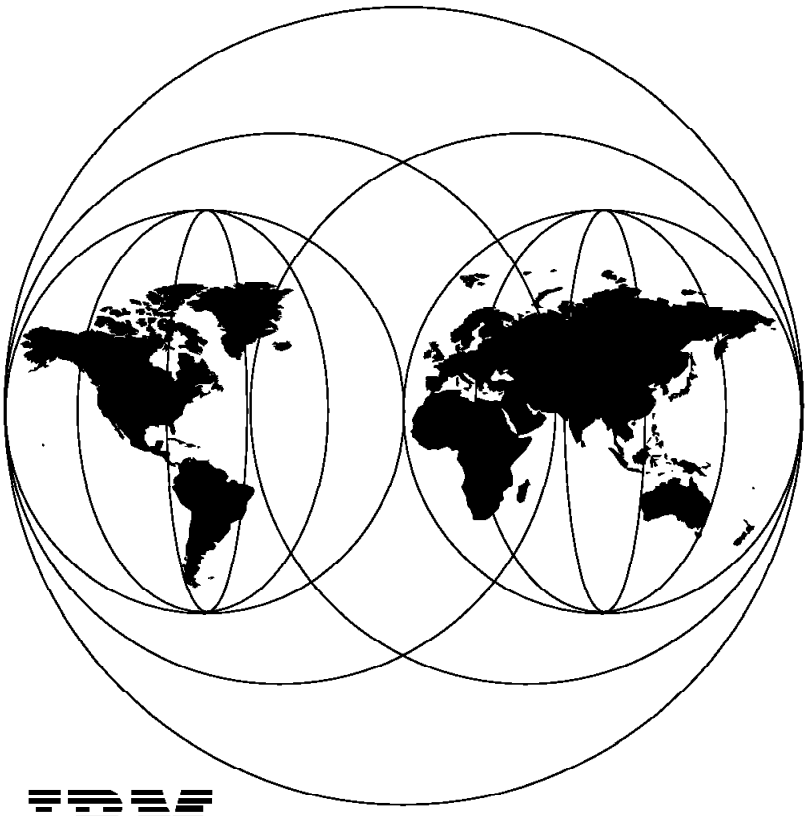
International Technical Support Organization

GG24-4488-00

VisualInfo MVS/ESA Installation and Customization Guide

June 1995



IBM

International Technical Support Organization
Dallas Center

International Technical Support Organization

GG24-4488-00

VisualInfo MVS/ESA Installation and Customization Guide

June 1995

Take Note!

Before using this information and the product it supports, be sure to read the general information under “Special Notices” on page xiii.

First Edition (June 1995)

This edition applies to Version 1 Release 0 of IBM ImagePlus VisualInfo Library and Object Servers for MVS/ESA, Program Numbers 5655-071 and 5655-072 for use with the MVS/ESA Operating System.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

An ITSO Technical Bulletin Evaluation Form for reader's feedback appears facing Chapter 1. If the form has been removed, comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept. 470 Building 40 Internal Zip 40-E4-203
1 E. Kirkwood Blvd
Roanoke, Texas 76299-0001

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Abstract

This document is unique in its detailed coverage of VisualInfo/MVS. It focuses on the preparation, installation, and configuration of VisualInfo/MVS, including a library server, object server, and OS/2 configuration server and client.

This document was written for technical people who install and support VisualInfo/MVS. Knowledge and experience on the prerequisite software is assumed (MVS, VTAM, DB2, CICS, OS/2, CM/2).

(140 pages)

Contents

Figures

Tables

Special Notices

This publication is intended to help systems programmers/installers/integrators to install VisualInfo/MVS. The information in this publication is not intended as the specification of any programming interfaces that are provided by VisualInfo/MVS or ImagePlus. See the PUBLICATIONS section of the IBM Programming Announcement for VisualInfo for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM′s product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM′s intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer′s ability to evaluate and integrate them into the customer′s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms, which are denoted by an asterisk (*) in this publication, are trademarks of the International Business Machines Corporation in the United States and/or other countries:

Advanced Peer-to-Peer Networking APPN
IBM

The following terms are trademarks of other companies:

Windows is a trademark of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

C-bus is a trademark of Corollary, Inc.

Other trademarks are trademarks of their respective companies.

Preface

This document is intended to give step-by-step guidance on installing VisualInfo/MVS. It contains preparation steps, installation and customizing of the servers and client, verification and problem determination hints, and descriptions of the JCL, tables and files used.

This document is intended for systems programmers, application programmers, LAN administrators, and other technical personnel who are installing and supporting VisualInfo/MVS.

How This Document is Organized

The document is organized as follows:

- Chapter 1, “Introduction”
- Chapter 2, “Preparation”

  This chapter describes the necessary preparation that must be done before installation can begin.
- Chapter 3, “Installing and Customizing the Library Server”

  This chapter describes the activities required to install the Library Server, the required tables for DB2, and the Folder Manager.
- Chapter 4, “Installing and Customizing the Object Server”

  This chapter provides information on the installation of the Object Server.
- Chapter 5, “OAM and SMS Customizing”

  This chapter provides the necessary steps for installation and customizing of OAM and SMS.
- Chapter 6, “Installing a VisualInfo Client on OS/2”

  This chapter describes installation of the Client Application.
- Chapter 7, “Installing a VisualInfo Configuration Server”

  This chapter provides the necessary details for installation of the Configuration Server.
- Chapter 8, “Installation Verification and Problem Determination”

  This chapter tells how to ensure that your system is configured correctly, and provides tips on fixing problems.
- Appendix A, “CICS Installation Jobs”

  This appendix provides more details on the CICS installation jobs, tables, and session definitions.
- Appendix B, “Optional OAM and Customization”

  This appendix provides more information on the optional steps that can be taken with the OAM and SMS customizing.
- Appendix C, “ Table Descriptions”

  This appendix gives a layout of the Library Server DB2 basic and Folder Manager tables.

- Appendix D, “Sample RDO Definitions”

  This appendix lists RDO definitions for C/370 and PL/I.

- Appendix E, “OS/2 Workstation Files”

  This appendix lists working OS/2 files such as CONFIG.SYS.

- Appendix F, “General Installation Jobs and Tips”

  This appendix lists jobs and tips for installing VisualInfo/MVS.

Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this document.

- *ImagePlus VisualInfo Installation Guide*, GC31-7729 (GK2T-1710-00)
- *ImagePlus VisualInfo Application Programming Reference, Volume 3: Common Data Structures and Database Tables*, SC31-7665
- *OAM Planning, Installation and Storage Administration Guide for Object Support*, SC26-4918
- *Using the Interactive Storage Management Facility*, SC26-4911
- *Storage Administration Reference for DFSMSdfp*, SC26-4920
- *CICS/ESA System Definition Guide*, SC33-0664
- *CICS/ESA Resource Definition (Macro)*, SC33-0667
- *DB2 Version 3 Administration Guide*, SC26-4888

International Technical Support Organization Publications

- *OAM/SMS Design Guidelines*, GG24-4143
- *A Simple Approach to VisualInfo*, GG24-4444

A complete list of International Technical Support Organization publications, with a brief description of each, are found in:

International Technical Support Organization Bibliography of Technical Redbooks, GG24-3070.

To get a catalog of ITSO technical publications (known as “Redbooks”), VNET users may type:

```
TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG
```

How to Order ITSO Technical Publications

IBM employees in the USA may order ITSO books and CD-ROMs using PUBORDER. Customers in the USA may order by calling 1-800-879-2755 or by faxing 1-800-284-4721. Visa and Master Cards are accepted. Outside the USA, customers should contact their local IBM office.

Customers may order hardcopy ITSO books individually or in customized sets, called GBOFs, which relate to specific functions of interest. IBM employees and customers may also order ITSO books in online format on CD-ROM collections, which contain books on a variety of products.

Acknowledgments

This project was designed and managed by:

Scott Chen
International Technical Support Organization, Dallas Center

The authors of this document are:

Jan M. Bester
IBM South Africa

Michael A. Ebbers
IBM Education and Training

James A. Tau
IBM Image Competency Center, USA

This publication is the result of a residency conducted at the International Technical Support Organization, Dallas Center.

Thanks to the following people for the invaluable advice and guidance provided in the production of this document:

David Rusnak
IBM VisualInfo Development

John DiClementi
IBM VisualInfo Development

Da-Sing Lu
IBM VisualInfo Development

Chapter 1. Introduction

1.1 The Product

VisualInfo is an ImagePlus client/server product from IBM that allows you to capture images and other files, store them, and retrieve them electronically. In combination with powerful storage and retrieval functions, its workflow capability allows you to manage the flow of paperwork information through your business.

VisualInfo offers server functions on MVS and OS/2 platforms, while clients are run on OS/2. A statement of direction for Windows clients and AIX servers is in place. API functions are supplied to allow you to create your own Windows client interface.

1.2 The Configuration

This documentation is designed to take you step-by-step through the installation of an MVS VisualInfo system, including an OS/2 client. Our configuration consists of an MVS library server, an MVS object server, an OS/2 configuration server, and an OS/2 client.

All components are connected using APPN. You could bypass the configuration server and use APPC to connect the servers and clients.

Our library server and object server reside on the same MVS system that is located in Poughkeepsie. Two locations, Bethesda and Dallas, were used for the client. Each location has a workstation that contains a configuration server and a client. In a production environment, a separate workstation is recommended for your configuration server for reliability and performance.

Chapter 2. Preparation

In order to ensure a successful installation, you should ensure that the environment for VisualInfo is created correctly. We are not detailing the installation of each of the required products, but covering only the areas that you should be aware of.

The VisualInfo documentation assumes that the installation and customizing of Library Server and Object Server are performed on CICS regions that are operational. This chapter describes steps to create the required data sets for each CICS region, and to enable the CICS support of each required program product.

The Appendixes contain some samples, which you may want to browse before you begin the installation.

2.1 Setting Up the Common CICS Data Sets

Our decision was to use a common CDS for the Library and Object Servers. If you plan to use your current CDS data sets on your system, you can skip this section.

The jobs DFHCOMDS, DFHCMACI, and DFHDEFDS were copied from CICS V3.3 sample JCL to VIUSR1.INSTALL.CNTL. The high-level qualifier of the data sets shared by each CICS regions is VIHOST.CICS330. The high-level qualifier of the data sets used by the Library Server CICS regions is VILIBSVR.CICS330. The high-level qualifier of the data sets used by the Object Server CICS regions is VIOBJSVR.CICS330.

Job DFHCONS was created to add CICS consoles as a group (BETHCONS) in the DFHCSD (CICS System Definition).

2.2 Enabling the CICS Support on the Required Program Products

The programs used in CICS have to be defined in the CICS program properties table (PPT). The CICS System Definition File Utility program is used to define these programs into a group.

For each VisualInfo CICS region, we are including the appropriate group into its group list. This is described further in a later chapter.

During the installation of the PL/I and C/370, be sure to follow instructions provided in the program directory, installation guide, and PSP buckets carefully. Most of the problems that cause VisualInfo programs to fail are due to improperly installed libraries.

Note: You *must* install the C/370 Specific library *after* the C/370 Common library is installed.

2.2.1 Enabling the DB2 Support for CICS

If your DB2 subsystem does not have the DB2 CICS Attach load modules link-edited, you have to use the DSNTIJSU from SDSNSAMP library to perform this task.

We modified the DSNTIJSU job to use our data sets and EXECKEY(CICS) was added due to APAR II06515. The modified JCL is found in A.1, “DB2 CICS Support (DSNTIJSU)” on page 81. The link-edited modules were stored in the VIHOST.DSN310.SDSNLOAD.

During the second step of this job, transactions and programs were defined in a group (DB2) which was added to a list called DB2LIST.

2.2.2 OS PL/I Library (5668-911)

The library should be installed as described in the program directory.

The member PLTCICS in the VIUSR1.INSTALL.TABLES contains the definitions for the program property table (PPT). We assembled it first and used RDO MIGRATE to create a group (PLIV2R3). See Appendix D, “Sample RDO Definitions” on page 111 for examples.

In addition, we also added programs IBMEPGDA and IBMEPMRA to the same group as follows:

```
DEFINE PROGRAM(IBMEPGDA) GROUP(PLIV2R3) LANGUAGE(ASSEMBLER)
       RELOAD(NO)   RESIDENT(NO) STATUS(ENABLED) CEDF(NO)
DEFINE PROGRAM(IBMEPMRA) GROUP(PLIV2R3) LANGUAGE(ASSEMBLER)
       RELOAD(NO)   RESIDENT(NO) STATUS(ENABLED) CEDF(NO)
```

2.2.3 C/370 Common Library (5688-082)

The common library referred to as Version 2 Release 3 Modification 0 (V2R3M0) by C/370 library V2R2M0 or OS PLI V2R3M0 is also known as Program Product 5688-082 Common Library V2R3M0. It is ordered as a feature of the C/370 Library V2R2M0.

The installation of the C/370 Common Library is performed by following the product program directory.

2.2.4 Enabling the C/370 Specific Library Support for CICS

The installation procedures for C/370 Specific Library support for CICS are found in the product program directory. After the Receive and Apply steps, follow instructions in Chapter 6.17, “ Installation of IBM C/370 Specific Library support for CICS ” to enable the CICS support for your CICS regions.

1. The sample job EDCCCSD was copied from the SEDCJCL1 into our VIUSR1.INSTALL.TABLES. It is the RDO definitions for the C/370 Specific Library entries for the CICS Processing Program Table (PPT). See Appendix D, “Sample RDO Definitions” on page 111 for examples. The CICS System Definition File Utility program is used to update the CSD. The group name EDC is created.
2. The Destination Control Table entries must be added to each CICS region. This step was performed when we built the DCT table for each of the CICS regions. See 3.2.1, “Customizing Required CICS Tables” on page 9, and 4.2.1, “Customizing Required CICS Tables” on page 20.

The DDNAME COUT is added to each CICS region startup JCL.

3. The SEDCLINK library is concatenated to the DFHRPL DD statement for each CICS region.
4. We also verified that the library SEDCLINK, where the EDCCICS module is stored, is APF authorized. The SEDCLINK library is concatenated to the STEPLIB DD statement for each CICS region.

2.2.5 Enabling the FFST Support for CICS

The installation procedures of FFST/MVS support for CICS are found in the program product directory. After the Receive and Apply steps, follow instructions in Chapter 7, " Pre-installation Programming Considerations ", to enable the FFST/MVS on the MVS/ESA system. Be sure that APAR PN54333 is installed, to enable you to perform the VisualInfo Post Link process.

To invoke FFST/MVS in a CICS environment, change the CICS initialization and termination procedures, and define programs associated with the FFST/MVS using the CICS Resource Definition. The required tasks are outlined below:

1. Define the FFST Task Related User Exit, EPWTRUEI, in the initialization Program List Table (PLT).
2. Define the FFST Task Related User Exit, EPWTRUET, in the termination PLT.
3. Define the FFST Task Related User Exit programs, EPWTRUE, EPWTRUEI, and EPWTRUET as CICS resources.
4. Update the CICS System Initialization Table, SIT, with the correct suffix for PLTPI, PLTSD, PPT, and PCT parameters.

   These tables are found in A.7.1, “Library Server SIT” on page 90, and A.7.2, “Object Server SIT” on page 93.
5. Run the FFST installation job EPW12011 that link-edits FFST modules with CICS modules. These modules are stored in the SEPWMOD1 library.
6. The RDO definitions for the FFST were taken from the FFST program directory. A new group EPWFFST was created. It is added to the group lists of the Library Server and and Object Server.
7. Concatenate SEPWMOD1 and SEPWMOD3 on the DFHRPL statement, and SEPWMOD1 on the STEPLIB statement of CICS to start up the JCLs.

2.2.6 Enabling OAM Support for CICS

In order to enable the OAM support for CICS, you must install OAM as instructed in Chapter 5, “OAM and SMS Customizing” on page 23. The following RDO definition defines an new group.

```
DEFINE PROGRAM(CBRICONN) GROUP(OAM) LANG(ASSEM) RESI(YES)
```

The group, OAM, is added to the group list of the Object Server, and SYS1.LINKLIB (where the CBRICONN is stored) is added to the DFHRPL DD statement in the Object Server CICS startup JCL.

Note: Only the Object Server CICS region requires OAM support.

2.3 AD/Cycle C/370 Compiler

The installation procedures of AD/Cycle C/370 Compiler were performed after the installation of the C/370 Common and Specific Libraries. The instructions in the product program directory should be followed.

The SYSLIB DDNAM of the link-edit job, EDCDLINK, should be modified to use C/370 libraries.

```
//SYSLIB    DD    DISP=SHR,DSN=EDC.V2R2M0.SEDCBASE
//          DD    DISP=SHR,DSN=PLI.V2R3M0.SIBMBASE
```

Note: The default SYSLIB of this job is to use LE/370 library.

You have to make sure the compiler is linked with C/370 libraries.

2.4 LE/370

Our system has LE/370 installed. The LE/370 library, SCEERUN, is included in our LNKLST00 member in the SYS1.PARMLIB. However, the VisualInfo Library Server program directory recommends using the C/370 library initially due to the problems encountered in the product testing. C/370 is also installed on our system.

Note: At the time this book was published, problems with LE/370 had not been resolved. We suggest you use C/370.

During the initialization of a CICS region, we get the message:

```
+DFHAP1203I IMLBRSV Language Environment/370 is being initialized.
+CEE1000S LE/370 INTERNAL ABEND. ABCODE = 00000FFD REASON = 00000028
+DFHAP1200I IMLBRSV A CICS request to the Language Environment/370 has failed. Reason code '0011040'.
```

To avoid it, we renamed the module, CEECCICS, in the SCEERUN library.

APAR PN59998 and PN62424 have detailed descriptions of how to circumvent this issue.

Chapter 3. Installing and Customizing the Library Server

This chapter discusses the necessary steps to install and customize the VisualInfo Library Server.

The jobs required for customizing the library server used in this chapter were copied from

```
VILIBSVR.SFRNINS1 to VIUSR1.INSTALL.CNTL
VILIBSVR.SFRNSMP1 to VIUSR1.INSTALL.CNTL
VIUSR1.PRODTAPE.LIBSVR to VIUSR1.INSTALL.CNTL
```

In the rest of this chapter, VILIBSVR.SFRNTBL1 is referred to as SFRNTBL1.

3.1 Receiving and Accepting the Product Tapes

We followed the instructions from the VisualInfo Library Server program directory to receive the SMP jobs from the tape to VIUSR1.PRODTAPE.LIBSVR.

The Library Server and the Object Server should not be installed in the same SMP/E zone. Receive the Library Server into an SMP/E zone primed for IMS/DB2 (SREL P115).

The high level qualifier that we used for the Library Server is VILIBSVR. All of the data sets are allocated on the BNKWK2 volume.

Table 1. Library Server SMP Jobs Description

Job Name	Description	Return Code
FRNALLOC	Defines required data sets for the Library Server. The number of the blocks for SFRNLMD1 and SFRNLMD3 should be increased. Otherwise, if you run the post link job, FRNPSTLK, several times, you receive a E37 completion code.	0
FRNDDDEF	Defines Library Server to the SMP/E system.	0
FRNREC	receives the Library Server FMID, HQX5110 into the system.	0
FRNAPPCK	Performs the Apply Check.	0
FRNAPP	Performs the Apply step.	4
FRNPSTLK	Performs the post-apply link-edit to resolve external references. The load libraries are copied as backups prior to the link-edit step. If the link-edit is not successful, restore the backups before running this job again (we created a job called FRNRSTOL for this purpose).	0
FRNACCC	Performs the Accept Check.	
FRNACC	Performs the Accept step.	

Notes on FRNPSTLK:

1. You should make sure that the PLI has the CICS support installed and the FFST/MVS CICS modules installed.
2. See the table below for the replacement tokens used.

