A DESIGN FOR A NEW CATALOG MANAGER AND ASSOCIATED FILE MANAGEMENT FOR THE LAND ANALYSIS SYSTEM (LAS)

Cheryl Greenhagen
TGS Technology, Inc.
EROS Data Center

A DESIGN FOR A NEW CATALOG MANAGER AND ASSOCIATED FILE MANAGEMENT FOR THE LAND ANALYSIS SYSTEM (LAS)

Cheryl Greenhagen

TGS Technology, Inc.

EROS Data Center

1 INTRODUCTION

Due to the large number of different types of files used in an image processing system, a mechanism for file management beyond the bounds of typical operating systems is necessary. The TAE Catalog Manager was written to meet this need.

Land Analysis System users at the EROS Data Center (EDC) encountered some problems in using the TAE catalog manager, including catalog corruption, networking difficulties, and lack of a reliable tape storage and retrieval capability. These problems, coupled with the complexity of the TAE catalog manager, led to the decision to design a new file management system for LAS, tailored to the needs of the EDC user community. This design effort addressed catalog management, label services, associated data management, and enhancements to LAS applications.

This paper briefly describes this alternate design for file management of an image processing system.

2 DESIGN GOALS

The following goals were set for the catalog manager design project:

Combine all components to create an integrated system.

Provide a single user interface for both VMS and UNIX.

Increase VMS/UNIX compatibility.

Provide the functional capabilities of the current TAE Catalog Manager, and incorporate the additional capabilities required by local users.

Provide a simple and flexible system, minimizing the amount of code to write and maintain, and minimizing the impact of adding vector, tabular, or other types of data in the future.

Minimize the impact on existing LAS Application Programs.

Minimize system maintenance and support.

Promote data integrity, minimize corruption, and allow several recovery procedures.

Eliminate performance bottlenecks wherever possible.

Simplify transfer of data between computer systems, for both network and tape transfer.

3 CATALOG MANAGER DESIGN APPROACH

Several possible designs for a new catalog manager were considered. The design chosen is a very simple one, based on the host operating system's disk file manager. It provides the same user interface on both VMS and UNIX.

Each user will have his own "catalog", a subdirectory named "CM" in the user's directory. Cataloged disk files are those residing in the user's CM directory.

In addition to the user's files, a set of sequential ASCII files is maintained in the CM directory for catalog manager bookkeeping functions, including tracking cataloged tape files, handling aliases, and corrupt file detection.

The catalog manager design also provides two types of offline storage: short-term and long-term archive tapes.

Files that have been copied to short-term archive tapes are part of the user's catalog and are accessible by the software; these files will be automatically retrieved if requested for input. Information about these files is found in the tape catalog in the user's directory.

Files that have been copied to long-term archive tapes are removed from the user's catalog. No information about these files is stored online.

The catalog manager file name (TAE name) will have the same format on all operating systems. Application programs accept and display the TAE name, so the user can move from one system to another without changing the TAE naming convention, and without being familiar with the host operating system file name. The TAE user does not see the host name except by explicit request.

The new catalog manager will translate the TAE name to a recognizable host name - in fact, there is a close correspondence between the TAE name and the host name. This makes it easier to use operating system tools and other non-TAE software to process cataloged files.

The catalog manager file name has the following format:

[user.directory1.directory2 ...]filename;extension

The username and directory name specification is enclosed in square brackets. "User" is any valid username on the system. The directory names may contain alphabetic characters and numeric characters. The username and directory names are separated by periods. The filename and extension name follow. They may contain alphabetic characters, numeric characters, and periods (".") . The filename and extension name are separated by a semicolon.

Each of the components (directories, filename and extension) of the catalog manager name may be up to 39 characters long. The whole name may be up to 248 characters long.

The username and directory name specification is optional. If not specified, the current directory is assumed. By using the full file specification, the user may access any file in any directory, subject to the file protections of the operating system.

The following example illustrates TAE name - host name correspondence:

TAE name: [SMITH.NY.MSS]STRETCHED.IMAGE;HIS

VMS host name: [SMITH.CM.NY.MSS]STRETCHED_IMAGE.HIS

UNIX host name: /smith/cm/ny/mss/stretched_image.his

4 CATALOG MANAGER UTILITIES

The current TAE catalog manager is a constantly-running program. By contrast, the new catalog manager is just a collection of utilities and support routines. A brief overview of these utilities follows.

4.1 Create, Delete, And List Aliases

The alias utilities allow the user to create (assign), delete, and list aliases.

An alias is just a short name for another string - the alias text. The use of aliases is supported in Catalog Manager program parameters. When an alias name is used, as a parameter or part of a parameter, the alias text is substituted. Each user will have his own set of defined aliases.

Alias information will be kept in an ASCII text file in the user's CM directory. The alias name is a 9-character string beginning with "$". The alias text is a 255-character string. In order to make the file editable by EDT, which does not handle records longer than 255 characters, the file will consist of pairs of records. The first record in every pair will contain the alias name; the second record of the pair will contain the alias text.

