (M. E. Matta and D. K. Kimura, eds.), Chapter 3. NOAA Professional Paper NMFS 13.

Matta, M. E., and B. J. Goetz.

2012. Otolith growth pattern interpretation. *In* Age determination manual of the Alaska Fisheries Science Center Age and Growth Program (M. E. Matta and D. K. Kimura, eds.), Chapter 2. NOAA Professional Paper NMFS 13.