Table 2. Replacement Tokens used for job FRNPSTLK

Token Name	Value	Comments
CIS	CICS330	High-level qualifier of the CICS load library
DB2	DSN310	High-level qualifier of the DB2 system library
DSN	SDSNLOAD	Low-level qualifier of the DB2 system load library
FFS	FFST	High-level qualifier of the FFST system library
FRN	VILIBSVR	High-level qualifier of VisualInfo library server libraries
LOD	SDFHLOAD	Low-level qualifier of the CICS load library
PLI	PLI.V2R3M0	High-level qualifier for the PL/I runtime library
V22	EDC.V2R2M0	High-level qualifier for the C/370 runtime library
UU	3380	Unit parameter
VV	BNKWK2	DASD volume

3.2 CICS Customizing

Perform the following steps:

1. Create required CICS tables and update required JCL statements in the Library Server startup procedure.
2. Define the connection and sessions from the Library Server to the Object Server and Client.

Important: You cannot start the Library Server CICS region successfully until you complete topic 3.3, “Defining the DB2 Tables” on page 12.

Tip: Module C7: Installing the IBM ImagePlus VisualInfo Library Server for MVS/ESA, page 7, talks about IPOUPDTE to modify jobs. If IPOUPDTE is not available, you can create your own EDIT macro using the following as a suggestion:

```
ISREDIT MACRO
ISREDIT RESET
ISREDIT CHANGE ?AVTCOLL?       FRNCOL01       ALL
ISREDIT CHANGE ?BASECOLL?      FRNIBMCL       ALL
ISREDIT CHANGE ?CATNAME?       FRN            ALL
ISREDIT CHANGE ?CICSHLQ?       CICS.V3R3M0    ALL
ISREDIT CHANGE ?COLLECTION?    FRNIBMCL       ALL
•
•
•
ISREDIT LOCATE 0
EXIT
```

3.2.1 Customizing Required CICS Tables

The updated JCLs tokens for the jobs described in this section are listed in Table 3.

Table 3. Replacement Tokens for CICS Installation

Token Name	Value	Job	Comments
CICSHLQ	CICS330	FRNCSDUP	High-level qualifier containing the CICS load library
CSDHLQ	VIHOST.CICS330	FRNCSDUP	High-level qualifier for CICS System Definition (CSD) where CICS resources are defined
CSDLLQ	DFHCSD	FRNCSDUP	Low-level qualifier for CSD where CICS resources are defined
FRNLIB	VILIBSVR.SFRNTBL1	FRNCSDUP	Fully-qualified name of the library containing the input to the CSD of the library server
FRNLOAD	VILIBSVR.SFRNLMD1	FRNCMSGC	Fully-qualified name of the load library containing the message program FRNMHBMG.
LEHLQ	EDC.V2R2M0	FRNCMSGC	High-level qualifier for the C/370 runtime library
LERUN	SECDBASE	FRNCMSGC	Low-level qualifier for the C/370 runtime library
MSGCLUSTER	VIHOST.FRNHBMSG	FRNCMSGC	Fully-qualified name of the VSAM KSDS cluster for the library server message file
MSGSRC	VILIBSVR.SFRNMSG1 (FRNMGENU)	FRNCMSGC	Fully-qualified name of the VisualInfo message file with the member name
MSGVOL	BNKWK2	FRNCMSGC	DASD volume where the message file VSAM cluster resides
PLIHLQ	PLI.V2R3M0	FRNCMSGC	High-level qualifier for the PLI runtime library
PLIRUN	PLIBASE	FRNCMSGC	Low-level qualifier for the PLI runtime library

1. The member FRNCSDUP is submitted to use members FRNCCSD1 and FRNCCSD2 in the SFRNTBL1 data sets.

   The FRNCCSD1 defines transactions, programs, and files to a group, FRNGMAIN.

   The FRNCCSD2 defines sample index class access program (FRNAnnn0) and index class view search programs (FRNVnnn0) in a group called FRNAVT01.

2. The sample DFHSIT$$ is copied from the CICS330.SDFHSAMP library into the VIUSR1.INSTALL.TABLES as DFHSIT10. The changes are:

Parameter	Value
APPLID	IMLBRSV
ISC	Yes
ECDSASZE	8M
ERDSASZE	8M
EUDSASZE	8M
GRPLIST	LSV1LIST
PDIR	No
UDSASZE	3M
SUFFIX	10
SYSIDNT	LSV1
TD	(3,3)
TS	(,3,3)

The table is successfully assembled and link-edited into the VIHOST.CICS330.SDFHAUTH.

3. The sample DFHDCT2$ is copied from the CICS330.SDFHSAMP library into the VIUSR1.INSTALL.TABLES as DFHDCT10. The Library Server DCT, FRNDCTU1, is copied from the SFRNTBL1 and merged with the DFHDCT10 in the VIUSR1.INSTALL.TABLES. The COPY macro statements:

```
COPY  DFH$DCTD          - ALL SDSCI ENTRIES MUST BE IN HERE
COPY  DFH$DCTR          - BASIC CICS FACILITIES
```

   contain definitions for the C/370.

   We include the C/370 Specific Library entries in the DFHDCT10.

4. The PLT Program Initialization (PI) table, FRNPLTU1, is copied into the VIUSR1.INSTALL.TABLES as DFHPLTI1. The PLT program Shut Down (SD) table, FRNPLTU2, is copied into the VIUSR1.INSTALL.TABLES as DFHPLTS1.

   In the DFHPLTI1, we also include the auto start for the DB2 Attachment facility and the Library Server. You have the option to enable the three Library Server initialization programs. If you have a problem bringing up your CICS region, you can disable these programs to ensure that CICS can come up.

   Once the CICS region is up, you have to define and install the FRNMHBP1 as a transaction (for example, HBP1). The sequence of transactions that you should issue is as follows:

```
HBP1
FRN2
FRNT
```

   In the DFHPLTS1, we also included the auto stop for the DB2 Attachment facility and the Library Server.

5. We updated the DFHSIT10 as follows:

Parameter	Value
DCT	10
PLTPI	I1

PLTSD S1

6. The FRNCMSGC job to load the message file for both servers is modified and submitted. The job was completed successfully. The input file to the message file is from SFRNMSG1(GRNMGENU).
7. We created a job DFHLSGL to add all of the related GROUPs and LISTs to the Library Server Group List called LSV1LIST. This job ran successfully.
8. We created a member LIBSVR1 in VIHOST.CICS330.SYSIN containing overriding CICS initialization parameters.
9. The member, SFRNSMP1(FRNCONFG) is the sample for override. To make the override active, you have to specify it in your CICS startup JCL FRNCONFG DDname statement. In our case, we decided to use the sample override.
10. As no sample CICS startup JCL is supplied, we copied one from a current CICS system into the VIUSR1.INSTALL.CNTL as DFHLS. The following adjustments were made:
    - Added FRNICJCL, FRNICTCR, FRNINTRD, FRNHBMSG, and FRNSTATS DD statements in addition to the FRNCONFG.
    - Changed SIT parameter on PARM to 10
    - Updated the STEPLIB and DFHRPL DD statements with the required link libraries
11. The Resource Control Table (RCT) is created in the 3.3, “Defining the DB2 Tables” on page 12.

3.2.2 Defining Connections and Sessions

We have to define the required LU 6.2 sessions to our Object Server and Client. The mode table that we used for this project is found in topic A.4, “Modetab and Logmode Definitions” on page 86.

1. The member FRNCNCLI from the SFRNSMP1 sample library is copied into our VIUSR1.INSTALL.TABLES for modification. We modified it based on the naming convention that we use for ITSO Bethesda. The group we created for the client workstations is LS2CLNT.

   The modified definitions are found in topic A.6.2, “Library Server to Client” on page 88.
2. The sample definitions for connection and session between the Library Server and Object Server are not provided. The FRNCNCLI from SFRNSMP1 is copied as a new member, LS2OS, in VIUSR1.INSTALL.TABLES for modification. The definitions that we defined are in a new group, LS2OS.

   The definitions are found in topic A.6.1, “Library Server to Object Server” on page 87.
3. The connection and session definitions for the Library Server to the LAN-based Object Server were not performed since we are not using any LAN-based Object Server.

3.3 Defining the DB2 Tables

The updated JCLs tokens for the jobs described in this section are listed in Table 4.

Table 4 (Page 1 of 2). Replacement Tokens for Base DB2 Table Installation

Token Name	Value	Job	Comments
CATNAME	VIHOST	FRNDBCRT FRNDBCR2 FRNVSDEL	The high-level qualifier for the datasets created for the DB2 tables and indexes.
COLLECTION	FRNIBMCL	FRNPKBD1 FRNPKBD2 FRNPLNBD	The name of the DB2 collection where VisualInfo DB2 packages reside.
CREATOR	VIUSR1	FRNDBALS FRNDBAL2 FRNDBCRT FRNDBCR2 FRNDBGRT FRNDBGR2 FRNDBLOD FRNDBLD2 FRNDBSYN FRNPKBD1 FRNPKBD2 FRNPLNBD	The qualifier for the VisualInfo DB2 tables that uniquely identifies the Library Server. This must be cross-referenced with the values in the TABLEPREFIX columns of the FRNCNTL table. Each client workstation must have a corresponding entry in the FRNOLINT.TBL
DATABASE	FRNDB001	FRNDBCRT FRNDBCR2	The name of the DB2 database where the tables for the Library Server reside.
DB2RUN	DSN310.RUNLIB.LOAD	FRNDBALS FRNDBAL2 FRNDBCRT FRNDBCR2 FRNDBGRT FRNDBGR2 FRNDBLOD FRNDBLD2 FRNDBSYN	The fully-qualified name of the DB2 runtime library containing DSNTIAD.
DB2SYS	DB3B	FRNDBALS FRNDBAL2 FRNDBCRT FRNDBCR2 FRNDBGRT FRNDBGR2 FRNDBLOD FRNDBLD2 FRNDBSYN FRNPKBD1 FRNPKBD2 FRNPLNBD	The name of the DB2 sub-system where the tables for the Library Server reside.

Table 4 (Page 2 of 2). Replacement Tokens for Base DB2 Table Installation

Token Name	Value	Job	Comments
DSN	DSN310	FRNDBALS FRNDBAL2 FRNDBCRT FRNDBCR2 FRNDBGRT FRNDBGR2 FRNDBLOD FRNDBLD2 FRNDBSYN FRNPKBD1 FRNPKBD2 FRNPLNBD	The high-level qualifier for DB2 system library DSNLOAD.
FRNDBRM	VILIBSVR.SFRNDBR1	FRNPKBD1 FRNPKBD2	The fully-qualified name of the database request module (DBRM) library containing the VisualInfo DBRMs.
OWNER	VIUSR1	FRNPKBD1 FRNPKBD2 FRNPLNBD	The owner of the DB2 packages and plans.
TIADPLAN	DSNTIA31	FRNDBALS FRNDBAL2 FRNDBCRT FRNDBCR2 FRNDBGRT FRNDBGR2 FRNDBLOD FRNDBLD2 FRNDBSYN	The plan name for the DB2 program DSNTIAD.
VOL	BNKWK1 BNKWK2	FRNDBCRT FRNDBCR2	The DASD volume for the DB2 tables.

1. The member, FRNDBCRT, needs to be modified to use correct the volumes and datasets for IDCAMS and DB2 to create the Library Server databases.

   We put all of the databases on one volume, BNKWK1. For better performance, they should be on different volumes. Table 1, in *Module C7: Installing the IBM ImagePlus VisualInfo Library Server for MVS/ESA* provides the information on the usage of the each table. You can select your volumes based on this.

2. The member, FRNDBCR2, needs to be modified to use correct volumes and datasets for IDCAMS and DB2 to create the Folder Manager databases.

   We put all of the databases on one volume, BNKWK2. For better performance, they should be on different volumes. Table 1, in *Module C7: Installing the IBM ImagePlus VisualInfo Library Server for MVS/ESA* provides the information on the usage of the each table. You can select your volumes based on this.

3. The job, FRNDBLOD, JCL was modified before submitting. The values of the table were changed as following:

   a. The value of the TablePrefix column in the table FRNCNTL is changed to VIUSR1.

b. The values of columns in the row where ObjServCode = 1 in the table, FRNOBJECTSERVER, were changed as follows:

   - SysID field '????' to OSV1.
   - ObjServName field from OBJSERV2 to IMOBRSV.

c. The value of SmsCollName field in the table, FRNCOLLNAME, is changed from CBR.CLLCT001 to VIHOST.CLLCT001.

   Notes:

   1) This collection name should be the same as the name specified for customizing OAM and SMS (see topic Chapter 5, "OAM and SMS Customizing" on page 23).

   2) If you plan to use CBR.CLLCT001 as your collection name, you must define the CBR as an alias.

The job was run with condition code 0.

For better performance, you might want to run jobs FRNPLBCS and FRNRUNC.

4. The job, FRNDBLD2, JCL was modified before submitting. It inserts the data into the Folder Manager DB2 tables. The job was run with condition code 0.
5. The FRNDBSYN job was modified and submitted. The synonyms for the Library Server tables were created.
6. The FRNPKBD1 job was modified and submitted. The packages for the Library Server DBRMs were created.
7. The FRNPKBD2 job was modified and submitted. The packages for the Folder Manager Index Class DBRMs were created.
8. The FRNPLBND job was modified and submitted. The binding of Library Server plans was done successfully.
9. The sample, FRNRCTU1, was copied into VIUSR1.INSTALL.TABLES as DSNCRCT1 for modification. The job was completed successfully.

   In our Library Server startup procedure, the DSNCRCT1 was added as the first parameter for the DFHSIP.
10. The job, FRNDBALS, for the Library Server alias SQL statements was submitted and completed successfully.
11. The job, FRNDBAL2, for the Folder Manager alias SQL statements was submitted and completed successfully.
12. The job, FRNDBGRT, grant SELECT, INSERT, DELETE, and UPDATE authorization on the Library Servertables to the transaction, FRNI, FRNT, and FRII. The job was completed successfully.
13. The job, FRNDBGR2, grant SELECT, INSERT, DELETE, and UPDATE authorization on the Folder Managertables to the transaction, FRNI and FRNT. The job was completed successfully.

Important: You can attempt to start up your Library Server CICS region. If the CICS region is up successfully, you can continue to the next section.

3.4 Enabling the Index Classes

The updated JCLs tokens for the jobs described in this section are listed in Table 5.

Table 5 (Page 1 of 2). Replacement Tokens for Index Class Generation

Token Name	Value	Job	Comments
AVTCOLL	FRNCOL01	FRNBDAVT	The name of the DB2 collection for the Claims (AVT) packages reside.
BASECOLL	FRNIBMCL	FRNBDAVT	The name of the DB2 collection for the base VisualInfo packages.
CATNAME	VIHOST	FRNBDAVT FRNDBCR3	The high-level qualifier for the datasets created for the DB2 tables and indexes.
COLLECTION	FRNIBMCL	FRNICOVR	The name of the DB2 collection for the base VisualInfo packages.
CREATOR	VIUSR1	FRNBDAVT FRNDBCR3 FRNDBLD3	The qualifier for the VisualInfo DB2 tables that uniquely identifies the Library Server.
DATABASE	FRNDB001	FRNICOVC	The name of the DB2 database where the tables for the Library Server reside.
DB2RUN	DSN310.RUNLIB.LOAD	FRNBDAVT FRNDBCR3 FRNDBLD3	The fully-qualified name of the DB2 runtime library containing DSNTIAD.
DB2SYS	DB3B	FRNBDAVT FRNDBCR3 FRNDBLD3	The name of the DB2 sub-system where the tables for the Library Server reside.
DSN	DSN310	FRNBDAVT FRNDBCR3 FRNDBLD3	The high-level qualifier for DB2 system library DSNLOAD.
ICOVRCLUSTER	VILIBSVR.IMLBRSV. FRNICTCR	FRNICOVC	The fully-qualified VSAM KSDS cluster name. This is the target file and is the data set referenced by the DDNAME of FRNICTCR in the CICS startup JCL.
ICOVRSRC	VILIBSVR.SFRNVSA1 (FRNICOVR)	FRNICOVC	The source file containing the the Index Class CREATE TABLE/INDEX and Package Bind OWNER and COLLECTION overrides.
ICOVRVOL	BNKWK2	FRNICOVC	The DASD volume containing the target VSAM cluster.

Table 5 (Page 2 of 2). Replacement Tokens for Index Class Generation			
Token Name	**Value**	**Job**	**Comments**
OWNER	VIUSR1	FRNBDAVT FRNICOVC FRNDBLD3	The owner of the DB2 packages and plans.
TIADPLAN	DSNTIA31	FRNBDAVT FRNDBCR3 FRNDBLD3	The plan name for the DB2 sample program DSNTIAD.
VOL	BNKWK1 BNKWK2	FRNDBCR3	The DASD volume for the DB2 tables.

1. The member FRNICOVR in the SFRNVSA1(*) was modified as follows:

```
FRIICOLLECTION*FRNIBMCL
FRIIOWNER     *VIUSR1
FRNICOLLECTION*FRNIBMCL
FRNIINDEX     *USING VCAT VILIBSVR PCTFREE 35 FREEPAGE
FRNIOWNER     *VIUSR1
FRNITABLE     *IN FRNDB001.+
FRNTCOLLECTION*FRNIBMCL
FRNTOWNER     *VIUSR1
```

   We used table 5 and table 6 in Module C9: Enabling Index Classes for the IBM ImagePlus VisualInfo Library Server for MVS/ESA as the guideline for the modification.

   The job, FRNICOVC, is used to create a VSAM file for the FRNICTCR. Our modified input, FRNICOVR, is REPRO into the new VSAM file. The DDNAME of the FRNICTCR in our Library Server start procedure is modified to point at this VSAM file.

2. The job, FRNDBCR3, is modified to create a sample 'claim' index class, AVT00008. This is an optional installation step, but we did it for installation verification.

3. Since we decided to do the optional step, job FRNDBLD3 was then modified and submitted to load the sample index class data.

4. To generate the AVT program for an index class, a job is submitted at runtime to the MVS internal reader. To create this job, we used the sample FRNICGN3 member that is supplied, and modified it to use our DB2 load libraries.

   Note: You also have to examine the other two samples to decide which is the best for your environment.

   In conjunction with this member, a procedure, FRNICGP3, is also supplied that we modified and placed in our procedure library, SYS1.PROCLIB. Refer to Table 6 on page 17 for the replacement tokens used.

   The FRNICJCL DD statement of the Library Server CICS startup procedure was updated to reflect the member as FRNICGN3.

5. To test FRNICGN3, we logged on to the Library Server CICS region and ran transaction FRII. The job was on the Hold queue, so we reviewed and released it. This job was run successfully.

6. Job FRNBDAVT is copied from the SFRNSMP1 to VIUSR1.INSTALL.TABLES. It is then modified and submitted to bind these DBRMs.

   Refer to Table 6 on page 17 for the replacement tokens used.

Note: The collection name specified must be the same as the one used in step 1.

The bind step was run successfully.

Table 6. Replacement Tokens for CLAIM index class

Token Name	Value	Comments
APPLHLQ	VILIBSVR	The 'application' high-level qualifier used to prefix the source(C), DBRM and LOAD libraries.
DSNHLQ	DSN310	Used to identify the DB2 libraries used in the DB2 precompile and program linkedit.
CCHLP	EDC.V2R2M0	The high-level qualifier of the C/370 libraries.
CCMPLR	EDCCOMP	Compiler name.
LEHLQ	EDC.V2R2M0	The high-level qualifier of the C/370 libraries.
LERUN	SEDCBASE	The low-level qualifier of the C/370 runtime library for compilation.
LELKED	SEDCLINK	The low-level qualifier for C-specific library used for linkedit.
CICSHLQ	VIHOST.CICS330	The high-level qualifier of the CICS libraries.
CTLLIB	DSN310.RUNLIB.LOAD	Identify a library of 'control cards'. The only member used is DSNTIAD that identifies the DB2 to attach to and invokes the program DSNTIAD.
PLIHLQ	PLI.V2R3M0	The high-level qualifier of the PL/I libraries.
PLIRUN	PLIBASE	The low-level qualifier of PL/I runtime library.
PLILKED	PLILINK	The low-level qualifier of the PL/I linkedit library.
RPLLIB	VILIBSVR.SFRNLMD1	The fully-qualified load library data set in the CICS RPL concatenation.

