MAY 21 1996

ENGINEERING DATA TRANSMITTAL

2. To: (Receiving Organization)
   Distribution

3. From: (Originating Organization)
   Media Systems
   Application/Media Management
   Systems/BCSR

4. Related EDT No.:

5. Proj./Prog./Dept./Div.:
   Engineering/Interim
   Stabilization Project/Tank
   Farm Transition Projects

6. Cog. Engr.:

7. Purchase Order No.:
   N/A

8. Originator Remarks:

9. Equip./Component No.:
   N/A

10. System/Bldg./Facility:
    VIDON

11. Receiver Remarks:

12. Major Assm. Dwg. No.:
    N/A

13. Permit/Permit Application No.:
    N/A

14. Required Response Date:

<table>
<thead>
<tr>
<th>Item No.</th>
<th>Document/Drawing No.</th>
<th>Title or Description of Data Transmitted</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WHC-SD-WM-SDD-069</td>
<td>PAMS Photo Image Retrieval Prototype System Design Description</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Approval Designator (F)</th>
<th>Reason for Transmittal (G)</th>
<th>Disposition (H) &amp; (I)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(see WHC-CM-3, Sec.12.7)</td>
<td>2. Release 5. Post-Review</td>
<td>2. Approved w/ comment 5. Reviewed w/ comment</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reason</th>
<th>Disp.</th>
<th>Name</th>
<th>Signature</th>
<th>Date</th>
<th>MISIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>Cog. Mgr. Darwin K. Griswold</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>QA: Kelly C. McBride</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Author: Max L. Conner</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Signature of EDT Originator</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5/10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Authorized Representative Date for Receiving Organization</th>
<th>Cognizant Manager Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DOE APPROVAL (if required)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl. No.</td>
</tr>
<tr>
<td>Approved</td>
</tr>
<tr>
<td>Approved w/comments</td>
</tr>
<tr>
<td>Disapproved w/comments</td>
</tr>
</tbody>
</table>

BD-7400-172-2 (04/94) GEFO97
PAMS PHOTO IMAGE RETRIEVAL PROTOTYPE
System Design Description

Max L. Conner
Westinghouse Hanford Company/Boeing Computer Services Services
Richland, WA 99352
U.S. Department of Energy Contract DE-AC06-87RL10930

Key Words: System, Design, Description

Abstract: This System Design Description (SDD) documents the detail design of the Photo Audio/Visual Management System (PAMS) Photo Image Retrieval Prototype (PPIRP) subsystem. This SDD shows how the software is structured to satisfy the requirements identified in the PAMS PHOTO IMAGE PROTOTYPE Requirements Document. It is a description of the software structure, software components, interfaces and data that make up the PPIRP subsystem.

Trademarks: DOS is a trademark of Microsoft Corporation.
Microsoft Windows is a trademark of Microsoft Corporation.
Targa+ is a trademark of Truevision.
Windows is a trademark of Microsoft Corporation.

TRADEMARK DISCLAIMER. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof or its contractors or subcontractors.

Printed in the United States of America. To obtain copies of this document, contact: WHC/BCS Document Control Services, P.O. Box 1970, Mailstop X6-08, Richland WA 99352, Phone (509) 372-2420; Fax (509) 376-4989.

Approved for Public Release

A-6400-073 (10/95) GEF321
PAMS PHOTO IMAGE RETRIEVAL PROTOTYPE

SYSTEM DESIGN DESCRIPTION

Prepared By:
Max L. Conner
IRM, HIMACS Systems Development
# TABLE OF CONTENTS

## 1.0 INTRODUCTION

1.1 Purpose .................................. 1  
1.2 Scope .................................. 1  
1.3 Overview ................................ 1  
1.4 Definitions ................................ 2

## 2.0 DECOMPOSITION DESCRIPTION

2.1 Module Decomposition ................................ 3  
  2.1.1 Logical Data Model ......................... 3  
  2.1.2 Logical Process Model ....................... 5

## 3.0 DEPENDENCY DESCRIPTION

3.1 Intermodule Dependencies ................................ 7  
3.2 Interprocess Dependencies ................................ 7  
3.3 Data Dependencies .................................. 7  
3.4 Conversion Dependencies ................................ 7

## 4.0 INTERFACE DESCRIPTION

4.1 Module Interface ................................ 8  
4.2 Process Interface ................................ 8

## 5.0 DETAILED DESCRIPTION

5.1 Module Detail Design ................................ 9  
  M1.0 IMAGE CAPTURE. ................................ 10  
    M1.1 Initialize .................................. 11  
    M1.2 Select Standard Setup ....................... 13  
    M1.3 Create Standard Setup ....................... 14  
    M1.4 Get Image .................................. 15  
    M1.5 Manipulate Image ............................. 16  
    M1.6 Save Image .................................. 17  
    M1.7 End ........................................ 20  
  M2.0 IMAGE RETRIEVE ................................ 21  
    M2.1 Version Control .............................. 22  
    M2.2 Initialize .................................. 24  
    M2.3 Image Query ................................ 26  
    M2.4 End ......................................... 29  
  M3.0 ADMINISTRATION ................................ 30  
    M3.1 User Access Control (PAMS) .................. 30  
    M3.2 List of Images With Incomplete Photo Data .. 30  
  5.2 Data Detail Design ................................ 31  
    5.2.1 Table: PAMS IMAGE FILE ..................... 31  
    5.2.2 Table: PAMS PHOTO FILE ..................... 33  
    5.2.3 Table: PHOTO FILE FIELDS .................. 34  
    5.2.4 APPLICATION.INI file ........................ 35  
    5.2.4.1 Capture Initialization File ............... 35
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2.4.2</td>
<td>Retrieval Initialization File</td>
<td>37</td>
</tr>
<tr>
<td>5.2.5</td>
<td>TARGASET.DAT file</td>
<td>39</td>
</tr>
<tr>
<td>5.2.6</td>
<td>VERSION.DAT file</td>
<td>40</td>
</tr>
<tr>
<td>5.2.6</td>
<td>User Query Save Files</td>
<td>41</td>
</tr>
<tr>
<td>5.3</td>
<td>General Design</td>
<td>42</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Application</td>
<td>42</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Command Buttons and Menus</td>
<td>42</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Help Files</td>
<td>42</td>
</tr>
<tr>
<td>5.3.4</td>
<td>Program Setup</td>
<td>43</td>
</tr>
</tbody>
</table>