The following example illustrates the alias file.

```
$WINDOW
"(200,200,100,100:2,3,4)"

$IN
"NEWYORK.IMAGE1(:1,2) + NEWYORK.IMAGE2(:3)"
```

4.2 Copy, Rename, Delete, And List Files

The catalog manager includes utilities to copy, rename, delete, and list cataloged files. The list utility includes the capability to list selected file attributes (e.g., file creation date), or image attributes (e.g., number of bands, data type) in addition to the file name.

4.3 Create, Delete, Display, And Set Directories

The directory utilities allow the user to create and delete directories in his directory tree, and to display or set his current, or "working", directory.

4.4 Copy Files To And From Archive Tapes

The tape-handling utilities allow the user to copy cataloged files to short-term or long-term archive tapes, and to retrieve files from archive tapes.

The physical format of the archive tapes will follow an ANSI standard, providing convenient multiple tape handling and usability on many systems.

The catalog manager will generate some archive information in addition to the ANSI label information.

The catalog manager information for archive tapes includes the file name (TAE name), file status, a tape library identifier, the creation timestamp of the file, the size of the file, the tape length and density, and the last access date of the tape.

For short-term archive tapes, the catalog manager stores this information about the tape and the files in a sequential ASCII file in the user's CM directory.

For long-term archive tapes, the same information is written to the end of the archive tape. No information is stored on-line for long-term archive tapes.

4.5 Support Functions

The catalog manager support utilities include statusing utilities and tape management utilities for use by the computer operations staff.

5 SYSTEM-LEVEL ENHANCEMENTS

5.1 Associated Files

In the course of processing raster images, a user may create several types of associated data that should reside with the image for easy reference. This information will be stored in a set of files associated with the image. Label information for the image will also be stored in associated files.

Files will be associated implicitly, by filename. All files in a directory with the same filename, but different extensions, are considered to be associated.

The existence of implicitly-associated separate files for the label information and associated data gives great flexibility, simplifies the task of accessing label information, and allows the image and its associated files to be easily treated as a "data set" for tape archive and retrieval, and file transfer.

Associated files will have records of variable length and different data types. Wherever possible, however, these files will be sequential in organization and will share a common feature: each record will begin with 3 ASCII fields of known size, containing the length of the record, the data type of the record contents, and a key.

Any extension may be used for an associated file. However, some standard extensions will be used in naming frequently used types of files:

DDR	Data Descriptor Records File
CWT	Convolution Weights File
DPF	Display Parameter File
POINT	Graphics Overlay Point File
LINE	Graphics Overlay Line File
POLY	Graphics Overlay Polygon File
ANNOT	Graphics Overlay Annotation File
HIS	History File
IMG	Image File
LUT	Look-up Table File
STAT	Statistics File

5.2 Input File Handling

If an application program attempts to access a file that is not online, the application program will automatically attempt to retrieve the file from the user's short-term archive tapes. If successful, the

file will remain on disk when processing completes. However, a user may not access files from another user's archive tapes.

5.3 Output File Handling

The EDC user community requires that cataloged files be unique in the catalog - the same file may not exist both on disk and on tape.

To ensure uniqueness, before opening a new output file, application programs will check both the disk catalog and the store tape catalog to ensure that the user's catalog does not already contain a file by the specified name. If a specified output file name already exists in the catalog, an error message will be generated, and the new file will not be created.

5.4 Corrupt File Detection

Corrupt file detection will be implemented by means of a convention followed the application programs. Whenever a file is opened for write access, an identifying entry is written into a corrupt file list in the user's CM directory. The entry will be deleted when the file is closed. If the program aborts or the system crashes before the file is closed, the entry will remain in the corrupt file list. A file found in the corrupt file list when no application program is using it is assumed to be incomplete, and may be replaced.

5.5 Aliases

LAS applications will be enhanced to handle aliases in parameters, by simple string substitution.

5.6 Image I/O Libraries

Currently, three different methods of reading and writing image files exist in different software systems at EDC:

TAE I/O, based on the XI package from Century, is available on both VMS and UNIX. TAE I/O stores all bands for an image in one file.

LAS I/O, based on VMS RMS I/O, is available on VMS only. LAS I/O stores each band in a separate file.

NEWLAS I/O, written at EDC and based on TAE I/O, is available on both VMS and UNIX. NEWLAS I/O stores all bands for an image in one file.

NEWLAS I/O is EDC's target image processing system. Until the target is reached, application programs will provide a consistent "virtual image" approach across all of the image processing systems. This means that the user will always view an images as a cohesive set. The applications will automatically access the correct host file(s) for the user-specified catalog name, isolating the user from the physical location and configuration of the image bands.

5.7 Build PDF From History File

A program will be provided to create a Proc Definition File from a history file.

6 SUMMARY - SYSTEM ENHANCEMENT BENEFITS

The new file management design will provide the many benefits, including improved system integration, increased flexibility, enhanced reliability, enhanced portability, improved performance, and improved maintainability.