3.5 Optimizing Index Class View Searches

To generate the static SQL searches for an index class, a job is submitted at runtime to the MVS internal reader from the Library Server CICS region. To create this job:

1. Modify the sample member FRNSQ2GN that is supplied.

   Note: You could also examine the other two samples to decide which is the best for your environment.

   In conjunction with this member, modify the procedure FRNSQ2PP, and copy to a procedure library, such as SYS1.PROCLIB.

2. Modify a second job, the FRNSQ2BJ bind job, to use the correct load libraries. Specify TYPRUN=HOLD if you wish to submit it manually. Modify the related procedure, FRNSQ2BP, and copy to a PROCLIB.

This job could be incorporated into job FRNSQ1GN to allow VisualInfo to substitute (for instance, +member+, +viewid+, and so on).

3. Concatenate FRNSQ2GN and FRNSQ2BJ as the DSN for //FRNSQJCL.
4. Update the FRNSQJCL DD statement of the Library Server CICS startup procedure to reflect the member as FRNSQ2GN.

Refer to Table 6 on page 17 for the replacement tokens used.

Chapter 4. Installing and Customizing the Object Server

This chapter discusses the necessary steps to install and customize the VisualInfo Object Server.

4.1 Receiving and Accepting the Product Tapes

We followed the instructions from the VisualInfo Object Server program directory to receive the SMP jobs from tape to VIUSR1.PRODTAPE.OBJSVR.

Since both servers cannot be installed in the same SMP/E zone, the Object Server was received into an SMP/E zone primed for CICS (SREL C150).

The members from the VIUSR1.PRODTAPE.OBJSVR are copied to the VIUSR1.INSTALL.CNTL. To prevent any duplicate member names, we added an O as the fourth character of the member name. The high-level qualifier that we used for the Object Server is VIOBJSVR. The members are listed in Table 7. All of the data sets are allocated on the BNKWK2 volume.

Table 7. Object Server SMP Jobs Description

Job Name	Description	Return Code
FRNOALLO	Defines required data sets for the Object Server. The number of blocks for SFRNLMD2 should be increased. Otherwise, if you run the post-link job FRNOPSTL several times, you receive a E37 completion code.	0
FRNODDEF	Defines Object Server to the SMP/E system.	0
FRNOREC	Receives the Object Server FMID, HQX8110 into the system.	0
FRNOAPPC	Performs the Apply Check.	0
FRNOAPP	Performs the Apply step. We did not get the message IEW0642 as stated, but the return code of 4 was caused by messages GIM32903W and IEW2454W.	4
FRNOPSTL	Performs the post-apply link-edit to resolve external references. The loadlib is copied as backup prior to the link-edit step. If the link-edit is not successful, restore the backup before running this job again (we created a job called FRNRSTOO for this purpose). **Note:** See the following table for the replacement tokens used.	0
FRNOACCC	Performs the Accept Check.	
FRNOACC	Performs the Accept step.	

Table 8. Replacement Tokens Used for Job FRNOPSTL		
Token Name	**Value**	**Comments**
CIS	CICS330	The high-level qualifier of the CICS load library.
DB2	DSN310	The high-level qualifier of the DB2 system library.
FFS	FFST	The high-level qualifier of the FFST system library.
FRN	VIOBJSVR	The high-level qualifier of VisualInfo Object Server libraries.
LOD	SDFHLOAD	The low-level qualifier of the CICS load library.
PLI	PLI.V2R3M0	The high-level qualifier for the PL/I runtime library.
V22	EDC.V2R2M0	The high-level qualifier for the C/370 runtime library.
UU	3380	The unit parameter.
VV	BNKWK2	The DASD volume.

4.2 CICS Customizing

The jobs required for customizing were copied from VIOBJSVR.SFRNSMP2 to VIUSR1.INSTALL.CNTL.

In the remainder of this chapter, VIOBJSVR.SFRNSMP2 is referred to as SFRNSMP2.

Perform the following steps:

1. Create the required CICS tables and update the required JCL statements in the Object Server startup procedure.
2. Define the connection and sessions from the Object Server to the Library Server and clients.

4.2.1 Customizing Required CICS Tables

The updated JCL tokens for the jobs described in this section are listed in Table 9.

Table 9 (Page 1 of 2). Replacement Tokens for CICS Installation			
Token Name	**Value**	**Job**	**Comments**
CICSHLQ	CICS330	FRNCSDUJ	The high-level qualifier containing the CICS load library.
CSDHLQ	VIHOST.CICS330	FRNCSDUJ	The high-level qualifier for CICS System Definition (CSD) where CICS resources are defined.
CSDLLQ	DFHCSD	FRNCSDUJ	The low-level qualifier for CSD where CICS resources are defined.

Table 9 (Page 2 of 2). Replacement Tokens for CICS Installation			
Token Name	**Value**	**Job**	**Comments**
FRNLIB	VIOBJSVR.SFRNSMP2	FRNCSDUJ	The fully-qualified name of the library containing the input to the CSD of the Object Server.

1. The job FRNCSDUJ is submitted to use member FRNCCSD1 in the SFRNSMP2 data set. The *Module C8: Installing the IBM ImagePlus VisualInfo Object Server for MVS/ESA* refers to this as SFRNTBL1, which is a Library Server table.

   FRNCCSD1 defines the transactions, programs, and files to a group called FRNSAMP2. *Module 8* refers to this as SFRNSMP2.

   You can expect a return code of 4 if the group FRNSAMP2 does not exist.

2. The SIT was copied from the one used for the Library Server into the VIUSR1.INSTALL.TABLES as DFHSIT20. Changes are:

Parameter	**Value**
APPLID	IMOBRSV
ISC	Yes
ECDSASZE	8M
ERDSASZE	8M
EUDSASZE	8M
GRPLIST	OSV1LIST
UDSASZE	3M
SUFFIX	20
SYSIDNT	OSV1
TD	(3,3)
TS	(,3,3)

   The table was successfully assembled and link-edited into VIHOST.CICS330.SDFHAUTH.

3. The Object Server DCT, FRNDCTU1, was copied from the SFRNSMP2 and merged with the Library Server DCT (DFHDCT10) in the VIUSR1.INSTALL.TABLES as DFHDCT20.

   The table was successfully assembled and link-edited into VIHOST.CICS330.SDFHLOAD.

4. The PLT Program Initialization (PI) table was copied into VIUSR1.INSTALL.TABLES as DFHPLTI2 from the Library Server PLT and updated.

5. The PLT Program Shut Down (SD) table was copied into the VIUSR1.INSTALL.TABLES as DFHPLTS2 from the updated Library Server PLT.

6. The step to create the message file was bypassed as this was already done when the Library Server was installed. The same message file is used.

7. The Resource Control Table (FRNRCTU1) was copied from the SFRNSAMP into the VIUSR1.INSTALL.TABLES as DSNCRCT2. It was modified and successfully assembled and link-edited into the VIHOST.CICS330.SDFHAUTH.
8. We updated the DFHSIT20 with the following:

Parameter	Value
DCT	20
PLTPI	I2
PLTSD	S2

9. We created a job called DFHOSGL to add all of the related GROUPs and LISTs to the Object Server Group list called OSV1LIST. This job ran successfully.
10. We created a member OBJSRV1 in VIHOST.CICS330.SYSIN containing overriding CICS initialization parameters.
11. The Object Server CICS startup JCL (DFHOS) was copied in VIUSR1.INSTALL.CNTL from the Library Server CICS startup JCL and the following changes made:
    - DSNCRCT2 as the first parameter on the PARM

    Note: The DD statements for the message file and the necessary DFHRPL link libraries were already added during the Library Server installation.

4.2.2 Defining Connections and Sessions

We have to define the required LU 6.2 sessions to our Library Server and Client. The mode table that we used for this project is found in the A.4, “Modetab and Logmode Definitions” on page 86.

1. The member FRNCNCLI from the SFRNSMP2 sample library was copied into our VIUSR1.INSTALL.TABLES for modification. We modified it based on the naming convention that we use for ITSO Bethesda. The group we created for the client workstations was OS2CLNT.

   The modified definitions are found in the A.6.4, “Object Server to Client” on page 89.
2. The sample definitions for connection and session between the Object Server and Library Server are not provided. The FRNCNCLI in VIUSR1.INSTALL.TABLES was copied as a new member (OS2LS) in VIUSR1.INSTALL.TABLES for modification. The definitions that we defined are in a new group (OS2LS).

   The definitions are found in the A.6.3, “Object Server to Library Server” on page 88.
3. The connection and session definitions for the Object Server to the LAN-based servers were not performed since we are not using any LAN-based servers.

Chapter 5. OAM and SMS Customizing

This chapter describes the steps required to successfully customize OAM for object management in a VisualInfo/MVS environment. It consists of:

- Customizing OAM
- Customizing SMS

5.1 Objectives

Our objectives are to provide a straightforward OAM environment that allows us to install and test the VisualInfo under MVS/ESA. We did not require all 100 DB2 object storage groups to achieve this, so we created only five, GROUP00 to GROUP04.

Both CICS and DB2 require modification to provide OAM support. For CICS, the changes must be made to the region running VisualInfo Object Server. SMS was not activated on our system, so we had to create an SCDS (source control data set). We did not have an optical library on our system.

We decided to run with the OAM address space active because we felt this would provide a more comprehensive test of OAM.

Note: The OAM address space is required to be active only if you are managing optical libraries and devices or using OSMC to process expired objects. Storage and retrieval of the objects does not require the OAM address space to be active.

5.2 Preparation

Before beginning this section, the following should be decided:

- How many object storage groups are required?
- What sizes those groups should be?
- What SMS management and storage class criteria are applied to those groups?

We suggest that you have the following manuals available:

- *OAM Planning, Installation and Storage Administration Guide for Object Support*
- *CICS/ESA Resource Definition (Macro)*
- *DB2 Version 3 Administration Guide*
- *Storage Administration Reference for DFSMSdfp*
- *OAM/SMS Design Guidelines*
- *Using the Interactive Storage Management Facility*

Caution: For this installation, we used DFSMS/MVS V1.2. There is a difference in the installation procedures from MVS/DFP 3.3.1. Please ensure that the right manual is used (SC26-4918). If a customer is installing OAM on DFP 3.3.1, manual SC35-0120, level 02 or 03 should be used.

5.3 Customizing OAM

To customize OAM, we followed the steps described in Chapter 3, “Migrating, Installing and Customizing OAM” of the *OAM Planning, Installation and Storage Administration Guide for Object Support*.

1. Changing DB2 installation parameters
2. Changing CICS installation parameters
3. Modifying the installation exit for deleted objects
4. Changing system libraries
5. Creating DB2 databases for object tables and directories
6. Creating optical configuration (LCS) database

   or

   Creating OSR application plans

   Note: If you want to start the OAM address space, you have to create the optical configuration database. If you want to store objects without starting the OAM address space, you have to create the OSR applications plans. In our case, we took the first option.
7. Creating OSMC application plans
8. Creating LCS, ISMF and OSR application plans
9. Running OAM IVP

All required jobs are in SYS1.SAMPLIB with the prefix CBR. We copied them to our VIUSR1.INSTALL.JCL library. The changes we made to the installing jobs to suit our own environment, are detailed below:

- STEPLIB to DSN310.SDSNLOAD as the DB2 load library
- DB2 runtime library to DSN310.RUNLIB.DATA
- DB2 system name to DB3B
- DB2 DBRM library to DSN310.DBRMLIB.DATA
- DB2 plan name for DSNTIAD to TIAD31A
- VCAT parameter to FRNR110

Note: Be careful when you are running the jobs using DSNTIAD. Unlike SPUFI, no rollback is performed; DSNTIAD proceeds creating up to ten errors. For this reason, you have to check which steps have been successfully performed if errors occur.

5.3.1 Changing DB2 Installation Parameters

For DB2, the modifications were made during the installation of DB2. When we installed DB2, we specified:

- ISO date format on panel DSNTIPF. OAM does not require ISO, but it displays the date and time in ISO format.
- DB2 Buffer Pool sizes on panel DSNTIPE. The recommendation in *OAM Planning, Installation and Storage Administration Guide for Object Support* varies from our DB2 installation.

Table 10. OAM DB2 Parameters		
DB2 Install Parameters	**DFSMS 1.2 Recommendation**	**Our DB2 Value**
MAX USERS	200	50
MAX TSO CONNECT	100	100
MAX BATCH CONNECT	100	20
MIN BP0 BUFFERS	200	224
MAX BP0 BUFFERS	300	224
MIN BP1 BUFFERS	200	224
MAX BP1 BUFFERS	300	500
MIN BP2 BUFFERS	100	112
MAX BP2 BUFFERS	200	500
MIN BP32K BUFFERS	50	320
MAX BP32K BUFFERS	100	320

The variations were assumed to have little effect in our test environment, so we decided to leave our DB2 installation parameters unchanged.

- EDM pool size is 9000.
- IMS Resource Lock Manager (IRLM) installation parameter should be 1700 page locks per table space. DB2 default is 1000. It was provided on IRLM Panel 2 (DSNTIPJ):

```
3  LOCKS PER TABLE(SPACE) ===> 1000      Maximum before lock escalation
```

A large number of DB2 locks are taken during OSMC storage group cycles. We did not increase it because we were not expecting major OSMC storage group cycles on our system since we did not have optical libraries.

Note: This parameter needs to be monitored and tuned closely.

5.3.2 Changing CICS Installing Parameters

OAM is accessed by Object Server and hence all the CICS changes must be made to the region running the Object Server. A brief overview of these changes follows:

1. Update or create CICS PLT to include OSR CICS initialization module CBRICONN:

   ```
   DFHPLT TYPE=ENTRY,PROGRAM=CBRICONN
   ```

2. Update CICS PPT to include module CBRICONN:

   ```
   CBRICONN DFHPPT TYPE=ENTRY,PGMLANG=ASSEMBLER,PROGRAM=CBRICONN,RES=YES
   ```

   Also with the PLT, where xx is the correct suffix:

   ```
   DFHPPT TYPE=ENTRY,PROGRAM=DFHPLTxx
   ```

3. Update CICS SIT to include PLTPI parameter and the correct suffix, also the PPT suffix:

   ```
   PLTPI=xx
   PPT=(xx,COLD)
   ```

4. Update DB2/CICS RCT (Resource Control Table) to include plan CBRIDBS:

5. Copy CBRICONN to a load library in the DFHRPL concatenation if you are not using SYS1.LINKLIB.

5.3.3 Modifying Installation Exit for Deleted Objects

Modify the auto-delete exit (CDRHXDUX) to ensure that objects are deleted during the storage management cycle. The default sample exit prevents objects from being deleted.

We decided not to modify the exit.

5.3.4 Changing System Libraries

For OAM support, you must add entries to both SYS1.PARMLIB and your procedure library, in our case SYS1.PROCLIB. The required changes depend on whether you require the OAM address space to be active or not. It must be active if you want to accommodate optical devices, tape object support or Object Storage Management Cycle (OSMC) for DASD space management or deletion of objects. We decided that we wanted the OAM address space to be active, so we did the following:

1. Updated the SCHED00 PARMLIB member:

   - This entry adds OAM OTIS to the PPT required to support OSREQ functions such as storage and retrieval of objects.

     ```
     PPT PGMNAME(CBRIIAS)                  /* OTIS ADDRESS SPACE         *
         KEY(5)                            /* USE DFP PROTECT KEY        *
         SWAP                              /* SWAPPABLE                  *
         SYST                              /* SYSTEM TASK , NOT TIMED    *
     ```

   - Since we planned to run with the OAM address space active, we had to provide this entry for PPT.

     ```
     PPT PGMNAME(CBROAM)                   /* OAM A/S                    *
         KEY(5)                            /* USE DFP PROTECT KEY        *
         NOSWAP                            /* NON-SWAPPABLE              *
         SYST                              /* SYSTEM TASK , NOT TIMED    *
     ```

2. Updated the IEFSSN00 member to include SMS and OAM1 subsystems.

   ```
   JES2,,,PRIMARY,NOSTART      JES2 IS THE PRIMARY SUBSYSTEM NAME
   CICS,DFHSSIN,DFHSSI00
   DB3B,DSN3INI,'DSN3EP,#'     DB2 3.1
   IRLM                        IMS RESOURCE LOCK MANAGER
   JRLM                        SECONDARY SUBSYSTEM NAME FOR IRLM
   FFST                        FFST SYSTEM
   CNMP                        NETVIEW
   EKGX                        NETVIEW
   SMS,IGDSSIIN,'PROMPT=DISPLAY'
   OAM1,CBRINIT,'TIME=GMT,MSG=EM'
   ```

3. Updated the IGDSMS00 PARMLIB member:

   Included the following OAM-related keywords:

   - OAMPROC(OAM) Specifies the procedure name that starts the OAM address space. In our case we have the OAM startup procedure in member OAM in SYS1.PROCLIB.
   - OAMTASK(OAM) Specifies the identifier that starts the OAM address space. In our case it is OAM.
   - DB2SSID(DB3B) Specifies the name of the DB2 subsystem.

4. Updated the IEAICS00 PARMLIB member (optional):

   This is an optional step and is only required if you want to collect RMF statistics. We decided not to do it.

   Note: For more information, refer to “Chapter 3, page 84” of *OAM Planning, Installation and Storage Administration Guide for Object Support*.

5. Updated PROCLIB:

   You must run two jobs to create OTIS and OAM startup procedures:

   a. CBRIPROC

   b. CBRAPROC

   For both jobs, we added a STEPLIB to point to our DB2 load library:

   ```
   //STEPLIB DD DSN=DSN310.SDSNLOAD,DISP=SHR   (NEEDED TO FIND DSNALI)
   ```

   Note: In a normal production environment, this is not necessary as the DB2 load library is concatenated as part of the LNKLST.

6. Verified device numbers:

   Verify the optical device numbers using HCD (hardware configuration definition), or define whether or not you have an optical library installed. You must define dummy optical libraries.

5.3.5 Creating Databases for Object Tables and Directories

Before OAM can work, you must create object storage databases. These databases contain either objects or information about objects. OAM requires a separate object storage group database for each storage group. The following steps must be done:

1. Allocate the OAM data sets for object storage groups using job CBRIALC0.
2. Define the OAM database using job CBRISQL0.
3. Allocate the OAMADMIN data sets for object storage groups using jobs CBRIALCX-Y.
4. Define the OAMADMIN database using jobs CBRISQLX-Y.

5.3.5.1 Allocating the OAM Data Sets for Object Storage Groups

Ten IDCAMS jobs are provided by OAM to allocate the VSAM clusters used by the image object storage groups. These are jobs CBRIALC0 through CBRIALC9. Each job creates 10 object storage groups. Each individual object storage group has a qualifier of GROUPxx, where xx is in the range 00-99. This is an OAM requirement and cannot be changed. The supplied CBRIALC0 allocates GROUP00 to 09.

We required five object storage groups, GROUP00 to GROUP04. We, therefore, modified CBRIALC0 to allocate the required VSAM clusters for GROUP00 to GROUP04. This is an example of the supplied JCL:

```
DEFINE CLUSTER
(NAME(cat_name.DSNDBC.GROUP00.OSMDTS.I0001.A001)
        LINEAR
        SHAREOPTIONS(3 3)
        VOLUMES(vol_ser)
        CYLINDERS(pri_alloc sec_alloc)
        UNIQUE   )
```

```
      DATA
(NAME(cat_name.DSNDBD.GROUP00.OSMDTS.I0001.A001))
```

We made the following modifications to job CBRIALC0:

- We changed cat_name to FRNR110 for the alias under which the databases are cataloged.
- We changed vol_ser to BNKWK1 or BNKWK2 for the volumes on which the databases reside. The split is for performance reasons.
- We set up the space allocation values as shown in Table 11:

Table 11. Cylinders (pri_alloc sec_alloc)

Cluster	GROUP00	GROUP01	GROUP02	GROUP03	GROUP04
OSMDTS	10 1	2 1	2 1	2 1	2 1
OSMOTS04	20 2	5 2	5 2	5 2	5 2
OSMOTS32	100 2	20 2	10 2	10 2	10 2
OBJDIRX1	5 1	1 1	1 1	1 1	1 1
OBJDIRX2	5 1	1 1	1 1	1 1	1 1
OBJDIRX3	5 1	1 1	1 1	1 1	1 1
OBJT04X1	5 1	1 1	1 1	1 1	1 1
OBJT32X1	5 1	1 1	1 1	1 1	1 1

- We removed job steps 05 through 09.