6.0 REFERENCES .................................................................................................................. 44
List Of Figures

Figure 1 Logical Data Model .................................................. 3
Figure 2 Logical Process Model ............................................... 5
Figure 3 Module 0 System ......................................................... 9
Figure 4 Module 1 Image Capture ............................................. 10
Figure 5 Module 1.1 Initialize ................................................ 11
Figure 6 Module 1.2 Select Standard Setup ............................... 13
Figure 7 Module 1.3 Create Standard Setup ............................... 14
Figure 8 Module 1.4 Get Image ................................................ 15
Figure 9 Module 1.5 Manipulate Image ...................................... 16
Figure 10 Module 1.6 Save Image ............................................. 17
Figure 11 Module 1.7 End ....................................................... 20
Figure 12 Module 2 Image Retrieve ........................................... 21
Figure 13 Module 2.1 Version Control ....................................... 22
Figure 14 Module 2.2 Initialize ............................................... 24
Figure 15 Module 2.3 Image Query ........................................... 26
Figure 16 Module 2.4 End ........................................................ 29
1.0 INTRODUCTION

1.1 Purpose

This System Design Description (SDD) documents the detail design of the Photo Audio/Visual Management System (PAMS) Photo Image Retrieval Prototype (PPIRP) subsystem. This SDD shows how the software is structured to satisfy the requirements identified in the PAMS PHOTO IMAGE PROTOTYPE Requirements Document. It is a description of the software structure, software components, interfaces and data that make up the PPIRP subsystem.

1.2 Scope

The design descriptions contained within this document describe in detail the software product which allows the capture and retrieval of digital images from photograph sources. This system is a subsystem of the PAMS and uses the PAMS photo database. No external interfaces with other systems are required.

1.3 Overview

The PPIRP subsystem provides the means of capturing low resolution digital images from Photography’s negative files and associating the digital images with a record in the PAMS photo database. The subsystem also provides HAN users with access to both the PAMS photo information and the digital images to assist the user in locating and identifying specific photographs. After identifying desired photographs, users will request photo prints or high resolution digital images directly from Photography.

Images will be captured from new photographs and those existing photographs that have reprint requests. Photography expects to capture images of approximately 10,000 photographs per year.

The PPIRP subsystem is composed of two basic processes - capturing the images and retrieving the images. Other administrative processes are required to support these basic processes.

Capturing of images is done at a specially configured work station(s) attached to HAN. This work station is referred to as the Capture Station and consists of an IBM compatible (486 or better) personal computer with a Targa+16/32 image capture board. The capture board is attached to an appropriate input camera.

The Microsoft Windows (3.1 or better) Graphical User Interface (GUI) is the target user interface for the capture station.

Captured images are stored on a file server in one or more resolutions and compressions. The PAMS database relates the image file names with the photo identifier (Photo ID and Photo Dash Number). A single PAMS photo record may have any of low, medium or high resolution associated images. The expected implementation will have two images for those photographs that have gone through the capture process. The two images are a medium resolution full color image stored in compressed format and a thumbnail sized low resolution image stored uncompressed.
Retrieval of images is done at HAN workstations which have Microsoft Windows and color monitors. Although any windows compatible monitor can be used to display images, color monitors and video boards capable of displaying at least 16-bit color should be used.

1.4 Definitions

Form - In the Microsoft Visual Basic context, a form is a screen window which serves as an interface to an application.

Multiple-Document Interface - A Microsoft Windows term for a master form which allows other child forms to be loaded within the master container form. Each child form can have multiple instances (this application does not use multiple instances).

Targa - Video capture board from TrueVision.

Thumbnail - A low resolution version of a digital image used for quick identification of the image.

ACRONYMS

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HAN</td>
<td>Hanford Area Network</td>
</tr>
<tr>
<td>MDI</td>
<td>Multiple-Document Interface</td>
</tr>
<tr>
<td>PAMS</td>
<td>Photo Audio/Visual Management System</td>
</tr>
<tr>
<td>PPIRP</td>
<td>PAMS Photo Image Retrieval Prototype subsystem</td>
</tr>
<tr>
<td>SDD</td>
<td>System Design Description</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
</tbody>
</table>
2.0 DECOMPOSITION DESCRIPTION

2.1 Module Decomposition

2.1.1 Logical Data Model

![Logical Data Model Diagram]

**Figure 1** Logical Data Model

**Logical Entities**

**PHOTOGRAPH**

Description: These are physical photographs and are primarily negatives and slides. Occasionally, photo prints and stills from videos will be captured.

Data Elements: Each photograph is identified by a Photo ID and Dash Number.

**PAMS PHOTO RECORD**

Description: This record is defined by the PAMS system. A record provides information about a particular photograph.

Data Elements: Section 5.2.2 contains the field names and sizes for this file. See PAMS system documentation for a listing of the data elements with descriptions. For processing efficiency, three new fields were added to the basic PAMS record for this project - IMAGE_LOW, IMAGE_MED and IMAGE_HGH. These fields contain the DOS file name for the images created by the photo capture. IMAGE_LOW refers to the thumbnail image and IMAGE_MED refers to the captured image. The field IMAGE_HGH was created for future expansion and is not used.
PHOTO IMAGE

Description: This is a DOS file that represents a digitized image of a photograph.