We ran job CBRIALC0 without a problem.

5.3.5.2 Defining the OAM Databases

For each allocation job run, you must run a corresponding SQL to define a tablespace job. These jobs are CBRISQL0 through CBRISQL9. Each step performs the following:

- Creates a database GROUPxx.
- Defines a 32K and 4K object tablespace plus a directory tablespace.
- Creates the associated indexes.
- Creates DB2 views.
- Performs grants to each tablespace.

We only ran job CBRISQL0, and did the following modifications:

- Removed steps 05 to 09.
- Added the following to the STEPLIB statement to run the DB2 command processor DSN under the batch TSO program IKJEFT01:

  ```
  //          DD DSN=DSN310.SDSNLOAD,DISP=SHR
  ```

- Changed the supplied JCL to:

  ```
  //SYSTSIN  DD *
   DSN SYSTEM(DB3B)
   RUN  PROGRAM(DSNTIAD) PLAN(TIAD31A) -
        LIB('DSN310.RUNLIB.LOAD')
  ```

- Changed cat_name to FRNR110

- Changed auth_id to EKCCICS, our AUTHID in the RCT for the Object Server CICS region

We ran job CBRISQL0 without any problem.

5.3.5.3 Allocating the OAMADMIN Data Sets for Object Storage Groups

Two jobs are supplied to allocate the OAMADMIN data sets:

- CBRIALCX
- CBRIALCY

These jobs must be modified as previously described for CBRIALC0:

- Change cat_name to FRNR110
- Change vol_ser to BNKWK1 and BNKWK2 for performance reasons

We ran both jobs without any problem.

5.3.5.4 Defining the OAMADMIN Database

There are two jobs supplied for the definition of the OAMADMIN database that are required for object administration:

- CBRISQLX
- CBRISQCY

These jobs must be modified as previously described for CBRISQL0. We ran both jobs without any problem.

5.3.6 Creating the Optical Configuration (LCS) Database

The optical configuration database is necessary if you plan to start the OAM address space. As this was our intent, we had to define the database.

Job CBRSAMPL creates the optical configuration database, CBROAM. The JCL was modified as follows:

- Changed vol_ser to BNKWK2
- Changed cat_name to alias FRNR110
- Changed the supplied JCL to:

```
//SYSTSIN  DD *
 DSN SYSTEM(DB3B)
 RUN  PROGRAM(DSNTIAD) PLAN(TIAD31A) -
      LIB('DSN310.RUNLIB.LOAD')
```

- Removed the password parameter on the CREATE STOGROUP since we were not using password protection at storage group level.

The *OAM Planning, Installation and Storage Administration Guide for Object Support* is not specific about what is required for the VCAT parameter. It recommends that you use the name of an ICF catalog but does not suggest which one. We used the alias for DB2 table spaces, FRNR110.

5.3.7 Creating OSR Application Plans

This step is necessary if you decide not to create the optical configuration database. For more information, refer to topic B.1, “OSR Application Plan” on page 99.

Note: Since we decided to create the optical configuration database, we bypassed this step.

5.3.8 Creating OSMC Application Plans

Job CBRHBIND creates the OSMC application plans. We made the following modifications to the JCL:

- Added the following to the STEPLIB statement to run the DSN in TSO batch:

  ```
  //          DD DSN=DSN310.SDSNLOAD,DISP=SHR
  ```

- Removed the LIB( ) parameter from the BIND commands
- Changed to DSN SYSTEM(DB3B)
- Made the following changes:

  ```
  //DBRMLIB  DD  DSN=DSN310.DBRMLIB.DATA,DISP=SHR
  ```

There should be a message for each bind performed.

The complete list of plans to be bound are found in Chapter 3, page 105 of the *OAM Planning, Installation and Storage Administration Guide for Object Support*.

Job CBRHGRNT grants run on the OSMC plans to PUBLIC. Besides the changes mentioned previously, we modified:

```
//SYSTSIN  DD *
 DSN SYSTEM(DB3B)
 RUN  PROGRAM(DSNTIAD) PLAN(TIAD31A) -
      LIB('DSN310.RUNLIB.LOAD')
```

We ran jobs CBRHBIND and CBRHGRNT successfully.

5.3.9 Creating LCS, OSR and ISMF Application Plans

After successful creation of the optical configuration database, binds and grants must be performed. Job CBRABIND creates plans CBROAM, CBRIDBS and CBRISMF. We made the following modifications to the JCL:

- Added the following to the STEPLIB statement to run the DSN in TSO batch:

  ```
  //          DD DSN=DSN310.SDSNLOAD,DISP=SHR
  ```

- Changed to DSN SYSTEM(DB3B)
- Changed to LIB(DSN310.RUNLIB.LOAD)
- Changed:

  ```
  //DBRMLIB  DD  DSN=DSN310.DBRMLIB.DATA,DISP=SHR
  ```

Look for the following message:

```
DSNT200I BIND FOR PLAN CBROAM SUCCESSFUL
DSNT200I BIND FOR PLAN CBRISMF SUCCESSFUL
DSNT200I BIND FOR PLAN CBRIDBS SUCCESSFUL
```

Job CBRAGRNT grants run on these plans to PUBLIC. Besides the changes mentioned previously, we modified:

```
//SYSTSIN  DD *
 DSN SYSTEM(DB3B)
 RUN  PROGRAM(DSNTIAD) PLAN(TIAD31A) -
      LIB('DSN310.RUNLIB.LOAD')
```

We ran jobs CBRABIND and CBRAGRNT successfully.

5.3.10 Running OAM IVP

At this stage of the installation, an IVP is run against the OAM system. This consists of two steps:

1. Verify that all application plans have been created
2. Verify that all of the application plans have been authorized

For Step 1, use SPUFI to run:

```
SELECT * FROM SYSIBM.SYSPLAN
   WHERE NAME LIKE 'CBR%'
```

You should get the following output:

```
CBROAM
CBRHSMSI
CBRHOBJP
CBRHSOBP
CBRHSBKV
CBRHSVOL
CBRHRDAS
CBRHWDAS
CBRHDUPD
CBRHSBCC
CBRHSPCC
CBRISMF
```

For Step 2, run:

```
SELECT * FROM SYSIBM.SYSPLANAUTH
   WHERE NAME LIKE 'CBR%'
```

to verify authorization to those plans. The output was the same as in the previous step and indicated that all plans had run authority of PUBLIC.

There is a further verification of OAM that can done using the OSREQ macro. This verification requires that your SMS environment be complete. See topic 5.4.12, “Verification of OAM Installation” on page 51 for more details.

At this point, OAM customizing is complete. All that remains is to create the SMS environment for managing image objects. The details are in the following section.

You also have to re-IPL your system to add the changes you made to your system libraries. After the IPL was completed, we got these system messages CBR8001I and CBR8002I. We also received CBR0095E. OAM was waiting for the SMS control data set activation. We had not yet created the SMS data sets.

Note: Ensure that OTIS is started before OAM.

5.4 Customizing SMS

This section describes the tasks required to enable SMS to manage image objects.

The major steps are:

1. Define data sets for SMS.
2. PARMLIB Updates.
3. ISMF Tailoring.
4. Define the base SCDS.
5. Define libraries and drives in the optical configuration database.
6. Define storage groups and relate the libraries to the storage groups.
7. Define backup storage groups and relate the libraries. to the storage groups.
8. Define storage classes.
9. Define management classes.
10. Define Automatic Class Selection.
11. Validate and activate the configuration.
12. Verify the OAM installation.
13. Activate the OAM address space.
14. Additional steps (optional).

Since SMS was not active on our system, it was necessary to create the base SMS data sets and SYS1.PARMLIB definitions. If your installation already has an active SMS subsystem, skip these steps and proceed with topic 5.4.3, “ISMF Tailoring” on page 33.

5.4.1 Defining Data Sets for SMS

You may find that *Storage Administration Reference for DFSMSdfp* is useful for this work.

SMS requires the following control data sets:

- Source Control Data Set (SCDS)
- Active Control Data Set (ACDS)
- Command Data Set (COMMDS)

We created job CBRSMSDS containing the following JCL to create these data sets:

```
//VIUSR1    JOB   (ITSO,B133),'JIM A TAU',CLASS=A,
//          MSGCLASS=X,MSGLEVEL=(1,0),NOTIFY=VIUSR1
//STEP1  EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=X
//SYSUDUMP DD SYSOUT=X
//SYSIN    DD *
  DEFINE CLUSTER( -
                NAME(SMS1.BASE.SCDS1) -
                LINEAR -
```

```
                VOL(ITSC00) -
                KILOBYTES(216) -
                SHAREOPTIONS(2,3)) -
                DATA(NAME(SMS1.BASE.SCDS1.DATA))
DEFINE CLUSTER( -
                NAME(SMS1.BASE.ACDS1) -
                LINEAR -
                VOL(ITSC00) -
                SHAREOPTIONS(3,3)) -
                DATA(NAME(SMS1.BASE.ACDS1.DATA))
DEFINE CLUSTER( -
                NAME(SMS1.BASE.COMMDS1) -
                LINEAR -
                VOL(ITSC00) -
                KILOBYTES(216) -
                KILOBYTES(10) -
                SHAREOPTIONS(3,3)) -
                DATA(NAME(SMS1.BASE.COMMDS1.DATA))
```

Check the size recommendations in Chapter 2 of *Storage Administration Reference for DFSMSdfp*.

5.4.2 PARMLIB Updates

We carried out the changes as part of topic 5.3.4, “Changing System Libraries” on page 26. We checked members of SYS1.PARMLIB:

- IGDSMS00

```
SMS ACDS(SMS1.BASE.ACDS1)
    COMMDS(SMS1.BASE.COMMDS1)
    ACSDEFAULTS(YES)
    OAMPROC(OAM)
    OAMTASK(OAM)
    DB2SSID(DB3B)
```

- IEASYS00 for the entry SMS=00 and SMF=00

5.4.3 ISMF Tailoring

You need storage administrator authority to complete the ISMF tasks. To check if you have this authority:

- Select Option 0 from the ISMF main menu panel.
- Then select Option 0 **USER MODE**.
- User Mode must be 2 for storage administrator.

Note: You must exit and enter ISMF again to view and use your selected session.

The following is a layout of the ISMF main panel for the storage administrator:

```
                          ISMF PRIMARY OPTION MENU
ENTER SELECTION OR COMMAND ===>

SELECT ONE OF THE FOLLOWING OPTIONS AND PRESS ENTER:

 0  ISMF PROFILE              - Specify ISMF User Profile
 1  DATA SET                  - Perform Functions Against Data Sets
 2  VOLUME                    - Perform Functions Against Volumes
 3  MANAGEMENT CLASS          - Specify Data Set Backup and Migration Criteria
 4  DATA CLASS                - Specify Data Set Allocation Parameters
 5  STORAGE CLASS             - Specify Data Set Performance and Availability
 6  STORAGE GROUP             - Specify Volume Names and Free Space Thresholds
 7  AUTOMATIC CLASS SELECTION - Specify ACS Routines and Test Criteria
 8  CONTROL DATA SET          - Specify System Names and Default Criteria
 9  AGGREGATE GROUP           - Specify Data Set Recovery Parameters
10  LIBRARY MANAGEMENT        - Specify Library and Drive Configurations
 C  DATA COLLECTION           - Process Data Collection Function
 L  LIST                      - Perform Functions Against Saved ISMF Lists
 R  REMOVABLE MEDIA MANAGER   - Perform Functions Against Removable Media
 X  EXIT                      - Terminate ISMF


 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 1. ISMF Administrators Panel

5.4.4 Defining the Base SCDS

In the following screen examples, the values we entered are highlighted. All other values are defaults.

ISMF panels are sometimes confusing. Be sure that all of the panels are completed, especially when defining management classes.

Begin by defining the base configuration:

- From the ISMF main menu, select option 8, **CONTROL DATA SET**.
- Select option 2, **DEFINE the Base Configuration**.
- Type the name of your base SCDS.
- Press the Enter key.

```
                          CDS APPLICATION SELECTION
COMMAND ===>

TO PERFORM CONTROL DATA SET OPERATIONS, SPECIFY:

  CDS NAME ===> 'SMS1.BASE.SCDS1'
                          (1 to 44 character data set name or 'ACTIVE')

SELECT ONE OF THE FOLLOWING OPTIONS ===> 2

 1  DISPLAY  - Display the Base Configuration
 2  DEFINE   - Define the Base Configuration
 3  ALTER    - Alter the Base Configuration
 4  VALIDATE - Validate the SCDS
 5  ACTIVATE - Activate the CDS









 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 2. CDS Application Selection Panel

The SCDS BASE DEFINE panel is shown:

```
                                SCDS BASE DEFINE
COMMAND ===>

SCDS NAME:    SMS1.BASE.SCDS1
SCDS STATUS:  INVALID

TO ALTER SCDS BASE, SPECIFY:
  DESCRIPTION ===> BASIC SMS SYSTEM FOR OAM
              ===>
  DEFAULT MANAGEMENT CLASS ===>            (1 to 8 characters)
  DEFAULT UNIT             ===> SYSDA      (esoteric or generic device name)
  DEFAULT DEVICE GEOMETRY
    BYTES/TRACK            ===> 47476      (1-999999)
    TRACKS/CYLINDER        ===> 15         (1-999999)

  SPECIFY ONE OF THE FOLLOWING OPTIONS ===> 1    (1 Add, 2 Delete, 3 Rename)
         SYSTEM NAME       ===> BETH
         NEW SYSTEM NAME   ===>                  (For option 3, Rename)

  SYSTEMS:
  SYSTEMS:

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 3. SCDS Base Define Panel

The only required field on this panel is “SYSTEM NAME.” The value BETH is from SID in member SMFPRM00 in SYS1.PARMLIB:

```
SID(BETH)                  /* SYSTEM ID IS BETH  */
```

Member SMFPRM00 is referenced by entry SMF=00 in SYS1.PARMLIB(IEASYS00).

Except for DESCRIPTION, all other fields are ignored for objects.

5.4.5 Defining Libraries and Drives in the Optical Configuration Database

If you intend to run with the OAM address space active, you must define an optical library.

You must define at least one optical library as PSEUDO (even if you do not have any physical optical device), because during address space setup, OAM attempts to access the optical library configuration. In such an environment, it is probably best to use the 9246 as the library device type. This is the route we followed.

However, in most environments, the 3995 optical library is used and in this case, at least one REAL library and one PSEUDO or OPERATOR-ACCESSIBLE drive must be defined. This is described under topic B.2.2, “3995 Optical Library” on page 103.

To define a PSEUDO 9426 optical library, do the following:

- From the ISMF main menu, select option 10, **LIBRARY MANAGEMENT**.
- Select option 1, **Library Configuration** on the Library Management selection menu.

The LIBRARY APPLICATION SELECTION panel is shown Figure 4:

- Enter LIBDUMMY in the library name field.
- Select option 3, and press the Enter key.
- Enter 9246 for library device type. Although we were trying to create a dummy library, we had to provide a valid value.

```
                    OPTICAL LIBRARY APPLICATION SELECTION
COMMAND ===>

TO PERFORM LIBRARY OPERATIONS, SPECIFY:

  CDS NAME            ===> 'SMS1.BASE.SCDS1'
                                (1 to 44 character data set name or 'ACTIVE')
  LIBRARY NAME        ===> LIBDUMMY  (For Optical Library List, fully or
                                      partially specified or * for all)
  LIBRARY DEVICE TYPE ===> 9246      (For Optical Library List, fully or
                                      partially specified or * for all)
SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

 1   LIST    - Generate a list of Libraries
 2   DISPLAY - Display a Library
 3   DEFINE  - Define a Library
 4   ALTER   - Alter a Library

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
   RESPECIFY SORT CRITERIA           ===> N  (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 4. Library Application Selection Panel

Note: If you do not have the SMS address space active at this point, you may receive the following error:

```
DB2SSID MISSING FROM THE SMS SSIVT CONTROL BLOCK
```

You must activate the SMS address space using the following command:

```
T SMS=00
```

and retry the operation.

The 9246 LIBRARY DEFINE panel is shown. Enter the parameters shown in Figure 5.

```
                              9246 LIBRARY DEFINE
COMMAND ===>

SCDS NAME:    SMS1.BASE.SCDS1
LIBRARY NAME: LIBDUMMY

TO DEFINE LIBRARY, SPECIFY:

  DESCRIPTION ===> DUMMY OPTICAL LIBRARY
              ===>

  LIBRARY TYPE           ===> PSEUDO             (REAL or PSEUDO)

  THE FOLLOWING FIELDS ARE FOR REAL LIBRARY TYPE ONLY:

  ONLINE STATUS          ===>                    (Y or N)
  CURRENT PATH           ===>                    (PRIMARY or ALTERNATE)
  PRIMARY CTC ADDRESS    ===>                    (Valid CTC address)
  PRIMARY PORT ADDRESS   ===>                    (1 or 2)
  ALTERNATE CTC ADDRESS  ===>                    (Valid CTC address)
  ALTERNATE PORT ADDRESS ===>                    (1 or 2)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 5. Library Define Panel

We then defined a 9247 drive in the library that was just defined (this step is not mandatory). This is defined as follows:

- Select option 2, **Optical Drive Configuration**, on the LIBRARY MANAGEMENT panel to get to the OPTICAL DRIVE APPLICATION SELECTION panel.

On this panel (Figure 6), specify:

- Name of your drive: DRIDUMMY.
- Device type: 9247.
- Select option 3, **Define a Drive**.
- Press the Enter key.

```
                    OPTICAL DRIVE APPLICATION SELECTION
COMMAND ===>

TO PERFORM DRIVE OPERATIONS, SPECIFY:

  CDS NAME          ===> 'SMS1.BASE.SCDS1'
                                (1 to 44 character data set name or 'ACTIVE')
  DRIVE NAME        ===> DRIDUMMY    (For Optical Drive List, fully or
                                      partially specified or * for all)
  DRIVE DEVICE TYPE ===> 9247        (For Optical Drive List, fully or
                                      partially specified or * for all)
SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

 1   LIST    - Generate a list of Drives
 2   DISPLAY - Display a Drive
 3   DEFINE  - Define a Drive
 4   ALTER   - Alter a Drive

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
   RESPECIFY SORT CRITERIA           ===> N  (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 6. Drive Application Selection Panel

Enter the drive definitions shown in Figure 7.

CTC AND SCSI address fields cannot be left blank.

```
                              9247 DRIVE DEFINE
COMMAND ===>

SCDS NAME:  SMS1.BASE.SCDS1
DRIVE NAME: DRIDUMMY

TO DEFINE DRIVE, SPECIFY:

  DESCRIPTION ===> Dummy drive 9247 for dummy library LIBDUMMY
              ===>

  LIBRARY NAME  ===> LIBDUMMY           (1 to 8 characters)
  DRIVE TYPE    ===> STDALONE           (LIBRARY or STDALONE)
  DRIVE NUMBER  ===>                    (0 to 3 or blank for STDALONE)
  ONLINE STATUS ===> N                  (Y or N)
  CTC ADDRESS   ===> 320                (Valid CTC address)
  SCSI ADDRESS  ===> 0                  (0 to 7)




 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 7. Drive Define Panel

5.4.6 Defining Storage Groups and Relating the Libraries

You now must define the object storage groups and relate them to the object databases previously defined in OAM. We related VIOBJB00 to GROUP00 and VIOBJB01 to GROUP01. We did not relate the other groups, GROUP02 to GROUP04, to any storage group. These groups are spare object storage groups not referenced in our ACS routines. If you need to use these groups, they are added later.

Proceed as follows for each group:

- From the ISMF main menu select **STORAGE GROUP**. The STORAGE GROUP APPLICATION SELECTION panel is shown. Figure 8.
- Enter VIOBJB00 in the storage group name field.
- Enter OBJECT in the storage group type field.
- Select **Define a Storage Group**.
- Press the Enter key.