Data Elements: None. The filename extension usually reflects the file format (example, BMP is the standard extension for a Windows Bitmapped file).

IMAGE CHARACTERISTICS

Description: These records contain the essential characteristics of the digitized photo image files. Each file has one such record. Essentially, the record maintains the photograph identification, the location of the image file and the format of the image file. The record is created when the photograph is digitized.

Data Elements: DOS File Name, Photo ID, Dash Number, Object Type, Storage Format, File Storage Location (DOS Volume ID and Path), and Image Resolution (Horizontal Resolution, Vertical Resolution, Color Depth). Data element descriptions are shown in section 5.2.1.
2.1.2 Logical Process Model

Figure 2 Logical Process Model
2.2 Concurrent Process Decomposition

2.3 Data Entity Decomposition
3.0 DEPENDENCY DESCRIPTION

3.1 Intermodule Dependencies

None identified at this time.

3.2 Interprocess Dependencies

The only interprocess dependencies are that the Capture Process and PAMS must both populate the databases before the Retrieve process has any benefit.

3.3 Data Dependencies

None have been identified. The Capture process will update the PAMS Photo File record only if it exists. This subsystem will not build PAMS Photo File Records.

Photo records may exist without Image records and vice versa. See detail design module 3.2 (List of Images With Incomplete Photo Data) in section 5 for the administrative task which reconciles the two files.

3.4 Conversion Dependencies

There is no data conversion requirement for this project.
4.0 INTERFACE DESCRIPTION

4.1 Module Interface

The two primary modules of this subsystem are Capture and Retrieve. No direct interface, other than the data repositories, exists for the two modules. Submodule interfaces are described in section 5 (detailed design).

4.2 Process Interface

No direct process interface exists for the two primary modules. See section 5 for submodule interfaces.
5.0 DETAILED DESCRIPTION

5.1 Module Detail Design

Figure 3 Module 0 System
M1.0 IMAGE CAPTURE.

The IMAGE CAPTURE process provides the means of controlling the capture process and creating and storing image files.

Figure 4 Module 1 Image Capture
M1.1 Initialize.

![Diagram of Module 1.1 Initialize](image)

**Figure 5 Module 1.1 Initialize**

M1.1.1 Get Custom Configuration.

The START form is loaded and displayed.

A check is made to ensure that the user does not already have a copy of the program running.

The OPEN DATABASE form is displayed and waits for the user to accept or modify the logon parameters. The connect to Server (M1.1.2) and connect to Database processes (M1.1.3) are invoked from this form. At the end of the connections the OPEN DATABASE form is unloaded. The OPEN DATABASE form also allows the user to bypass connecting, in this case the program continues but no updates are allowed (this feature is useful for testing and training).

The user's custom configuration file is read into variables. The custom configuration for the user is maintained in the "application.INI" file within the application directory. The file is a standard windows ".ini" file and is accessed by calls to windows API's (write profile and get profile). Some processes within this system may update the ".ini" file immediately. The ".ini" file is usually updated when the user exits.

User definable buttons are initialized. These are the buttons that allow the user to quickly change capture parameter setups.
The MULTIPLE-DOCUMENT INTERFACE (MDI) form is loaded. The MDI form is the master form for the capture process. Individual Child forms are loaded into the working area of the MDI form. Although, the capture application does not use multiple instances of Child forms, the MDI concept presents a common menu, button bar and a constrained work area on the screen.

The load procedure of the MDI form performs the following:

- Initialize the Targa+ capture board.
- Load and show the IMAGE PROCESSING form (frmIP).
- Load and show the DATA form.
- Load and show the THUMBNAIL form.

The START form is unloaded.

M1.1.2 Attach To Image Server.

The batch file "SRVRCON.BAT" is invoked to connect to the Image Server. After a successful connection, the drive letter is returned in file "drive.tmp" within the application's directory.

M1.1.3 Attach To PAMS Database.

A connection is made to the PAMS database using Microsoft's Open DataBase Connectivity (ODBC). Global variables are used to save information about the connection.
M1.2 Select Standard Setup.

This process is initiated when the user selects one of the standard user definable buttons. Each such button is associated with a capture device, a capture device setup and a screen setup (see M1.3 Create Standard Setup).

The setup values are looked up in the UserButton array and the capture device values are sent to the appropriate device. From session to session, the button values are kept in the "application.ini" file.

The screen setup values control the size and position of the image and cropping display regions on the IMAGE PROCESSING form but are not sent to the screen at this time. Instead, the regions are resized each time a capture is initiated. This allows the user to resize the regions on a case by case basis.

![Diagram](image)

Figure 6 Module 1.2 Select Standard Setup
M1.3 Create Standard Setup

The BUTTON SETUP form is loaded and displayed. This form allows the user to examine and modify the standard setup parameters for each of the user definable buttons (currently there are six). The setups include the following: caption; image region size; crop size; input device; input device setup id and preprocessing options (currently only convert to negative).

When the user signals Done, the form is unloaded.

Figure 7 Module 1.3 Create Standard Setup
M1.4 Get Image.

The IMAGE PROCESSING (IP) form's image region and crop region are sized and positioned to the current setup values.

An image is captured by the current input device and loaded into control TheImage of the IP form. A separate routine will be written for each such input device. Currently only the TARGA+ capture board is supported.

Preprocessing of the image is performed as indicated by the current setup.

![Diagram](image_url)
M1.5 Manipulate Image

Numerous Image Processing routines exist to modify the appearance of the image on the IP form. All processing routines are available to the user from the standard IMAGE portion of the main menu. The most commonly used image manipulations, such as rotate and flip, are callable by buttons on the MDI form.

Processing routines currently implemented include the following:

- Mirror Horizontal
- Mirror Vertical
- Rotate Right
- Rotate Left
- Negate
- Brightness/Contrast Adjustment
- Sharpen
- Blur
- Soften
- Convert To Color/Grayscale at specific resolutions.

Generally, each Image Processing option is written as a separate routine. An UNDO option is available to reverse the last image processing routine.

![Figure 9 Module 1.5 Manipulate Image](image-url)
M1.6 Save Image

![Diagram of Module 1.6 Save Image](image)

**Figure 10 Module 1.6 Save Image**

M1.6.1 Input Photo Identifiers.

Currently, the only Photo File information collected by this application are PHOTO ID and CAPTURED DASH NO. These are input by the user onto the DATA form. Other Photo File information is entered into the PAMS application.

M1.6.2 Edit Photo Identifiers.

Edits of the Photo Identifiers are performed as the first step to saving an Image to file. Both the PHOTO ID and CAPTURED DASH NO are required fields.

If records exist in the PAMS IMAGE FILE with the same identifiers, the user is notified and the Save Image process is canceled.

M1.6.3 Create Thumbnail Image.

A thumbnail sized image is created from the cropped portion of the current image on the IP form. Standard sizing and color depth for the thumbnail is based upon values stored in the "application.ini" file. The thumbnail image is displayed on the THUMBNAIL form.
M1.6.4 Build File Id’s for Image & Thumbnail

The PAMS COUNTERS FILE is queried to obtain the last used image number (CF_LAST_USED_VALUE where CF_COUNTER_NAME = IMAGE_NO). This value is incremented by one and becomes the current image number. The PAMS COUNTERS FILE is updated to reflect the change.

Two standard DOS file names are built as follows: (1) the full image file name is "1Mnnnnnn.ext" and (2) the thumbnail image file name is "THnnnnnn.ext". "nnnnnn" is the next file number and "ext" is the appropriate extension for the type of file being saved ("JPG" for the full image and "BMP" for the thumbnail image to reflect JPEG compressed and Bit Mapped, respectively).

M1.6.5 Save Image Files.

Each of the images (thumbnail image and cropped portion of the full image) is processed to format the image in the appropriate save format. Currently, the formats and required parameters, such as compression ratios, are hard coded. After processing, the images are written to the appropriate directory on the image file server.

Limiting the number of entries in a DOS subdirectory provides quicker response times for retrievals. The number portion of the name of the image file will be used to determine the initial storage subdirectory with the intent of limiting each subdirectory to approximately 200 file entries. The image file names have the format "1Mnnnnnn.JPG" and "THnnnnnn.BMP" where "nnnnnn" is a sequentially assigned number (see M1.6.4). The DOS subdirectory structure will be as below:

```
PHOTOIMG  IF00  IF0000
          IF0001
          IF0002
          ...
          IF0099
  IF01  IF0100
          IF0101
          IF0102
          ...
          IF0199
  ...
  IF99  IF9900
          ...
          IF9999
```

At this time no archiving of files is planned. However, to allow for an archive process and possible restructuring of the physical storage, the complete file location for each file is maintained in the PAMS_IMAGE_FILE table. All image retrieval requests must use the table to determine the file location and not depend upon the initial file location algorithm.
M1.6.6 Update PAMS Photo Data.

If a PAMS PHOTO FILE record exists for the entered PHOTO ID and CAPTURED DASH NO, the fields IMAGE_LOW and IMAGE_MED are updated with the file names of the thumbnail and full image, respectively. The Administrative process will list images that do not have comparable PAMS PHOTO FILE records.

A record is created in the PAMS IMAGE FILE for each of the thumbnail image file and full image file. The record contains the file number, photo identifier and complete information about the image file location and characteristics of the image.
M1.7 End

The END process performs the following functions:

Save the user configuration information in the "application.INI" file.

Close any open databases.

Disconnect from the Image Server (if attached).

Figure 11 Module 1.7 End
M2.0 IMAGE RETRIEVE

Image retrieval allows users with standard HAN workstations (with Windows software) to locate and view Photo data and images by querying the PAMS database.

Figure 12 Module 2 Image Retrieve
M2.1 Version Control.

The Image Retrieve process is designed to work as a client/server application. Client programs reside on the users workstation and access data in the PAMS database server and the Image File server. To ensure that only the latest version of the Image Retrieve is used, a process is in place to detect the user's program version and, if it is not current, download the latest version of the necessary programs to the user's workstation.

![Figure 13 Module 2.1 Version Control](image)

M2.1.1 Attach to Program Server.

The START form is loaded and displayed.

A check is made to ensure that the user does not already have a copy of the program running.

The OPEN DATABASE form is displayed and waits for the user to accept or modify the logon parameters. The batch file "SRVRCON.BAT" is invoked to connect to the Image Server. After a successful connection, the drive letter is returned in file "drive.tmp" within the application's directory.
M2.1.2 Check User's Version Against Latest Version.

The latest program version identification is read from the first line of the file "VERSION.DAT" on the program server. This is compared to the same line in the user's "VERSION.DAT" file. If the user's version differs from the latest version, the Download Process (M2.1.3) is invoked.

M2.1.3 Download Latest Version.

This process will download the latest version of programs to the user's workstation from the program server. Depending upon the user's current version, different programs may require downloading (in case the user missed one or more version updates). To handle this situation, the program server file "VERSION.DAT" contains the names of batch files to invoke for each possible user version (actually a limited number of user versions are supported - very older versions will require a complete reinstall).

Based upon the user's current version, the appropriate download batch file on the program server is initiated to transfer the program files to the user workstation. A text file explaining the process and describing any special instructions is also transferred to the user workstation and is displayed to the user.

M2.1.4 Disconnect from Program Server.