```
                    STORAGE GROUP APPLICATION SELECTION
COMMAND ===>

TO PERFORM STORAGE GROUP OPERATIONS, SPECIFY:

  CDS NAME           ===> 'SMS1.BASE.SCDS1'
                          (1 to 44 character data set name or 'ACTIVE')
  STORAGE GROUP NAME ===> VIOBJB00            (for Storage Group List, fully or
                                               partially specified or * for all)
  STORAGE GROUP TYPE ===> OBJECT              (VIO, POOL, DUMMY, OBJECT,
                                               OBJECT BACKUP, or TAPE)
SELECT ONE OF THE FOLLOWING OPTIONS ===> 2

  1  LIST    - Generate a list of Storage Groups
  2  DEFINE  - Define a Storage Group
  3  ALTER   - Alter a Storage Group
  4  VOLUME  - Display, Define, Alter or Delete Volume Information

IF OPTION 1 CHOSEN ABOVE,
  RESPECIFY VIEW CRITERIA           ===> N   (Y or N)
  RESPECIFY SORT CRITERIA           ===> N   (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 8. Storage Group Selection Panel

The OBJECT STORAGE GROUP DEFINE panel is shown. (Figure 9). On this panel:

- Enter GROUP00 in the qualifier field.
- The value 20 is the cycle start time and 21 is the cycle end time.
- Press the Enter key.

```
                          OBJECT STORAGE GROUP DEFINE
COMMAND ===>

SCDS NAME:          SMS1.BASE.SCDS1
STORAGE GROUP NAME: VIOBJB00

TO DEFINE STORAGE GROUP, SPECIFY:
  DESCRIPTION ===> Primary object storage GROUP00
              ===>

  QUALIFIER        ===> GROUP00  (GROUP00-GROUP99)
  CYCLE START TIME ===> 20   (0-23 or NONE)  END TIME ===> 21 (0-23 or blank)

  LIBRARY NAMES  (1 to 8 characters each):
    ===>           ===>           ===>           ===>
    ===>           ===>           ===>           ===>
  VOLUME FULL THRESHOLD      ===>      (0-9999)
  DRIVE START THRESHOLD      ===>      (0-9999)
  VOLUME FULL AT WRITE ERROR ===>      (Y or N)

  DEFINE SMS STORAGE GROUP STATUS    ===> Y  (DEFINE - Y, ALTER - Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 9. Object Storage Group Define Panel

The SMS STORAGE GROUP STATUS DEFINE panel is shown (Figure 10):

- Enter ENABLE in the SMS SG STATUS field.

```
                        SMS STORAGE GROUP STATUS DEFINE
COMMAND ===>

SCDS NAME:           SMS1.BASE.SCDS1
STORAGE GROUP NAME:  VIOBJBOO
STORAGE GROUP TYPE:  OBJECT

TO DEFINE STORAGE GROUP SYSTEM STATUS, SPECIFY:

    SYSTEM        SMS SG
    NAME          STATUS
   --------       ------
   BETH      ===> ENABLE               ( Possible SMS SG Status
             ===>                        values for each System
             ===>                        - Pool SG Type
             ===>                          NOTCON, ENABLE, DISALL,
             ===>                          DISNEW, QUIALL, or QUINEW
             ===>                        - Object, Object Backup, or
             ===>                          Tape SG Type
             ===>                          NOTCON, ENABLE, DISALL,
                                           or DISNEW )

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 10. SMS Storage Group Status Define Panel

Note: We repeated the same steps for GROUP01. No definitions are required for any of the other groups.

5.4.7 Defining Backup Storage Groups and Relating the Libraries

An Object Backup Storage group must also be defined. Proceed as follows:

- Select **STORAGE GROUP**. on the ISMF main menu. The STORAGE GROUP APPLICATION SELECTION panel is shown (Figure 11).
- Enter VIOBJBUP in the storage group name field and OBJECT BACKUP in the storage group type field.
- Select **Define a Storage Group** and press the Enter key.

```
                    STORAGE GROUP APPLICATION SELECTION
COMMAND ===>

TO PERFORM STORAGE GROUP OPERATIONS, SPECIFY:

  CDS NAME           ===> 'SMS1.BASE.SCDS1'
                          (1 to 44 character data set name or 'ACTIVE')
  STORAGE GROUP NAME ===> VIOBJBUP           (for Storage Group List, fully or
                                              partially specified or * for all)
  STORAGE GROUP TYPE ===> OBJECT BACKUP      (VIO, POOL, DUMMY, OBJECT,
                                              OBJECT BACKUP, or TAPE)
SELECT ONE OF THE FOLLOWING OPTIONS ===> 2

  1  LIST    - Generate a list of Storage Groups
  2  DEFINE  - Define a Storage Group
  3  ALTER   - Alter a Storage Group
  4  VOLUME  - Display, Define, Alter or Delete Volume Information

IF OPTION 1 CHOSEN ABOVE,
  RESPECIFY VIEW CRITERIA           ===> N   (Y or N)
  RESPECIFY SORT CRITERIA           ===> N   (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 11. Storage Group Application Selection Panel

The OBJECT BACKUP STORAGE GROUP DEFINE panel is shown (Figure 12). Since no optical library was available or tape device was available for backup, we specified our DUMMY library.

```
                      OBJECT BACKUP STORAGE GROUP DEFINE
COMMAND ===>

SCDS NAME:          SMS1.BASE.SCDS1
STORAGE GROUP NAME: VIOBJBUP

TO DEFINE STORAGE GROUP, SPECIFY:

  DESCRIPTION ===> OBJECT STORAGE BACKUP GROUP00
              ===>

  LIBRARY NAMES  (1 to 8 characters each):
    ===> LIBDUMMY  ===>           ===>           ===>
    ===>           ===>           ===>           ===>

  VOLUME FULL THRESHOLD      ===> 100  (0-9999)
  DRIVE START THRESHOLD      ===> 200  (0-9999)
  VOLUME FULL AT WRITE ERROR ===> Y    (Y or N)

  DEFINE SMS STORAGE GROUP STATUS    ===> Y  (DEFINE - Y, ALTER - Y or N)


 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 12. Object Storage Group Define Panel

We set the status as DISALL (disabled for all access) on the SMS STORAGE GROUP STATUS DEFINE panel (Figure 13).

```
                        SMS STORAGE GROUP STATUS DEFINE
COMMAND ===>

SCDS NAME:           SMS1.BASE.SCDS1
STORAGE GROUP NAME:  VIOBJBUP
STORAGE GROUP TYPE:  OBJECT BACKUP

TO DEFINE STORAGE GROUP SYSTEM STATUS, SPECIFY:

    SYSTEM        SMS SG
    NAME          STATUS
   --------       ------
   BETH      ===> DISALL               ( Possible SMS SG Status
             ===>                        values for each System
             ===>                        - Pool SG Type
             ===>                          NOTCON, ENABLE, DISALL,
             ===>                          DISNEW, QUIALL, or QUINEW
             ===>                        - Object, Object Backup, or
             ===>                          Tape SG Type
             ===>                          NOTCON, ENABLE, DISALL,
                                           or DISNEW )

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 13. SMS Storage Group Status Define Panel

5.4.8 Defining Storage Classes

Define the storage classes you intend to use. We only defined one, DB2DASD, for storage of objects on DASD. More storage classes are added later.

- Select **STORAGE CLASS**. on the ISMF main menu.
- Enter DB2DASD in the storage class name field. See Figure 14 on page 44.
- Select **DEFINE** and press the Enter key.

```
                    STORAGE CLASS APPLICATION SELECTION
COMMAND ===>

TO PERFORM STORAGE CLASS OPERATIONS, SPECIFY:

  CDS NAME           ===> 'SMS1.BASE.SCDS1'
                              (1 to 44 character data set name or 'ACTIVE')

  STORAGE CLASS NAME ===> DB2DASD        (For Storage Class List, fully or
                                          partially specified or * for all)

SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

 1   LIST    - Generate a list of Storage Classes
 2   DISPLAY - Display a Storage Class
 3   DEFINE  - Define a Storage Class
 4   ALTER   - Alter a Storage Class

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
   RESPECIFY SORT CRITERIA           ===> N  (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 14. Storage Class Selection Panel

- The STORAGE CLASS DEFINE panel is shown (Figure 15).

```
                              STORAGE CLASS DEFINE
COMMAND ===>

SCDS NAME:          SMS1.BASE.SCDS1
STORAGE CLASS NAME: DB2DASD

TO DEFINE STORAGE CLASS, SPECIFY:
  DESCRIPTION ===> Storage class for objects on DASD
              ===>

  PERFORMANCE OBJECTIVES
   DIRECT MILLISECOND RESPONSE     ===>             (1 to 999 or blank)
   DIRECT BIAS                     ===>             (R, W or blank)
   SEQUENTIAL MILLISECOND RESPONSE ===>             (1 to 999 or blank)
   SEQUENTIAL BIAS                 ===>             (R, W or blank)
   INITIAL ACCESS RESPONSE SECONDS ===> 0           (0 to 9999 or blank)
   SUSTAINED DATA RATE (MB/SEC)    ===>             (0 to 999 or blank)
  AVAILABILITY                     ===> STANDARD    (STANDARD or CONTINUOUS)
  ACCESSIBILITY                    ===> P           (C, P or S)
  GUARANTEED SPACE                 ===> N           (Y or N)
  GUARANTEED SYNCHRONOUS WRITE     ===> N           (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 15. Storage Class Define Panel

- The initial access response must be 0 to force the objects to DASD.

  Note: If the initial access response is set to a value other than 0, OAM attempts to store the object directly to optical. An error occurs indicating that there is no eligible optical drive available to satisfy the request.

- For guaranteed synchronous write, we entered N. Everything else was ignored.

5.4.9 Defining Management Classes

Two management classes, OBJ20 and OBJ365, were defined. A more detailed description of these classes is found later in this chapter.

- Select **MANAGEMENT CLASS**. on the ISMF main menu. The panel in Figure 16 is shown.
- Enter the management class name.
- Select **DEFINE** and press the Enter key.
- You must then define the management class in more detail. Be aware that there are five panels and pressing enter does not take you to the next panel. You must use the Down command or F8.

See Figure 17 to Figure 20.

```
                    MANAGEMENT CLASS APPLICATION SELECTION          Page 1 of 2
COMMAND ===>

TO PERFORM MANAGEMENT CLASS OPERATIONS, SPECIFY:

  CDS NAME              ===> 'SMS1.BASE.SCDS1'
                                 (1 to 44 character data set name or 'ACTIVE')

  MANAGEMENT CLASS NAME ===> OBJ20        (For Management Class List, fully or
                                           partially specified or * for all)

SELECT ONE OF THE FOLLOWING OPTIONS ===> 3

  1  LIST    - Generate a list of Management Classes
  2  DISPLAY - Display a Management Class
  3  DEFINE  - Define a Management Class
  4  ALTER   - Alter a Management Class

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA          ===> N    (Y or N)
   RESPECIFY SORT CRITERIA          ===> N    (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 16. Management Class Application Selection Panel

```
                          MANAGEMENT CLASS DEFINE                  Page 1 of 5
COMMAND ===>

SCDS NAME:              SMS1.BASE.SCDS1
MANAGEMENT CLASS NAME:  OBJ20

TO ALTER MANAGEMENT CLASS, SPECIFY:

  DESCRIPTION ===> DELETE OBJECTS AFTER 20 DAYS OF NON-USAGE
              ===> OR 60 DAYS IN THE SYSTEM

  EXPIRATION ATTRIBUTES
    EXPIRE AFTER DAYS NON-USAGE ===> 20          (1 to 9999 or NOLIMIT)
    EXPIRE AFTER DATE/DAYS      ===> 60          (0 to 9999, yyyy/mm/dd
                                                  or NOLIMIT)

  RETENTION LIMIT               ===> NOLIMIT     (0 to 9999 or NOLIMIT)




 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 17. Management Class Define Panel (1 of 5)

```
                          MANAGEMENT CLASS DEFINE                  Page 2 of 5
COMMAND ===>

SCDS NAME:              SMS1.BASE.SCDS1
MANAGEMENT CLASS NAME:  OBJ20

TO ALTER MANAGEMENT CLASS, SPECIFY:

  PARTIAL RELEASE               ===> N           (Y, C, YI, CI or N)

  MIGRATION ATTRIBUTES
    PRIMARY DAYS NON-USAGE      ===>             (0 to 9999 or blank)
    LEVEL 1 DAYS NON-USAGE      ===>             (0 to 9999, NOLIMIT or blank)
    COMMAND OR AUTO MIGRATE     ===> NONE        (BOTH, COMMAND or NONE)


  GDG MANAGEMENT ATTRIBUTES
    # GDG ELEMENTS ON PRIMARY  ===>             (0 to 255 or blank)
    ROLLED-OFF GDS ACTION       ===>             (MIGRATE, EXPIRE or blank)



 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 18. Management Class Define Panel (2 of 5)

```
                          MANAGEMENT CLASS DEFINE                 Page 3 of 5
COMMAND ===>

SCDS NAME:              SMS1.BASE.SCDS1
MANAGEMENT CLASS NAME:  OBJ2O

TO DEFINE MANAGEMENT CLASS, SPECIFY:

  BACKUP ATTRIBUTES
    BACKUP FREQUENCY              ===>         (0 to 9999 or blank)
    NUMBER OF BACKUP VERS         ===>         (1 to 13 or blank)
      (DATA SET EXISTS)
    NUMBER OF BACKUP VERS         ===>         (0 to 13 or blank)
      (DATA SET DELETED)
    RETAIN DAYS ONLY BACKUP VER   ===>         (1 to 9999, NOLIMIT or blank)
      (DATA SET DELETED)
    RETAIN DAYS EXTRA BACKUP VERS ===>         (1 to 9999, NOLIMIT or blank)
    ADMIN OR USER COMMAND BACKUP  ===> NONE    (BOTH, ADMIN or NONE)
    AUTO BACKUP                   ===> N       (Y or N)
    BACKUP COPY TECHNIQUE         ===> S       (P=Conc Preferred,
                                                R=Conc Required or S=Standard)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 19. Management Class Define Panel (3 of 5)

Note: If AUTO BACKUP = N, you must change it to Y when you begin using the OSMC cycle and transition objects to optical. Otherwise, no optical backup copy is made.

```
                          MANAGEMENT CLASS DEFINE                 Page 4 of 5
COMMAND ===>

SCDS NAME:              SMS1.BASE.SCDS1
MANAGEMENT CLASS NAME:  OBJ2O

TO ALTER MANAGEMENT CLASS, SPECIFY:

  OBJECT CLASS TRANSITION CRITERIA
    TIME SINCE CREATION YEARS  ===>          (0 to 9999 or blank)
                       MONTHS ===>          (0 to 9999 or blank)
                       DAYS   ===>          (0 to 9999 or blank)
    TIME SINCE LAST USE YEARS  ===>          (0 to 9999 or blank)
                       MONTHS ===>          (0 to 9999 or blank)
                       DAYS   ===>          (0 to 9999 or blank)
    PERIODIC
      MONTHLY   ON DAY   ===>           (1 to 31, FIRST, LAST or blank)
      QUARTERLY ON DAY   ===>           (1 to 92, FIRST, LAST or blank)
                IN MONTH ===>           (1 to 3 or blank)
      YEARLY    ON DAY   ===>           (1 to 366, FIRST, LAST or blank)
                IN MONTH ===>           (1 to 12 or blank)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 20. Management Class Define Panel (4 of 5)

```
                          MANAGEMENT CLASS ALTER                  Page 5 of 5
COMMAND ===>

SCDS NAME:              SMS1.BASE.SCDS1
MANAGEMENT CLASS NAME:  OBJ20

TO ALTER MANAGEMENT CLASS, SPECIFY:

  AGGREGATE BACKUP ATTRIBUTES:

    # VERSIONS             ===>          (1 to 9999, NOLIMIT or blank)
    RETAIN ONLY VERSION    ===>          (1 to 9999, NOLIMIT or blank)
      UNIT                 ===>          (D=days, W=weeks, M=months, Y=years
                                          or blank)
    RETAIN EXTRA VERSIONS  ===>          (1 to 9999, NOLIMIT or blank)
      UNIT                 ===>          (D=days, W=weeks, M=months, Y=years
                                          or blank)
    COPY SERIALIZATION     ===>          (C=continue, F=fail or blank)
    ABACKUP COPY TECHNIQUE ===> S        (P=Conc Preferred,
                                          R=Conc Required or S=Standard)


 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 21. Management Class Define Panel (5 of 5)

Just two management classes were defined, OBJ20 and OBJ365. Because we only had DB2DASD in our storage hierarchy, there was no need to define class transitions. The following properties were planned:

- OBJ20 - Object eligible for deletion after 20 days of non-usage or 60 days in the system.
- OBJ365 - Object eligible for deletion after 365 days of non-usage or 1000 days in the system.

To achieve this, the parameters shown in Table 12 on page 49 were specified.

,

Table 12. Management Class Attributes		
Parameter name	**Values**	
Management Class	OBJ20	OBJ365
Expire non-usage	20	365
Expire date/days	60	1000
Retention Limit	NOLIMIT	NOLIMIT
Partial Release	No	No
Primary Days	<blank>	<blank>
Level 1 Days	<blank>	<blank>
CMD/AUTO Migrate	NONE	NONE
# GDG on Primary	<blank>	<blank>
Rolled-off GDG	<blank>	<blank>
Backup Frequency	<blank>	<blank>
# Backup (exists)	<blank>	<blank>
# Backup (delete)	<blank>	<blank>
Retain only bkup	<blank>	<blank>
Retain extra bkup	<blank>	<blank>
Admin/user bakup	NONE	NONE
Auto Bck technique	NO	NO
Backup copy tech	NO	NO
CTRANS creat yrs	<blank>	<blank>
CTRANs creat mths	<blank>	<blank>
CTRANS creat days	<blank>	<blank>
CTRANS used yrs	<blank>	<blank>
CTRANS used mths	<blank>	<blank>
CTRANS used days	<blank>	<blank>
CTRANS abs mthly	<blank>	<blank>
CTRANS abs qu day	<blank>	<blank>
CTRANS abs qu mth	<blank>	<blank>
CTRANS abs yr day	<blank>	<blank>
CTRANS abs yr mth	<blank>	<blank>

5.4.10 Defining Automatic Class Selection

The final step is to create and validate ACS routines. You need to supply routines for storage classes, management classes, and storage groups. These allow VisualInfo to manage the migration of your data from expensive media to less expensive media.

Place your ACS routines into a data set. We used VIUSR1.INSTALL.CNTL. We created three members: VIACSMC, VIACSSC and VIACSSG, based on the samples described in the topic “Automatic Class Selection” on page 335 of the *OAM Planning, Installation and Storage Administration Guide for Object Support*.

The required steps are as follows:

- Select **AUTOMATIC CLASS SELECTION** on the ISMF main menu.
- Create your routines.
- Translate the finished ACS routines into object format.
- Validate the routines.

  Note: You might get message IGD06023I stating that a storage group is not referenced by the storage group ACS routine. Ignore that message if you are not going to make use of those storage groups.
- Test the routines.

The routines we coded are the minimum required to satisfy the verification for the VisualInfo Object Server. Examples of the routines used are given below:

- Management Class selection routine

```
PROC MGMTCLAS
FILTLIST COLL_1 INCLUDE('VIHOST.CLLCT001')
FILTLIST COLL_2 INCLUDE('VIHOST.CLLCT002')
IF &ACSENVIR = 'STORE'  THEN
  SELECT
    WHEN (&DSN = &COLL_1)
      SET &MGMTCLAS = 'OBJ20'
    OTHERWISE
      SET &MGMTCLAS = 'OBJ365'
  END
END
```

- Storage Class selection routine

```
PROC STORCLAS
IF &ACSENVIR = 'STORE' THEN
 SET &STORCLAS = 'DB2DASD'
END
```

- Storage Group selection routine

```
PROC STORGRP
FILTLIST COLL1   INCLUDE('VIHOST.CLLCT001')
IF &ACSENVIR = 'STORE'  THEN
  SELECT
    WHEN (&DSN = &COLL1)
      SET &STORGRP = 'VIOBJB00'
    OTHERWISE
      SET &STORGRP = 'VIOBJB01'
  END
END
```

At this point you should decide how to handle the high-level qualifiers of the collection names. Most installations prevent users from writing directly to the Master Catalog (through RACF protection). Consider setting up a user catalog for high-level qualifiers (HLQ). An OAM store request fails on the first use of a collection name if that name cannot be placed in a catalog because it is RACF protected. If you do not create an alias pointing to a user catalog, then you must permit UPDATE authority to the Object Server CICS region using:

```
PERMIT CATALOG('mcatname')  ID(XXX) ACCESS(CONTROL)
```

where XXX is the value of your Object Server region in your installation.

5.4.11 Validating and Activating the Configuration

Use ISMF option 8 to validate and then activate your SMS configuration. Activation of the configuration causes the following WTOR to be issued on the operator's console:

```
00 IGD034D REPLY 'Y' TO ALLOW ACTIVATION OF A CONFIGURATION BY USERID OR
'N' TO DENY THE REQUEST
```

A response of Y causes the following message to be issued:

```
IGD008I NEW CONFIGURATION ACTIVATED FROM SCDS 'SMS1.BASE.SCDS1'
```

Note: The following paragraph is taken from *IBM SAA ImagePlus MVS/ESA Installation and Customization Guide Version 2.1.1* and may be of value to you.

"Although an SMS configuration may successfully test and validate, a successful store may not follow. Note that if, in the ACS routine "&ACSENVIR='STOR'" instead of "&ACSENVIR='STORE'" is used, the test and validation succeeds, but the store fails. An EKC0703E error with these values is returned: EKC0703E return_code=20 reason_code=00 external_return_code=12
external_reason_code=A0020400. See INFO APAR II05257."

5.4.12 Verification of OAM Installation

Now that your SMS configuration is completed and activated, it is possible to further verify the installation of OAM and the ACS routines. This entails driving the OSREQ macro in TSO batch by creating a sample verification program as discussed under topic "Installation Verification Program" on page 350 of the *OAM Planning, Installation and Storage Administration Guide for Object Support*.

We created an IVP job called OAMIVP to do the verification.

5.4.13 Additional Steps (Optional)

If, in your MVS IOCP definitions, you do have CTCs defined for optical libraries, you can add a REAL library. Refer to topic B.2, "Defining Optical Libraries" on page 101 for information on using ISMF to define real libraries.

You still need to define at least one PSEUDO library even if you do not have any optical devices, and if you want the OAM address space to be active.

Chapter 6. Installing a VisualInfo Client on OS/2

This chapter lists the steps for installing an OS/2 VisualInfo client system that is connected to VisualInfo MVS/ESA library and object servers. These steps assume that a separate workstation is used for the configuration server and a separate chapter is provided for that installation. However, both the client and the configuration server can reside on the same workstation. Currently, only an OS/2 platform is supported for these two components. A Windows client is planned in the future.

It is also assumed that all of the prerequisite software and hardware are in place.

6.1 Software Prerequisites

In order to support a VisualInfo client, you must install the following software:

- OS/2 V2.1 or above
- Communications Manager/2 V1.1 or above
- One of the following, compatible with your CM/2 release level:
  - Network Transport Services/2 (NTS/2)
    or
  - LAN Adapter and Protocol Support (LAPS)

If your client workstation does not have a CD-ROM drive, you also need software to access the installation files from another workstation.

Be sure to have the latest code level and APARs installed. Otherwise you could run into problems with old code which have been fixed in the latest release. For example, APAR PN66298 should be installed, or a release of code which incorporates it.

6.2 Accessing the CD-ROM

VisualInfo software is shipped on a CD-ROM. There are several ways to access a CD-ROM drive from your workstation:

- Attach an external or internal CD-ROM drive to your workstation.
- Use network manager code such as LAN Server to define the CD-ROM as a redirected drive from another workstation. This is known as a pull install.
- Use a download utility, such as Software Installer/2 with NetView/DM, to download from another workstation. This is known as a push install. Software Installer/2 is included with the VisualInfo product.

For more information on installing VisualInfo using these methods, see *VisualInfo Systems Management: CID Installation*.

We are using LAN Server for this chapter. You need LAN Server software on a server with a CD-ROM drive, and LAN Requester software on your client workstation.

6.3 Preparing Your Workstation

Before you can install, you must follow the steps described in detail below.

- Define the CD-ROM drive as a re-directed drive from the LAN server to the workstation.
- Log on from the client workstation as a requester to the code server.
- If you have a Showcase CD, run VisualInfo Setup to prepare the workstation for the code.
- On a Showcase CD, unlock the modules you want to install. This process is documented in the README.ENG file. You also need a customer number, plus an access key for each product to be installed.
- Optionally, copy the code directly to a hard drive on your workstation or server. This speeds up the installation process, especially if you might need to do it again.

6.3.1 Defining a Re-directed Drive

Since we have assumed a LAN Server environment, you can perform these steps on your code server. They give your client workstation access to the code using a LAN Requester connection. The VisualInfo code can reside on a CD-ROM drive or a hard drive in the server. It is available to the client workstation as a re-directed drive.

- Install LAN Server on your code server (the workstation with the CD-ROM drive), and then start it by typing `net start srv`.
- From an OS/2 Full Screen prompt, type `net`.
- From the Definitions pull down, select **Aliases**, then **Files**.
- Select **New**, then **Create**. You are now creating an alias for a re-directed (network) drive.
- Fill in the panel as follows:
  - Type an alias name such as `CD-ROM`.
  - Type a descriptive comment such as `Entire CD`.
  - Type the name of the server.

    Note: To find this name, go to an OS/2 prompt on the server and type:
    `type ibmlan\ibmlan.ini |more`
    Look for the Computer name of the server and use it for the server name in your LAN Server alias definition.
    Also note the Domain name so you can log on to it later. See Figure 22.

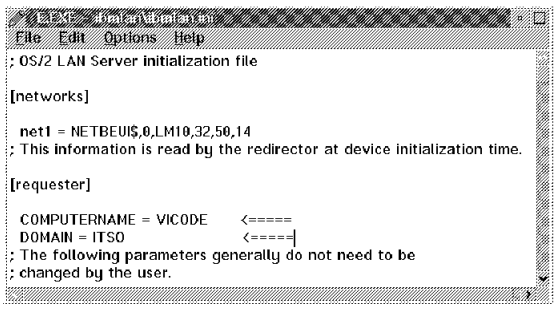


Figure 22. Finding Your Computer name and Domain Name

  - Type the root path of the CD, such as F:\.
  - Leave the maximum number of users blank.
  - Select **When shared** to be At server startup.
  - Press the Enter key when you have filled in all of the fields.
- Select the new alias with the space bar, and from the Access Profile pull-down, select **Create**.
- Give Permissions of XR (run the program and read) and press the Enter key.
- With the alias still highlighted, from the Access Profile pull-down, select **Apply** and press the Enter key to proceed. This applies the XR permissions to all sub-directories under the alias directory. It may take awhile for this to complete.
- Exit back to the main panel.
- From the Definitions pull-down, select **Users**.
- For each User ID accessing the alias, press the Enter key and select **File Assignments**.
- Place an asterisk or a drive letter, such as R, under the Drive heading for the alias. This assigns a drive letter to each user ID that logs onto the LAN server. You can switch to that drive letter (representing the CD-ROM root directory) to install the VisualInfo code.
- Exit out of the LAN Requester panels into the full screen.
- Stop the LAN server (`net stop server`) and restart it (`net start srv`).

6.3.2 Logging On to the Server from the Requester

You are now ready to walk over to your client workstation and log onto the code server as a LAN requester. The code server provides you with an additional drive (known as a re-directed drive) that contains the VisualInfo code. This re-directed drive is either the CD-ROM drive with the VisualInfo CD, or the subdirectory where the VisualInfo code is installed.

- From an OS/2 prompt on the client workstation, enter `logon /d`. When prompted, fill in the user ID and password of a User Profile Management ID (UPM user ID) with Administrator authority (the defaults are user ID and PASSWORD). Ensure that the Domain name on the Logon panel matches the Domain name in the \IBMLAN\IBMLAN.INI file on the server. See Figure 23.

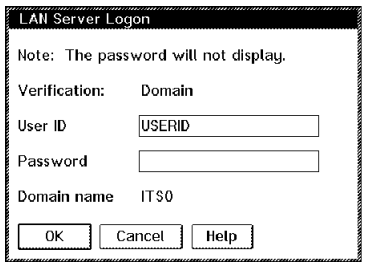


Figure 23. Logging On to LAN Requester

- When you see the message `Logon is successful`, enter `net use`. The alias or aliases you defined are displayed, along with the re-directed drive letter or letters, as shown in the following figure.

```
C:\ net use

Status          Local name      Remote name
_____________________________________________