A disconnect from the Program Server is performed.
M2.2 Initialize

![Diagram]

**Figure 14 Module 2.2 Initialize**

M2.2.1 Get User's Custom Configuration

The user's custom configuration file is read into variables. The custom configuration for the user is maintained in the "application.INI" file within the application directory. The file is a standard Windows '.ini' file and is accessed by calls to Windows Application Program Interfaces (APIs). The "write profile" and "read profile" APIs are used. Some processes within this system may update the '.ini' file immediately. The '.ini' file is usually updated when the user exits.

The MULTIPLE-DOCUMENT INTERFACE (MDI) form is loaded. The MDI form is the master form for the retrieve process. Individual Child forms are loaded into the working area of the MDI form. Although, the Image Retrieve application does not use multiple instances of Child forms, the MDI concept presents a common menu, button bar and a constrained work area on the screen.

The load procedure of the MDI form performs the following:

Load and show the QUERY BUILDER form.

The START form is unloaded.

M2.2.2 Attach to Image Server.
The batch file "SRVRCON.BAT" is invoked to connect to the Image Server. After a successful connection, the drive letter is returned in file "drive.tmp" within the application's directory.

M2.2.3 Attach to PAMS Database

A connection is made to the PAMS database using Microsoft’s Open DataBase Connectivity (ODBC). Global variables are used to save information about the connection.

M2.2.4 Get Field Metadata.

Information about each of the fields in the PAMS Photo File is maintained in the table PHOTO_FILE_FIELDS in the PAMS database. This table contains information such as the field’s database name, common name, type, user viewable flag, user query flag, etc. A global array is constructed from the information in the PHOTO_FILE_FIELDS table.
M2.3 Image Query.

**Figure 15 Module 2.3 Image Query**

M2.3.1 User Build Query.

This process allows the user to construct a query against the PAMS Photo File. The user determines the fields they want to see, the order of the fields and selection criteria.

A Grid Control on the Query form is used to construct the query. Each column of the grid control holds query information about a field. Columns can be added, modified or deleted as the user desires. To add (or modify) a column, the user selects a column and a field name. Based upon the type of field, the program then presents the user with a list of allowable selection operators for the field (example, date fields can select "equal", "less than", "greater than" or "between"). If the user chooses a selection operator, the selection criteria is then entered. When entering selection criteria, the keywords "or" and "and" are detected by the program and highlighted to indicate to the user that a connection was entered. When the user selects OK, the selection criteria is edited for format.
If selection criteria is entered in more than one column, the program constructs an "and" connection between the columns in the resulting SQL statements.

The user can save queries using a common "file save" dialog. Saved queries will have the extension ".QRY" and are kept in the application directory. The query file contains one line for each column in the query grid (see section 5.2.6 for the specific format of the file). When a query is retrieved, any fields which no longer match the existing fields in the PHOTO_FILE_FIELDS table (see process M2.2.1) are ignored.

M2.3.2 Execute Query.

After the user completes the query builder grid, a simple Select SQL command is constructed from the data in the grid. Each portion (SELECT, FROM, WHERE and ORDER BY) of the SQL command is generated independently and then combined into a single statement. Because the user only sees the common name (caption) for the data fields, each field caption is translated to the appropriate database field name.

The fields "KEYWORD" and "CHARACTER" are actually multivalued fields in the PAMS_PHOTO_FILE. Therefore, selections in one of these fields are "OR" connected in the "where clause" to all the fields of the same type.

Additional fields (image file names) are appended to the select portion of the SQL statement to identify those records with images available for viewing.

A SQL select command has the possibility of returning a large number of records from the database. To prevent this situation, a "row count" SQL command is built using the WHERE clause and sent to the PAMS Database. If the row count exceeds the default maximum, the user is notified. The user can elect to redo the query or receive the records. An "absolute maximum" that cannot be overridden is a constant in the system.

The SQL command is sent to the PAMS Database for action.

M2.3.3 View Query Result.

A set of row entries is returned by the SQL action initiated in the Execute Query process (M2.3.2). The row entries are loaded into a Visual Basic Snapshot (dynaset). From the snapshot, a table (grid control) is generated in the Query Result form to display the text results.

Entries within the multivalued fields, "KEYWORD" and "CHARACTER" are combined into a single entry for the type and separated with a semicolon. Example, KEYWORD1 of "this" and KEYWORD2 of "that" are displayed in the KEYWORD field as "this; that".

If the Image field (IMAGE_MED) is not empty for a row, then an Image indicator is placed in column one of the grid to alert the user to the fact that a viewable image is on file for the record.
For any row in the result, the user can ask for additional information for that photo by highlighting the row and then selecting the "zoom" button. This action causes the ZOOM form to load and show. The ZOOM form sends a simple SQL SELECT command to the PAMS Database which returns a single record containing all the data for the photo. The data is displayed in a list on the ZOOM form. After viewing, the user closes the form.

The query result and/or viewable image (see process M2.3.4) are intended to provide the user with enough information to select photographs of interest. It is intended that the user will utilize this selection to complete a Photo Request form.

M2.3.4 View Images.

The query result (process M2.3.3) displays photo text information to the user. To view images, the user first selects rows from the query result grid for those photographs of interest. Provisions are made to toggle individual row selections on and off and to select-all and select-none.

After marking the rows for selection, the user enters the command to view images. Thumbnail images are retrieved from the Image Server for all the marked rows in the Query Result grid. Using the IMAGE_LOW field as a key (actually the file name for the image), the PAMS IMAGE FILE is queried to obtain the file location and image description. The images are retrieved six (currently) at a time and displayed on the THUMB form. Each image is displayed with a caption identifying the photograph. Provisions on the form allow for scrolling through the images.

An individual image can be selected by the user for viewing at a higher resolution on the IMAGE form. When selected, the IMAGE_MED field is used as a key to the PAMS IMAGE FILE to obtain the image information. In the current implementation, the medium resolution images are stored in compressed format and will be converted to an uncompressed viewing format.

M2.3.5 Local Save Images.

This process has not been designed at this time.
M2.4 End.

M2.4.1 Clean Up Loose Ends
M2.4.2 Disconnect from Image Server
M2.4.3 Disconnect from PAMS Database
M2.4.4 Save User’s Custom Configuration
M3.0 ADMINISTRATION

M3.1 User Access Control (PAMS)

This functionality is provided by the PAMS system.

M3.2 List of Images With Incomplete Photo Data

This process simply compares the PAMS_IMAGE_FILE to the PAMS_PHOTO_FILE and lists a record for each IMAGE_FILE entry that does not have a corresponding PHOTO_FILE entry. The process is expected to be user initiated on a scheduled basis - probably weekly.
5.2 Data Detail Design

5.2.1 Table: PAMS IMAGE FILE

This table is in the PAMS database and contains one record for each image file being maintained by the PAMS system. Key to the file is IF_FILE_NAME. The fields IF_PHOTO_ID and IF_CAPTURED_DASH_NO are used to join to the PAMS_PHOTO_FILE table with fields PF_PHOTO_ID and PF_CAPTURED_DASH_NO respectively. It is not required that the PAMS_PHOTO_FILE have corresponding entries. Also, the PAMS_PHOTO_FILE will have many records that do not have corresponding records in the PAMS_IMAGE_FILE.

The IF_FILE_NAME field is also carried in the PAMS_PHOTO_FILE as PF_IMAGE_LOW or PF_IMAGE_MED depending upon the resolution of the image. This redundancy is to improve the processing speed of the image retrieval.

<table>
<thead>
<tr>
<th>FIELD</th>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF_FILE_NAME</td>
<td>char(12)</td>
<td>Name of the file where the image is stored. Established by the &quot;Image Capture&quot; program. Format is xxnnnnnn.eee xx = &quot;IM&quot; image (medium res.) &quot;TH&quot;thumbnail (low res) nnnnnn is a unique counter maintained by the capture program eee is the extension for file type.</td>
</tr>
<tr>
<td>IF_PHOTO_ID</td>
<td>char(8)</td>
<td>PAMS Photo ID</td>
</tr>
<tr>
<td>IF_CAPTURED_DASH_NO</td>
<td>char(4)</td>
<td>PAMS Dash ID</td>
</tr>
<tr>
<td>IF_OBJ_TYPE</td>
<td>char(3)</td>
<td>Code for resolution</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOW = low resolution image</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MED = medium resolution image</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HGH = high resolution image</td>
</tr>
<tr>
<td>IF_STORG_FMT_CD</td>
<td>char(3)</td>
<td>Code for storage format - the usual DOS file extension for each format is shown within the parenthesis:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JPG  JFIF (JPG) Joint Photographic Experts Group format. JFIF is JPEG File Interchange format as described by C-Cube Microsystem using the JPEG compression standard.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BMP  Bitmapped file.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TIF  Tagged Image File Format.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TF6  (JPG) A TIFF/JPEG extension of the Aldus TIFF format.</td>
</tr>
<tr>
<td>Field</td>
<td>Data Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IF_VOLUME_ID</td>
<td>varchar(11) NULL</td>
<td>Volume Identifier for storage disk containing the object file.</td>
</tr>
<tr>
<td>IF_FILE_PATH_TXT</td>
<td>varchar(52)</td>
<td>Full path to directory containing object file. Example: \mydir\subdir\</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOS limits the full path to 66 characters including the drive, colon and file name.</td>
</tr>
<tr>
<td>IF_RESOLUTION_HORIZ</td>
<td>smallint</td>
<td>Horizontal resolution in pixels.</td>
</tr>
<tr>
<td>IF_RESOLUTION_VERT</td>
<td>smallint</td>
<td>Vertical resolution in pixels.</td>
</tr>
</tbody>
</table>
| IF_COLOR_CD         | char(2)    | 1B = Black/White  
  4C = 4-bit color (16 Colors)  
  4G = 4-bit gray scale  
  8C = 8-bit color (256 colors)  
  8G = 8-bit gray scale  
  12 = 12-bit (4,096 colors)  
  16 = 16-bit (65,536 colors)  
  24 = 24-bit (16.7 million colors)  
  36 = 36-bit Photo CD quality |
5.2.2 Table: PAMS PHOTO FILE