OK              R:              \\VICODE\CDROM
The command completed successfully.
```

Figure 24. Using Net Use to Display LAN Resources

- Switch to the CD-ROM drive by typing its drive letter and a colon, for example, R:. Now you are ready for Setup.

6.3.3 Running Setup on Your Workstation

VisualInfo is usually packaged on a CD in a format called CD Showcase. For more information on installing VisualInfo from a Showcase CD, see *A Simple Approach to VisualInfo*. Before you install the code, you must run Setup from the Showcase to prepare your workstation.

- From an OS/2 prompt on the re-directed drive (CD-ROM), type showcase.

  Note: If the Showcase command is not found, your CD may not be in Showcase format. If not, go to topic 6.4, "Installing the Client Component" on page 57. When you see the Showcase command, skip over it and perform the next task.

- Select **Continue**.
- Select the **Setup** icon from the bottom of the Showcase panel. See Figure 25 on page 58.
- Accept the default path of C:\SHOWCASE, or modify it as desired.
- When setup is complete, choose the **Exit** icon. Shut down and re-boot your system.

  Note: Shut down is an option on the Desktop pop-up menu, accessed by clicking on the right mouse button from an open area on the Desktop. An OS/2 system should always be cleanly shut down before re-booting. This allows buffers to be written to disk and applications to complete their processing.

6.3.4 Unlocking the VisualInfo Components

Many of the components on a VisualInfo CD need to be registered before they are installed. This registration process is documented in the README.ENG file. This process gives you an access key for each product you want to install.

It is a good idea is to print all of the README files. Type dir readme*.* /s to find where they are. They have many important hints that may not be in the documentation.

After you have your customer number and access keys ready, follow these steps:

- From the CD alias drive, type Showcase.
- Select **Continue**.

- From the list of products, click on **VisualInfo client**. See Figure 25 on page 58.
- Select **Details** from the same panel.
- Select **Unlock**. See Figure 26 on page 58.
- Select **OK**.
- Select **Yes** to update your config.sys file.
- Select **OK**.
- Type in your customer number and client access key.
- Select **Save** and **OK**.

Your component is now unlocked and available to install.

6.3.5 Copying the Code to Your Hard Drive (optional)

It is often faster to copy the code from the CD to your hard drive and install the product from the copy. This is an optional step. Showcase provides a way to do this:

- From the Showcase notebook, select **Install**. See Figure 26 on page 58.
- From the right side of the install page, select **Copy**. When you are prompted, enter the name of the directory where the code should be placed (for example, VICLIENT). The copying process takes some time, but not as long as installing directly from the CD.
- When the copying is complete, exit from Showcase, shut down and re-boot.

6.4 Installing the Client Component

After re-booting, you are ready to install the VisualInfo client. Make sure you have printed all of the README files. Type `dir readme*.* /s`. This tells you all of the directories that contain a README file. Then type `print filename`, where *filename* is the full name of the README file. This sends the file to your OS/2 default printer. Now you are ready to install.

- If your VisualInfo code resides on a code server, go to the client workstation and log onto the server. See topic 6.3.2, “Logging On to the Server from the Requester” on page 55.
- Switch to the drive (re-directed LAN Requester alias, such as R:) or subdirectory on your workstation (C:\VICLIENT) where the installation code resides. Type `net use` to obtain which drive letter you need.
- If you are not using Showcase, type `install` and press the Enter key. Then skip to the next bullet and click on **Continue**. If your installation uses a Showcase CD, do the following tasks.
  - Type `Showcase`.
  - Double-click on the Client component. See Figure 25 on page 58.
  - Select **Install** in the notebook. See Figure 26 on page 58.
  - Select **Install** on the right margin.

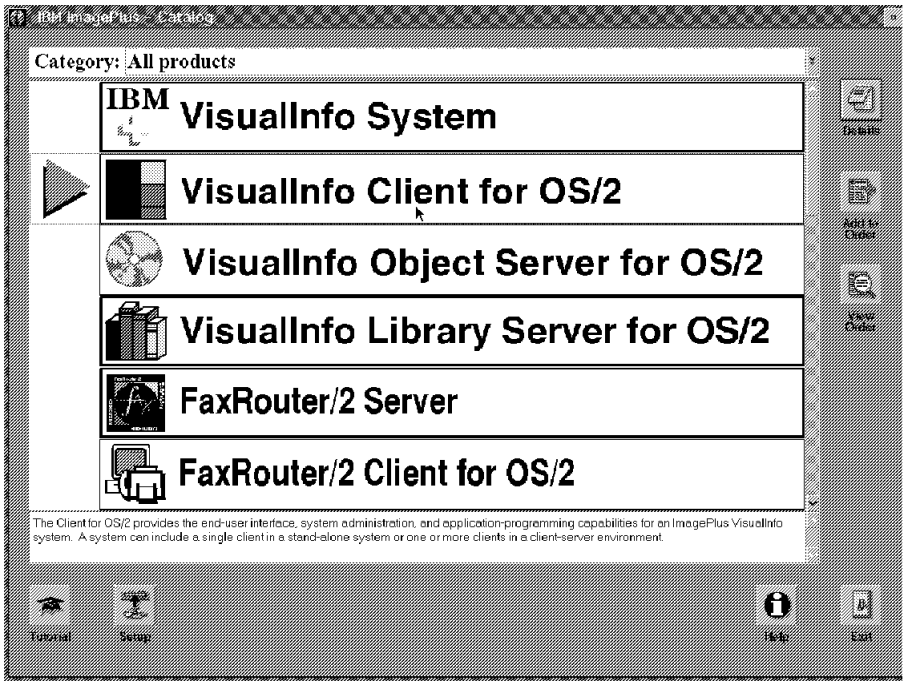


Figure 25. Selecting the Client Installation

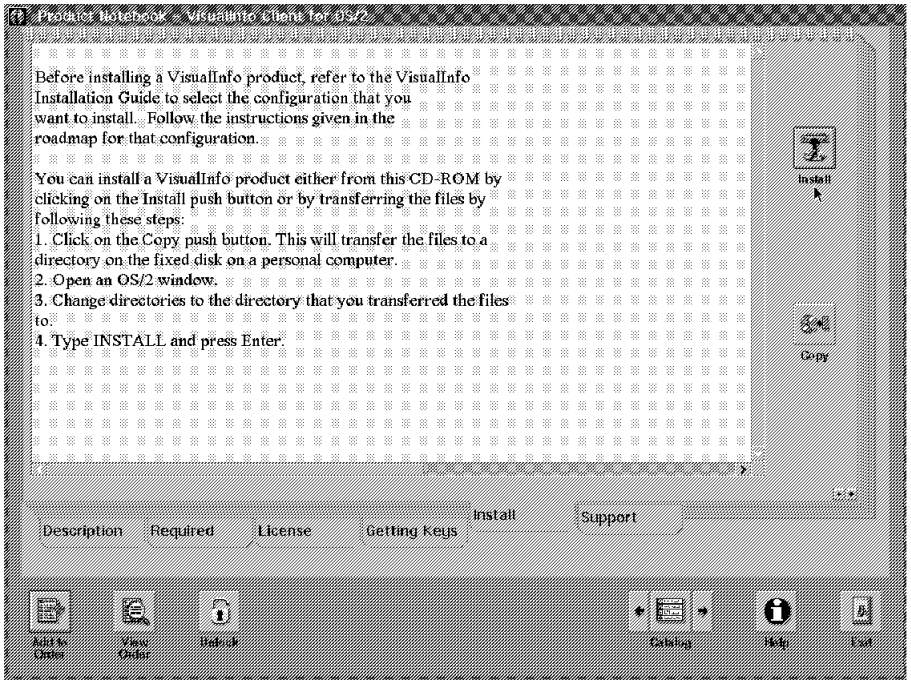


Figure 26. Selecting Install from the Product Notebook

- Click on **Continue**.
- From the Action pull-down menu, select **Install**.

  Note: If VisualInfo is already installed, you should delete the old files at this point by selecting Delete. After they are deleted, you may continue by selecting Install. Always use this method to delete the VisualInfo files if possible. If this method is unsuccessful, see the README file titled ″VisualInfo General Information″ for which files to delete manually.

See Figure 27 on page 59.

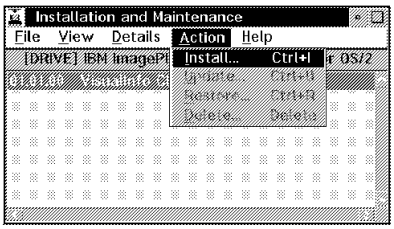


Figure 27. Selecting Install from the Action Pull-down

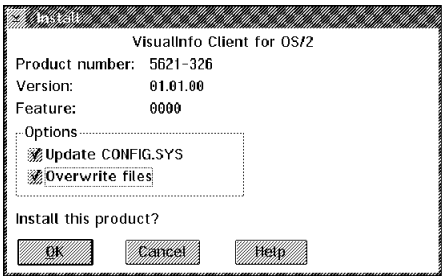


Figure 28. Overwriting Files

- Select **Overwrite files**, then select **OK**. See Figure 28.
- Select all of the components. After selecting your components, you can use the disk space push button to check space on other drives. You can select a drive other than C, (see Figure 29). Check the change directories box to the selected drive.

  Note: If you are conserving disk space and do not want all of the components, you should at least select the following:

  – System Configuration Utility
  – Library Client
  – Client Application
  – IBM Image Services

  See Figure 30 on page 60.

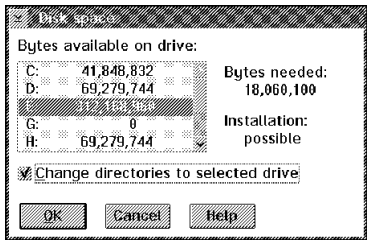


Figure 29. Selecting a Drive to Receive the Code

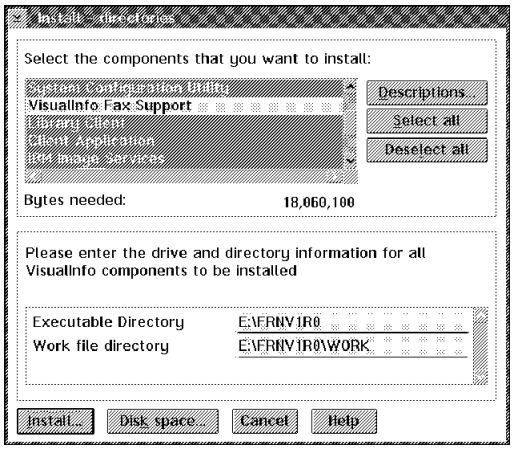


Figure 30. Selecting VisualInfo Components

- Select **Install**.
- Your progress is shown by a sliding bar. See Figure 31.
- When prompted by the Install Checklist, select **Install**. See Figure 32.

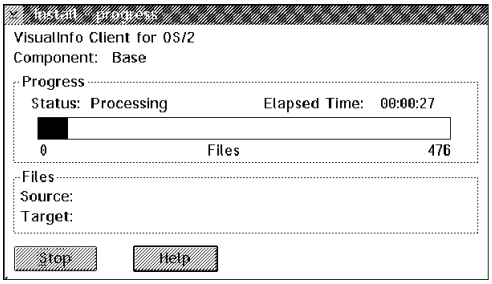


Figure 31. Checking Your Progress

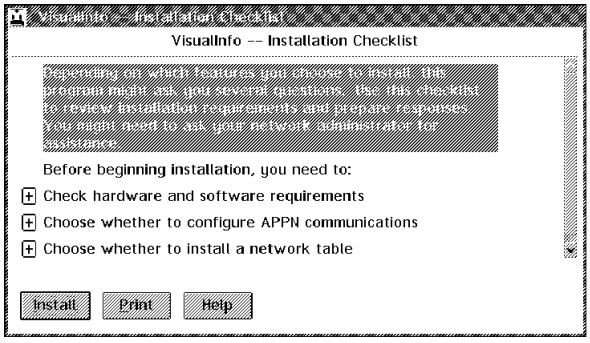


Figure 32. Checking Your Prerequisites

- At the next prompt, select **Specify a configuration file to be updated**, then select **FRNOCLNT**. See Figure 33 on page 61.

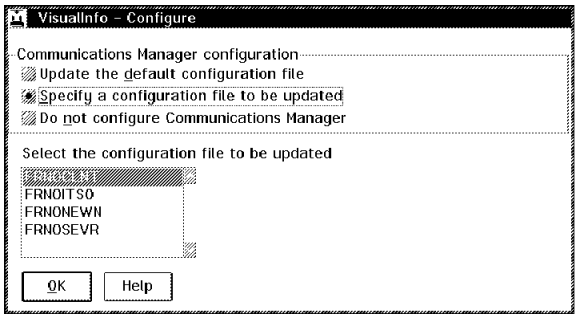


Figure 33. Modifying a CM/2 Configuration File

- Confer with your VTAM system programmer if needed, then fill out the fields of the configuration file as follows. Sample values for you to use are shown in parentheses. See also Figure 34.
  - Network ID is the name of your network (for example, USIBMSC).
  - Node name is the name (local control point) of your workstation (BEA0003).
  - Node ID is the ID number of your workstation (A0003).
  - Node type is for an APPN configuration (End node with Network Node).

    Note: This requires VTAM V4 or later.
  - Destination address is the token-ring address of your network node (for example, 400050002007; get this from your LAN administrator or VTAM system programmer. This is where your VisualInfo Configuration Server resides.
  - Adapter selects which LAN adapter card you are using (Primary).
  - Press **Configure**.
  - A new Communications Manager configuration (.CFG) file called FRNONEWN is created by the installation program. It tells you that after the installation is finished, you can stop and restart Communications Manager. See Figure 35 on page 62.

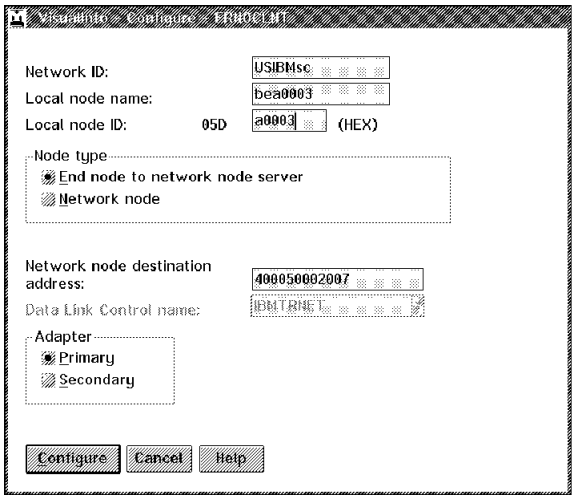


Figure 34. CM/2 Configuration File Parameters

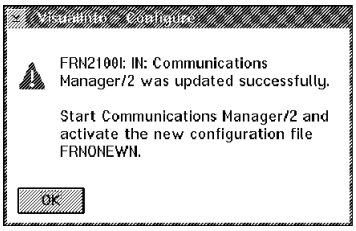


Figure 35. CM/2 File Updated

- When you are prompted, choose the option to install the default network table. See Figure 36.

You are shown a panel letting you know that the installation is complete. See Figure 37. At this time, you should exit from the File pull-down menu on each installation panel. Then shut down the system and re-boot to activate the changes to the CONFIG.SYS file.

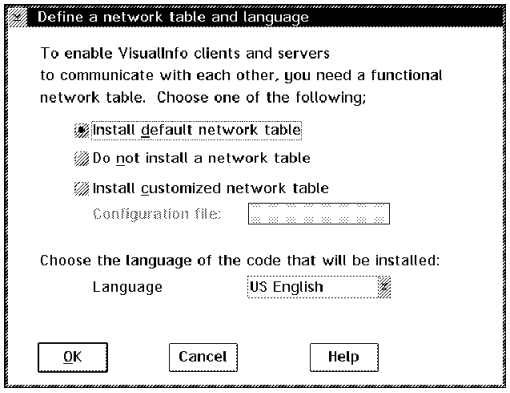


Figure 36. Choosing a Default Network Table

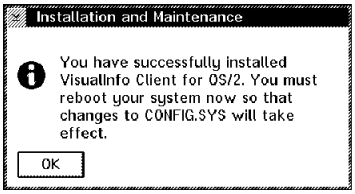


Figure 37. A Successful Installation

6.5 Configuring Communications Manager on the Client

When VisualInfo has been successfully installed, you can view and tailor the new Communications Manager/2 configuration file (FRNONEWN) that was created during installation. It is possible that no changes are needed at this time. Verify your CM/2 parameters against Figure 34 on page 61 while you do the following steps.

Note: If you are also installing a configuration server on this workstation, see topic 7.2, “Tailoring Communications Manager on the Configuration Server” on page 71 for the steps to follow.

- From the Communications Manager folder, open **Communications Manager Setup**.

- Choose **Setup**; select the FRNONEWN file that you created; select **OK**.
- Answer Yes (the configuration is used for this workstation).
- Select (click once) your Workstation Connection Type (Token-ring or other LAN types).
- Open (double-click) **APPC APIs** from the Feature or Application list.

6.5.1 SNA Local Node

- Open (select and configure) SNA local node characteristics.
- Ensure that your Network ID (USIBMSC in this example) and your workstation's local node name (BEA0003) are correct from the VisualInfo installation.
- Select the **Options** push button. We recommend that you make the *Local node alias name* the same as the local node name (BEA0003) from the previous panel. If you leave the field blank, the system fills the local node name in for you. You could add a different name for the alias, but you then need to remember it during other configuration steps.
- Press **OK** twice to return to the Communications Manager Profile List.

6.5.2 SNA Connections

- Select **SNA connections**, then select **To host**. This defines your connection from the configuration server to the MVS VisualInfo servers.
- Select **Create**, then **Continue**. Enter the values for these fields, using our sample data in the following list as a guide.
  - Link name: HOST0001
  - Activate at startup: Yes (Check box)
  - APPN support: Yes
  - Node ID: 05D A0003
  - LAN destination address: 400008210210
  - Address format: Token Ring
  - Remote SAP: 04
  - Use this host connection as your focal point support: Yes
- Press OK, then press Close to return to the Communications Manager Profile List.

6.5.3 SNA Features

- Select SNA features. Here you are creating an LU to LU session definition between your local LU and the MVS library and object servers (Partner LUs).

6.5.3.1 Local LU

- Select **Local LUs** and **Create**.
- Enter an LU name of BEA0003A. Name the alias the same for simplicity.
- Obtain your VTAM systems programmer if your LU was defined on the MVS system as independent or dependent. Select the correct radio button; if dependent, also type in the LU number (1 for this example).
- Choose HOST0001 as your Host link.
- Press **OK**.

6.5.3.2 Partner LUs

- Now select **Partner LUs** and **Create**.
- Here you create the connection to the host-based library server (HBLS). A fully qualified LU name consists of the network name followed by the application logical unit name (VTAM APPLID) that you are connecting to. You can get these from your VTAM system programmer. In our example, we typed in USIBMSC and IMLBRSV. We called the Alias IMLBRSV also, for simplicity. In our example, the MVS partner LU is independent; check with your system programmer. Press **OK**.
- Create the host-based object server (HBOS) in the same way. We used the same network name with an LU name/alias of IMOBRSV, and our MVS partner LU is independent.

6.5.3.3 Modes

- The .CFG file supplied during your VisualInfo client installation should have all of the modes you need. The MVS side of the session negotiates the mode parameters during the session connection (BIND). If your system programmer has coded another mode name from the host, you may want to define it here as well. Select **Modes**, **Create**, and fill in the desired parameters. Select **OK**.
- To save your SNA features, select **Close** three times. CM/2 verifies your configuration. If errors are found, see the next section.

  Note: You may need the CM/2 diskettes to add additional support to Communications Manager/2. If so, you are prompted at this time.

- If no errors are found, select **Close**, stop Communications Manager, restart it, and begin communicating with the host.

6.5.4 Handling Errors or Warnings

- If you have errors or warnings, you may get a Verify Configuration panel. To get this panel, you must have the Verify option on and FFST/2 started. Follow these instructions to correct the errors:
  - Press the **Show log** push button.
  - The Message Log Formatter shows you a list of messages, beginning with the most recent. You can double-click on any message to get more detail.
  - After you have read all of the messages and written down any user action needed, close the message log. Say Yes when asked if you are sure you want to exit.
- Press **Fix** to fix your errors, or **OK** to exit. Press **Close** until you are back to the OS/2 Desktop. Stop CM/2 and restart to put your changes into effect. If you are unable to get a connection, shut down, then power down and restart.

6.6 Verifying Your NTS (LAPS) Parameters

Communications Manager/2 must have certain network transport components in order to communicate on your LAN. Run LAPS or MPTS from the IBMCOM subdirectory to verify that you have the correct LAN support. See Figure 38 on page 65, lower left corner, for the correct support for our configuration.

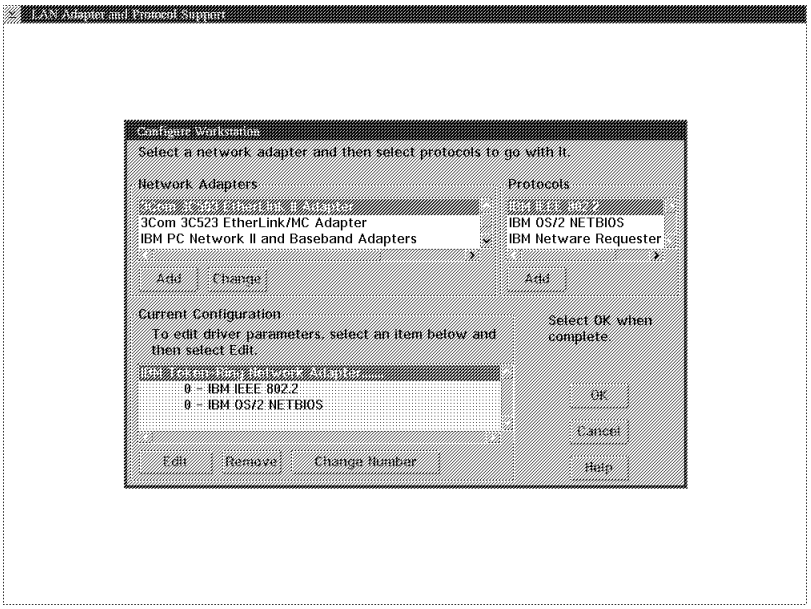


Figure 38. LAN Transport Parameters

If you want to change your LAN adapter card address, you can from this panel by editing the LAN adapter definition or one of the transport (802.2, NETBIOS) definitions. Be sure to shut down and power off your system after changing an address in order to remove the old address from the LAN.

6.7 Generating a VisualInfo Network Table

The VisualInfo program uses Communications Manager to send and receive information to and from other VisualInfo components. We defined the LUs to Communications Manager/2 in the previous steps. VisualInfo needs a table to know the names of the LUs that contain other components (object server, library server, configuration server, and so on). This table, called a Network Table, is named FRNOLINT.TBL. It is created using two VisualInfo utilities: the System Configuration Utility and the Network Table Generation Utility.

Open the VisualInfo folder and open the VisualInfo Utilities folder. See Figure 39.

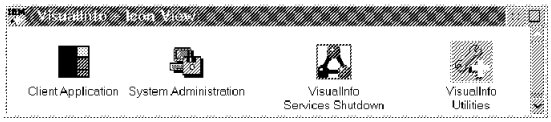


Figure 39. Opening the VisualInfo Utilities Folder

6.7.1 Running the System Configuration Utility

- Open the System Configuration Utility. See Figure 40 on page 66.
- Open the New Nodes object. See Figure 41 on page 66. This example contains four nodes: the client, the configuration server, the object server and the library server.

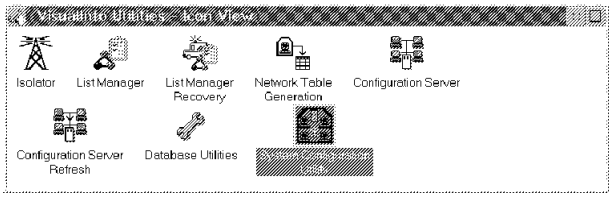


Figure 40. Selecting the System Configuration Utility

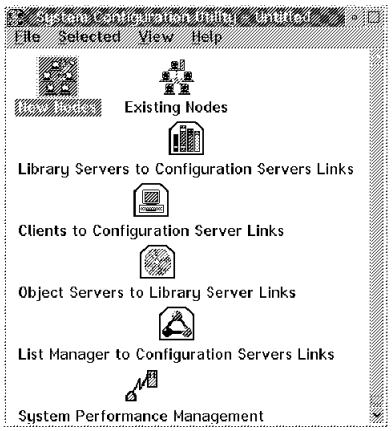


Figure 41. Selecting New Nodes

6.7.1.1 Client

The Node nickname is the local node alias name of the client workstation. We recommend naming the alias the same as the node name for simplicity. Our sample node name is BEA0003, so our nickname is also BEA0003. The operating system for this node is OS/2. The component installed in this node is the client (check the box). Select the APPN check box. Enter the network name (USIBMSC) and the node name (BEA0003).

Press the **Apply** push button and define the next node. Our sample values are provided. See Figure 42 on page 67.

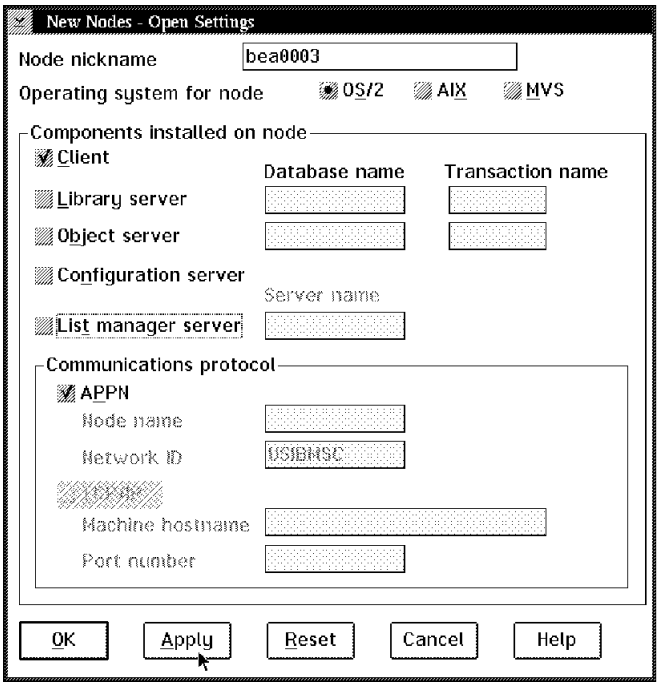


Figure 42. Defining a Client Node to VisualInfo

6.7.1.2 Configuration Server

- Node nickname: BEA0007
- Node name: BEA0007
- Network ID: USIBMSC
- Operating system for node: OS/2
- Components installed on node: Configuration server

Press **Apply**.

6.7.1.3 Object Server

- Node nickname: IMOBRSV
- Node name: IMOBRSV
- Network ID: USIBMSC
- Operating system for node: MVS
- Components installed on node: Object server
- Database name: HBOS0001 (any name will work)
- Transaction name: FRNO

Press **Apply**.

6.7.1.4 Library Server

- Node nickname: IMLBRSV
- Node name: IMLBRSV
- Network ID: USIBMSC
- Operating system for node: MVS
- Components installed on node: Library server
- Database name: VIUSR1
- Transaction name: FRNI

- Press **Apply**. When all of the nodes are entered, press **OK**.

We now tell VisualInfo which components are linked together.

- Open the Library Servers to Configuration Servers Links object. Add the Available library servers to the Assigned library servers column. Press OK. See Figure 43.

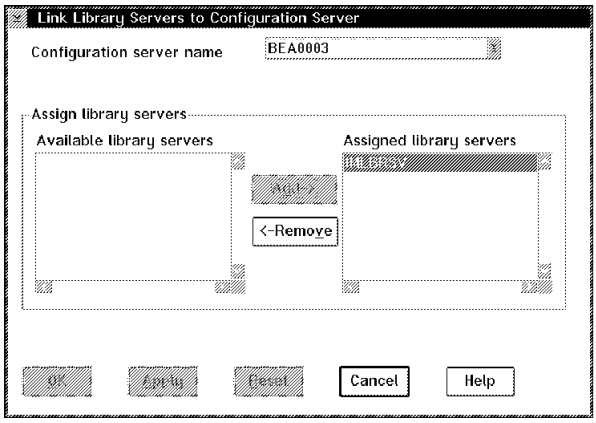


Figure 43. Defining Configuration Links

- Open the Clients to Configuration Server Links object. Add the Available clients to the Assigned clients column. Press **OK**.
- Open the Object Servers to Library Server Links object. Add the Available object servers to the Assigned object servers column. Press **OK**.
- From the File pull-down, select **Verify**. If your configuration does not verify, follow any message instructions and go back through the steps to correct your error.
- From the File pull-down, select **Save**. In the Save as filename box, replace the * with a name you can remember (for example, BEA0003.CFG). Press **OK**. See Figure 44. This file is your VisualInfo System Configuration file (not the CM/2 CFG file).

  Note: Remember to save your file and record the name. Otherwise, you must perform the configuration steps again.

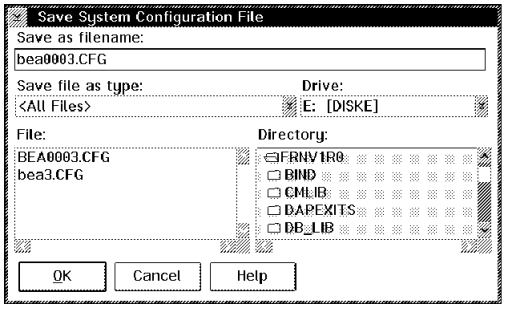


Figure 44. Saving the System Configuration File

- Exit from the System Configuration Utility.
- Later on, if you want to modify the configuration file after entering the utility, you would select **File**, then **Open**, then select your CFG file that you saved. Then you can select the **Existing Nodes** object, perform your modifications, verify the changes, and save the file again.

6.7.2 Running the Network Generation Utility

- Open the Network Table Generation Utility.
- Type the fully-qualified VisualInfo Configuration file name that you saved in the previous steps (for example, BEA0003.CFG). See Figure 45.

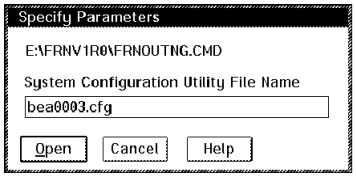


Figure 45. Entering the Configuration File Name

- Type the node name of the workstation you are currently running on (for example, BEA0003). See Figure 46.

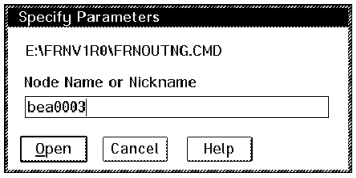


Figure 46. Entering the Node Name of the Workstation

- The utility names the mode as FRNOMVS. Be sure to create a mode on your MVS system with this name. Otherwise, you can edit the network table and change the mode name to match your host. However, any user changes to the table will be overwritten the next time you run the Network Generation Utility.
- Look carefully at the resulting message to be sure the generation was successful. You should see that FRNOLINT.TBL and FRNOSTOP.CMD were renamed as backup files and replaced with new ones. FRNOLINT is the actual network table, and FRNOSTOP is the file used to shut down VisualInfo when you select the Shutdown icon.

  Note: You can always go back to a previous configuration by re-naming your backups to FRNOLINT.TBL and FRNOSTOP.CMD.

6.8 Bringing Up a VisualInfo Client

- Be sure other components are active: VTAM and Communications Manager, your configuration server, your library server, and your object server.
- From the VisualInfo folder, select the **Client Application**, or the **System Administration** program.
- Use a User ID of FRNADMIN and a password of PASSWORD for the initial logon. When your system is ready for production, add other Admin and User IDs, then disable FRNADMIN if desired.

  Note: When you disable the FRNADMIN ID, **be sure you have at least one other ID with full system administration rights**. A user ID can pass on only the rights that it is authorized to when creating another user ID.
- When you are logged on to VisualInfo Administration, you deserve congratulations for doing a fine job of installation and connectivity.

6.9 Shutting Down VisualInfo

- After you have logged off from a VisualInfo application and want to close the system down, open the VisualInfo folder.
- Open the VisualInfo Services Shutdown object and press any key when prompted.
- After the VisualInfo shutdown completes, get a Window list (press CNTL and ESC) and close any remaining programs that begin with FRN.

6.10 Configuration Server Refresh

- Whenever you change the network table on the configuration server, run the Configuration Server Refresh program. This updates the configuration and restarts the configuration server.
- The refresh program is run from an icon in the VisualInfo utilities folder, or from the command line by typing `frnolirn`. It resides in the VisualInfo directory, currently FRNV1R0.

Chapter 7. Installing a VisualInfo Configuration Server

The Configuration Server is a LAN-based VisualInfo component that keeps a record of the locations of the other components. This is done through Advanced Peer-to-Peer Networking (APPN) that allows the other components to refer to the configuration server when establishing a communications session. This session uses Advanced Program-to-Program Communications (APPC), also known as LU 6.2. Using APPN with APPC allows changes to be recorded in one place (the configuration server), rather than in each workstation.

Note: This requires VTAM V4 or later.

In a production environment, you should consider placing your configuration server on a separate workstation for reliability and performance. You do not want the configuration server to be affected by other applications on the workstation.

7.1 Installing a Configuration Server

To install a configuration server, follow all of the steps for a client installation, but with these differences:

- When choosing a system from the Showcase catalog, select VisualInfo System. If you are not using Showcase, install from the VISYSTEM directory.
- Select only the configuration server when you are given a list of components.
- Tailor the CM/2 configuration during installation with the appropriate values. You will probably want your Configuration Server workstation to act as a Network Node. The parameter values are similar to our client example but must be unique on the network. See your LAN administrator for help with this.
- You do not need a CM/2 destination address if your configuration server is on the Network Node.
- You do need to generate a Network Table on the configuration server workstation. Here is one method:
  - Copy the VisualInfo configuration file (BEA0003.CFG, not the CM/2 configuration file) from the client to the FRNV1R0 subdirectory on the configuration server. Rename it to BEA0007.CFG.
  - Run the Network Generation utility, inputting the name of the configuration file (BEA0007.CFG), and the nickname of the configuration server workstation (BEA0007 in this example).

7.2 Tailoring Communications Manager on the Configuration Server

When VisualInfo has been successfully installed, you can view and tailor the new Communications Manager/2 configuration file (FRNONEWN) that was created during installation.

- From the Communications Manager folder, open Communications Manager Setup.
- Choose **Setup**; select the FRNONEWN file that you created; select **OK**.
- Answer Yes (the configuration is used for this workstation).

- Select your Workstation Connection Type (Token-ring or other LAN types) and APPC APIs. This menu varies, depending on the release of CM/2 you are running.

7.2.1 SNA Local Node

- Select **Advanced Configuration**.
- Open (select and configure) **SNA local node characteristics**.
- Ensure that your Network ID (USIBMSC in this example) and your workstation's local node name (BEA0007) are correct from the VisualInfo installation.
- For a configuration server, we recommend a Node type of *Network node.*
- Select **Options**. We recommend that you make the *Local node alias name* the same as the local node name (BEA0007) from the previous panel. If you leave the alias field blank, the system places the local node name in the field. You could add a different name for the alias, but you then need to remember it during other configuration steps.
- Press **OK** twice to return to the Communications Manager Profile List.

7.2.2 SNA Connections

- Select **SNA Connections**, then select **To host**. This defines your connection from the configuration server to the MVS VisualInfo servers.
- Select **Create**, then **Continue**. Enter the values for these fields, using our sample data in the following list as a guide.
  - Link name: HOST0001
  - Activate at startup: Yes (Check box)
  - APPN support: Yes
  - Node ID: 05D A0007
  - LAN destination address: 400008210210
  - Address format: Token Ring
  - Remote SAP: 04
  - Use this host connection as your focal point support: Yes
- Press **OK**, then press **Close** to return to the Communications Manager Profile List.

7.2.3 SNA Features

- Select SNA features. Here you are creating an LU to LU session definition between your local LU and the MVS library and object servers (Partner LUs).

7.2.3.1 Local LU

- Select **Local LUs** and **Create**.
- Enter an LU name of BEA0007A. Name the alias the same for simplicity.
- Ask your VTAM systems programmer if your LU was defined on the MVS system as independent or dependent. Select the correct radio button; if dependent, also type the LU number (1 for this example).
- Choose HOST0001 as your Host link.
- Press **OK**.

7.2.3.2 Partner LUs

- Now select **Partner LUs** and **Create**.
- Here you are creating the connection to the host-based library server (HBLS). A fully qualified LU name consists of the network name followed by the application logical unit name (VTAM APPLID) that you are connecting to. Obtain these from your VTAM systems programmer. In our example, we typed USIBMSC and IMLBRSV. We called the Alias IMLBRSV also, for simplicity. In our example, the MVS partner LU is independent (check with your system programmer). Press **OK**.
- Create the host-based object server (HBOS) in the same way. We used the same network name with an LU name/alias of IMOBRSV, and our MVS partner LU is independent.

7.2.3.3 Modes

- Obtain from your VTAM systems programmer the name of the mode that is defined to MVS/VTAM. Create a mode with the same name in CM/2. VTAM negotiates the mode parameters during the session connection (BIND) and CM/2 agrees to the mode.
- To save your SNA features, select **Close** three times. CM/2 verifies your configuration.

  Note: You may need the CM/2 diskettes to add additional support to Communications Manager/2. If so, you are prompted at this time.
- If no errors are found, select **Close**, stop Communications Manager, restart it, and begin communicating with the host. If a connection is not established, shut down OS/2, power off and restart.

7.2.4 Handling Errors or Warnings

- If you have errors or warnings, you may get a Verify Configuration panel. To get this panel, you must have the Verify option on and FFST/2 started. Follow these instructions to correct the errors:
  - Press the **Show log** push button.
  - The Message Log Formatter shows you a list of messages, beginning with the most recent. You can double-click on any message to get more detail.
  - After you have read all of the messages and written down any user action needed, close the message log. Say Yes when asked if you are sure you want to exit.
- Press **Fix** to fix your errors, or **OK** to exit. Press **Close** until you are back to the OS/2 Desktop. Stop CM/2 and restart to put your changes into effect.

7.3 Bringing up a Configuration Server

- Be sure other components are active: VTAM, Communications Manager, your library server, and your object server.
- From the VisualInfo folder, select **Utilities**.
- From the Utilities folder, select **Configuration Server** refresh. This ensures that the latest network table is being used, and it starts the configuration server program. See topic 7.5, “Configuration Server Refresh” on page 74.

7.4 Shutting Down a Configuration Server

- Open the VisualInfo folder.
- Open the **VisualInfo Services Shutdown object** and press any key when prompted.
- After the VisualInfo shutdown completes, go to the Window list (press CNTL and ESC) and close any remaining programs that begin with FRN.

7.5 Configuration Server Refresh

- Whenever you change the network table on the configuration server, run the Configuration Server Refresh program. This updates the configuration and restarts the configuration server.
- The refresh program is run from an icon in the VisualInfo utilities folder, or from the command line by typing `frnolirn`. It resides in the VisualInfo directory, currently FRNV1R0.

Chapter 8. Installation Verification and Problem Determination

This chapter helps you verify your installation and lists some ways to isolate and solve any problems that may arise.

Problems occur mostly in connecting the components together over a network, so it is important to have people available who are familiar with connectivity on each platform. For this setup, a Communications Manager guru and a VTAM guru would be most helpful.

8.1 Verifying Your Installation

Module C10: Verifying the Installation of an IBM ImagePlus VisualInfo System covers the steps for verifying that the OS/2-based components operate together correctly. Read it along with the steps in the following section. If you received any errors during the installation steps, correct those before proceeding with the verification.

8.1.1 Verification Preparation

Since you have made a number of changes to your OS/2 system or systems, you should perform an orderly shutdown and power off the machine or machines before verifying your installation. Powering off resets the LAN adapter and re-booting allows the OS/2 changes (CONFIG.SYS, CM/2) to take effect.

8.1.2 Starting the Configuration Server

To start the Configuration Server, open the VisualInfo folder and activate Configuration Server refresh. This updates any configuration changes and starts the server.

Note: This step should be done at the configuration server any time changes are made to your configuration.

8.1.3 Starting Your Sessions

Before starting your sessions, you should start FFST/2, which then records any error message into a log. This greatly aids in problem determination. Simply open the FFST/2 folder and double-click on the Start FFST/2 object, or place the START EPW command in your STARTUP.CMD file for the next boot.

Start up the communications session in this order:

1. Start the object server and the library server.
2. Start Communications Manager/2 on the configuration server and on each client.

Check to see if sessions have been established. Open the Subsystem Management object in the CM/2 folder, and select:

1. SNA Subsystem
2. LU 6.2 Sessions

Depending on whether you are displaying this from the client or the configuration server, you should have at least one line item with at least 2 sessions.

- If not, try the Establish pull-down to see if you can get a session to start.
- If not, close down and restart the servers and try to re-establish your session. After that, see topic 8.2, “Problem Determination Procedures -- Communication.”

8.1.4 Logging on to VisualInfo

When you have your SNA sessions established, you are ready to try out your VisualInfo. From your client, open the VisualInfo folder and double-click on the Client Application. You should soon see a Logon box that contains the name of a Library Server. If the name is not what you expected, click on the arrow to the right of it and select the name you expected. If it is not in the list, or if you do not get a name at all, you need to check the client connection with the configuration server. See section 8.1.3, “Starting Your Sessions” on page 75 and section 8.3, “Problem Determination Procedures -- Configuration” on page 77.

If you have successfully selected your Library Server in the Logon box, enter FRNADMIN for the User ID and PASSWORD for the password. Note that VisualInfo does not use UPM for logon verification (so user ID and PASSWORD do not work), but instead uses a separate process.

After a time, you should see disk lights blinking on both client and server or servers. Then you should get a VisualInfo Client Application window. If so, you have successfully installed a VisualInfo client and connected to the Library server. Close the window that logs you off the client.

8.1.5 Logging on to System Administration

Now it is time to set up a test system. After logging off of the client, open the System Administration object from the VisualInfo folder. Use the previous steps and parameters (FRNADMIN/PASSWORD).

When you are logged on, follow the steps in the *Module C10: Verifying the Installation of an IBM ImagePlus VisualInfo System* to create an index class and user ID and define the object server. This verifies that all components are installed and working.

Then close System Administration and log on to the client (from the previous steps) using your newly created user ID. Import the TIFF file that is supplied during installation (see Module C10: Verifying the Installation of an IBM ImagePlus VisualInfo System for exact steps).

8.2 Problem Determination Procedures -- Communication

If you have a problem establishing a communications session, scan the online Problem Determination Guide in the CM/2 folder and use the the following tips to isolate and correct the problem.

8.2.1 Refresh the Configuration Server

Always refresh the configuration server after you make any configuration changes to the system. This is one of the most successful problem-solving steps known. On the configuration server, simply open the VisualInfo icon and double-click on the Refresh Configuration Server icon. Then, try the startup process again.

8.2.2 Look at the FFST/2 Message Log

If you are unsuccessful in getting a session, check the FFST/2 message log. Open the FFST/2 folder and access the Message Log Formatter. It shows each error online, newest to oldest. Double-click on any error to see more detail.

8.2.3 Trace and Print

If you still do not have a session after correcting any FFST/2 messages, you may want to retry the session and trace the events. To set up the trace, open the CM/2 folder and access (double-click on) the Problem Determination Aids - Trace. Start out tracing a few items, then add more if you need to. From the APIs window, highlight the APPC and perhaps the Services. The more types you select, the more detail you get but the more cumbersome it is to read. From the DLCs window, select the IBMTRNET and perhaps the SDLC. From the Events window, select all of the APPC_ events and perhaps the Services. Then type in a file name, such as SESSION1.TRC. From the File pull-down, clear the trace buffer to begin again. Then press the Start pushbutton and begin your Start Session sequence.

When you are finished tracing (for example, when activity stops), go to the Trace window and press the **Save** push button. Then from the File pull-down, select the Format trace file option. On that window, type in your trace file name (for example, session1.trc). Select the **Both** option for output file format, then press the **Format** trace push button. When this is complete, edit or print each file, beginning with the .DET file. You may see messages in the trace that contain return codes. This can help you isolate the problem.

8.3 Problem Determination Procedures -- Configuration

After you have successfully started your communication sessions (CM/2 to VTAM), you should try to log on to the VisualInfo library server from the client. If your VisualInfo configuration is not correct, you may experience one of these problems:

1. The Library Server name does not appear in the logon box.
2. The Library Server name appears, but you do not get logged on.
3. The Library Server name appears and you get logged on, but you do not receive any menu or client activity.

8.3.1 Name Does Not Appear

If the Library Server name does not appear when you try to log on to the client, then your configuration server is at fault. You may have:

- An incomplete connection (session)
- A faulty definition
  - On the client
  - On the configuration server

- Forgotten to refresh the configuration server.

8.3.1.1 An Incomplete Connection

See section 8.1.3, “Starting Your Sessions” on page 75 for the proper way to bring up communication sessions. Use the CM/2 Subsystem Management tool and the trace if needed.

8.3.1.2 A Faulty Definition

See section 6.7, “Generating a VisualInfo Network Table” on page 65 for the proper way to define your VisualInfo system. Ensure that each component has a correct definition. Begin with the configuration server, then check the library server, and the client workstation. For the complete set of steps to configure a client, begin with topic 6.5, “Configuring Communications Manager on the Client” on page 62, and use the hints below along with those steps. For the complete set of steps to configure a configuration server, begin with topic 7.2, “Tailoring Communications Manager on the Configuration Server” on page 71 along with the following hints.

1. Be sure you have shut down and re-booted all of the workstations. This allows changes to CONFIG.SYS to be activated. If you have changed LAN addresses, shut down and power off before re-booting to let the new address take effect.
2. Look at the VisualInfo System Configuration on one of your workstations by running the System Configuration Utility and loading the saved configuration file. (Our example was BEA0003.CFG.) Check the network name, node name, and nickname for each workstation definition to ensure they match your Communications Manager definitions.
3. Ensure that you have allocated the proper VisualInfo component (for example, client) to the correct workstation (for example, BEA0003).
4. After creating a VisualInfo System Configuration with the System Configuration Utility, copy that saved file to the other workstations, thus insuring that you have the same file for all.
5. When each workstation has the correct VisualInfo configuration file, run the Network Generation Utility on each workstation, giving the name of the Configuration file and the appropriate node name of that workstation. Be sure you give the name of the workstation you are on, and not the workstation name you originally created the file on.

8.3.1.3 Refresh the Configuration Server

Even if all of your definitions are correct, you may have forgotten to refresh the configuration server. Go to the workstation containing that component and from the VisualInfo Utilities folder, double-click on **Configuration Server Refresh**. This starts the Configuration Server and updates the definitions from the Network Table. The refresh is run from the command line, for example, frnolirn.

8.3.2 Not Able to Log On

If a Library Server name appears on your logon panel, but the logon times out or the library server is not available, perform the same checks as above on the Library Server workstation or system.

If the library server appears to be defined and configured correctly, check your client and configuration server definitions again using the previous steps.

8.3.3 No Menu or Client Activity

If you do get a message that you are logged on successfully, but no menu appears afterwards, check that you have all of the maintenance for your components (VisualInfo, Communications Manager/2, OS/2, DB2/2).

Appendix A. CICS Installation Jobs

A.1 DB2 CICS Support (DSNTIJSU)

The following JCL is used to link-edit the DB2 CICS modules.