This table is in the PAMS Database and contains one record for each photo in the system. Structure of the table is documented in the PAMS system documentation. The field names and sizes are shown below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF_PHOTO_ID</td>
<td>CHAR(8),</td>
</tr>
<tr>
<td>PF_CAPTURED_DASH_NO</td>
<td>CHAR(4),</td>
</tr>
<tr>
<td>PF_UPDT_BY</td>
<td>CHAR(20),</td>
</tr>
<tr>
<td>PF_UPDT_DTE</td>
<td>DATETIME,</td>
</tr>
<tr>
<td>PFRELATED_DASH_NO</td>
<td>CHAR(10) NULL,</td>
</tr>
<tr>
<td>PF_NEG_CNT</td>
<td>SMALLINT NULL,</td>
</tr>
<tr>
<td>PF_LEAD_NAME</td>
<td>CHAR(25) NULL,</td>
</tr>
<tr>
<td>PF_DTE_WRK_PRFMD</td>
<td>DATETIME NULL,</td>
</tr>
<tr>
<td>PF_CUST_NAME</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_CONTR_CODE</td>
<td>CHAR(5) NULL,</td>
</tr>
<tr>
<td>PF_CONTACT</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_BLDG_OFFCL_NO</td>
<td>CHAR(11) NULL,</td>
</tr>
<tr>
<td>PF_BLDG_NAME</td>
<td>VARCHAR(50) NULL,</td>
</tr>
<tr>
<td>PF_FAC_TYPE_CD</td>
<td>CHAR(2) NULL,</td>
</tr>
<tr>
<td>PF_FAC_TYPE_DESC</td>
<td>CHAR(30) NULL,</td>
</tr>
<tr>
<td>PF_AREA</td>
<td>CHAR(6) NULL,</td>
</tr>
<tr>
<td>PF_CITY</td>
<td>CHAR(25) NULL,</td>
</tr>
<tr>
<td>PF_STATE</td>
<td>CHAR(2) NULL,</td>
</tr>
<tr>
<td>PF_SUBJ_TYPE</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_STYLE</td>
<td>CHAR(15) NULL,</td>
</tr>
<tr>
<td>PF_TECHNIQUE</td>
<td>CHAR(15) NULL,</td>
</tr>
<tr>
<td>PF_SITE_FUNC</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_KYWDO1</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_KYWDO2</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_KYWDO3</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_KYWDO4</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_KYWDO5</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_KYWDO6</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_CHAR_NAME_01</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_CHAR_NAME_02</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_CHAR_NAME_03</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_CHAR_NAME_04</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_CHAR_NAME_05</td>
<td>CHAR(20) NULL,</td>
</tr>
<tr>
<td>PF_FIRST_USED_FOR</td>
<td>CHAR(15) NULL,</td>
</tr>
<tr>
<td>PF_IMPACT_FLAG</td>
<td>CHAR(1) NULL,</td>
</tr>
<tr>
<td>PF_RIDS</td>
<td>CHAR(2) NULL,</td>
</tr>
<tr>
<td>PF_STORAGE_BOX</td>
<td>CHAR(7) NULL,</td>
</tr>
<tr>
<td>PF_DTE_SHIPPED</td>
<td>DATETIME NULL,</td>
</tr>
<tr>
<td>PF_PROJ_NO</td>
<td>CHAR(10) NULL,</td>
</tr>
<tr>
<td>PF_PROJ_NAME</td>
<td>CHAR(10) NULL,</td>
</tr>
</tbody>
</table>
5.2.3 Table: PHOTO FILE FIELDS

This table is in the PAMS database. It contains one record per field in the PAMS_PHOTO_FILE table and is used by the Photo Image Retrieval process.

Key to the table is NAME. CAPTION must also be unique for the table.

<table>
<thead>
<tr>
<th>FIELD</th>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFF_NAME</td>
<td>CHAR(30)</td>
<td>The actual name of the field as it appears in the PAMS_PHOTO_FILE.</td>
</tr>
<tr>
<td>PFF_CAPTION</td>
<td>CHAR(30)</td>
<td>The common name of the field. Used as a caption on queries.</td>
</tr>
<tr>
<td>PFF_USER_SEQUENCE</td>
<td>SMALLINT</td>
<td>A sequence number to reflect the order of the field in presentations to the user. Need not be consecutive. The fields are shown in order of this sequence number with low numbers first.</td>
</tr>
<tr>
<td>PFF_USER_SEARCH</td>
<td>CHAR(1)</td>
<td>&quot;Y&quot; if user is allowed to enter search criteria for this field. &quot;N&quot;, if not.</td>
</tr>
<tr>
<td>PFF_DISPLAY_WIDTH</td>
<td>SMALLINT</td>
<td>Width of a column (in characters) usually reserved to display the field. If set to zero, then the field is to be hidden from the user.</td>
</tr>
<tr>
<td>PFF_SELECT_TYPE</td>
<td>CHAR(1)</td>
<td>Type of selection operator allowed for this field - usually reflects the general type of the field. Values include: T for text, D for date, N for numeric, B for boolean.</td>
</tr>
</tbody>
</table>
5.2.4 APPLICATION.INI file

This file is a standard windows initialize file (".ini"). However, the file is kept in the same directory as the application for two reasons: (1) the removal of an outdated application is simplified and (2) multiple installations of the application can be made on the same machine.

Structure of the file is as follows:

[section1]
variable1 = value
variable2 = value
variable3 = value
...
[section2]
...

Windows API's are used to read the file and to write to the file. Common ASCII editors can also be used to modify a file.

The section names and variable names are application specific. Section names are unique within the file and variable names are unique within the section.

5.2.4.1 Capture Initialization File

The current "*.ini" file for the capture application is "PHOTOIN.INI" and contains the following:

[DBServer]
Server = it
DataBase = IRMTEST
UserName = mconner
Password =
OpenOnStartup = No
QueryTimeout = 5
DataType = ODBC
LoginTimeout = 20

[Setup]
TempArea = d:\

[ImageServer]
ServerName = \bcs29\ml_conner 
ImagePath = \images\

[ThumbNail]
 ThumbPixelWide = 100
ThumbPixelHigh = 100
ThumbBitDepth = 24
[Capture]
CapturePixelWide=512
CapturePixelHigh=486

[Targa]
Interlaced=1
LiveSource=0
Genlock=1
DisplayMode=3

[UserButton1]
Caption=6x6
WindowWide=6405
WindowHigh=4980
CropWide=4245
CropHigh=4350
CropTop=60
CropLeft=1215
ConvertNegative=0
TargaSetup=10

[UserButton2]
Caption=35mm
WindowWide=5895
WindowHigh=4950
CropWide=1455
CropHigh=2265
CropTop=645
CropLeft=1695
ConvertNegative=0
TargaSetup=20

[UserButton3]
Caption=three
WindowWide=6405
WindowHigh=4980
CropWide=3180
CropHigh=3150
CropTop=630
CropLeft=2010
ConvertNegative=0
TargaSetup=10

[UserButton4]
Caption=xxx
WindowWide=6390
WindowHigh=4935
CropWide = 4185
CropHigh = 4350
CropTop = 30
CropLeft = 960
ConvertNegative = 0
TargaSetup = 10

[UserButton5]
Caption = 640x480
WindowWide = 6405
WindowHigh = 5325
CropWide = 5745
CropHigh = 4605
CropTop = 30
CropLeft = 330
ConvertNegative = 0
TargaSetup = 30

[UserButton6]
Caption = 512x486
WindowWide = 6390
WindowHigh = 5100
CropWide = 3600
CropHigh = 3675
CropTop = 405
CropLeft = 1635
ConvertNegative = 0
TargaSetup = 20

5.2.4.2 Retrieval Initialization File

The current ".ini" file for the image retrieval application is "PICFIND.INI" and contains the following:

[DBServer]
Server = wc25931
DataBase = pamsdev
UserName = pamsuser
Password = ****
OpenOnStartup = No
QueryTimeout = 5
DataType = ODBC
LoginTimeout = 20

[Setup]

[ImageServer]
ServerName=\bscs29\ml_conne ****
ImagePath=\images\n
[ThumbNail]
ThumbPixelWide=100
ThumbPixelHigh=100
ThumbBitDepth=24

[QUERY]
QueryPath=C:\PICFIND\TEST\QUERY
5.2.5 TARGASET.DAT file

This file contains the user’s desired setups for the TrueVision Targa+ capture board. Each record represents a single setup and contains the register contents required for the setup.

Key to the file is the TSEQUENCE field.

Type TargaSave

tsequence As String * 2 'sequence. A user defined sequence number to uniquely identify this setup.

tdescribe As String * 40 'caption. User description of the setup.

tRegs As tPlusRegs 'registers. Contents of the 130 registers. (see type tPlusRegs below)

cr As String * 2 'carriage return, line feed.

End Type

Type tPlusRegs

    tReg(1 To 130) As Integer 'Array for contents of the Targa Registers as described in the Targa+ Technical Guide.

End Type
5.2.6 VERSION.DAT file

This file is used to control the version of the Retrieve programs on the client workstations. Part of the Retrieve program start process will compare the most recent version number to the client’s version number and initiate appropriate actions.

The file is formatted as a standard ASCII file.

Line 1:
- Column 1 to 10: Version id. Standard format is mm/dd/yy-n where mm/dd/yy is the version date and n is a version number for the day (in case more than one version is issued during a single day).
- Column 12 to 36: blank

Remaining Lines:
- Column 1 to 10: Version ID
- Column 12 to 23: Batch File Name to initiate
- Column 25 to 36: Name of the text file to display to the user. This file is a simple ASCII text file and will tell the user about the features of the new version and provide any information on special procedures the user must perform to complete the upgrade.

Sample: -------------------------------------
11/02/93-1
10/13/93-1 down0010.bat info0010.txt
10/01/93-2 down0010.bat info0010.txt
10/02/93-1 down0010.bat infostd.txt
09/15/93-1 down0009.bat info0009.txt
-------------------------------------
5.2.6 User Query Save Files

A photo retrieve user may elect to save their query structures. The user can establish their own file names - the normal extension for these files is ".QRY".

Structure of a query file is as follows:

Each record contains comma delimited fields. Character fields are enclosed in quotes. The file consists of a header record and, for each column (except 0), a detail record.

Header Record: Number of columns in grid, number of rows in grid, query format
Detail Record: Column number (zero is first column), one character string for each row

Example:

```
5,4,"QUERYSAVE-FORMAT1"
1,"Photo ID","","",""
2,"Dash#","","",""
3,"Impact Flg","Equal To","y",""
4,"Lead Name","","",""
```
5.3 General Design

5.3.1 Application

The Capture and Retrieve modules are written in Microsoft's Visual Basic (Professional Edition) for Windows. Media Architect's ImageKnife/VBX custom control for Visual Basic is used for all image manipulation functions.

In general, the application programs follow the Visual Basic event driven processing structure.

Both the Capture and Retrieve modules use a Multiple-Document Interface (MDI) form as the primary presentation form. Process dependent forms are generally loaded as child forms within the MDI form. Some forms which demand immediate action are loaded modal (user cannot switch from the form without completing the action) and as such cannot be child forms.

Forms will be documented as part of the development process.

5.3.2 Command Buttons and Menus

Menu commands will be documented as part of the development process.

Command buttons for the most commonly used menu items are displayed immediately below the menu section on the MDI form.

The Menu structure is imposed at the MDI form level and not the individual form level. Individual forms are responsible for enable/disable of menu items as appropriate. Because Visual Basic only allows form procedures to be called from within the same form, the child forms contain hidden command buttons for those procedures which are called from the menu or command buttons. This allows the MDI form menu item to start a child form procedure by setting the command button value to True.

All procedures applicable to more than one form, are made a global and, as such, can be called from any form.

5.3.3 Help Files

Standard Windows help files are constructed for each of the Capture and Retrieve modules. These files can be called in the usual Windows manner.

Because of the limited number of users, the Capture Help file will initially provide minimal assistance. A more easily modified printed user manual will serve as the primary source of user instructions.

On the other hand, the Retrieve Help file will be complete enough to serve as the user manual. Therefore, no user manual will be routinely published and distributed to HAN users. Additional "getting started" instructions will be made available to the users.
5.3.4 Program Setup

The Capture module is expected to reside on one or two workstations and does not require a standard setup procedure.

Because the Retrieve module will be made available to HAN users across the site, a standard Windows Setup Procedure will be developed. The setup program moves appropriate programs, data files, and dynamic link libraries to the user workstation and then creates a program group on the user station. The user is allowed to determine their own directory location for the Retrieve application.

The Setup programs are stored on the Retrieve Program File Server and are initiated from the Windows Program Manager.
6.0 REFERENCES

WHC CM3-10 - SOFTWARE PRACTICES

PAMS PHOTO IMAGE RETRIEVAL PROTOTYPE REQUIREMENTS DOCUMENT - 8/27/93