```
//*
//SSRENT EXEC PGM=IEWL,
//        REGION=1024K,
//     PARM='SIZE=(900K,124K),XREF,LET,RENT,NCAL,AMODE=24,RMODE=24'
//SYSPRINT DD SYSOUT=*
//DSNALOAD DD DISP=SHR,
//            DSN=DSN310.ADSNLOAD
//DSNLOAD  DD DISP=SHR,
//            DSN=DSN310.SDSNLOAD
//*  DLOADLIB IS USED FOR DFHEAI AND DFHEAIO CICS MODULES
//DLOADLIB DD DISP=SHR,
//            DSN=CICS330.SDFHLOAD
//SYSLMOD  DD DISP=SHR,
//            DSN=VIHOST.DSN310.SDSNLOAD
//*           DSN=DSN310.SDSNLOAD
//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSLIN   DD  *
 ORDER DFHEAI,DFHEAIO,DSNCCOM0,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCCOM0)
 ENTRY DSNCCOM0
 NAME DSNCCOM0(R)
 ORDER DFHEAI,DFHEAIO,DSNCCOM1,DSNCCMDP,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCCOM1)
 ENTRY DSNCCOM1
 NAME DSNCCOM1(R)
 ORDER DFHEAI,DFHEAIO,DSNCCOM2,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCCOM2)
 ENTRY DSNCCOM2
 NAME DSNCCOM2(R)
 ORDER DFHEAI,DFHEAIO,DSNCEXT1,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCEXT1)
 ENTRY DSNCEXT1
 NAME DSNCEXT1(R)
 ORDER DFHEAI,DFHEAIO,DSNCEXT2,DSNAA,DSNARIB
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCEXT2)
 ENTRY DSNCEXT2
 NAME DSNCEXT2(R)
 ORDER DFHEAI,DFHEAIO,DSNCMSG0,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
```

```
 INCLUDE DSNLOAD(DSNCMSGO)
 ENTRY DSNCMSGO
 NAME DSNCMSGO(R)
 ORDER DFHEAI,DFHEAIO,DSNCSTOP,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCSTOP)
 ENTRY DSNCSTOP
 NAME DSNCSTOP(R)
 ORDER DFHEAI,DFHEAIO,DSNCSTRT,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCSTRT)
 ENTRY DSNCSTRT
 NAME DSNCSTRT(R)
 ORDER DFHEAI,DFHEAIO,DSNCUEXT,DSNAA
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCUEXT)
 ENTRY DSNCUEXT
 NAME DSNCUEXT(R)
 ORDER DFHEAI,DFHEAIO,DSNCEDF1,DSNAA
 MODE AMODE(31),RMODE(24)
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCEDF1)
 ENTRY DSNCEDF1
 NAME DSNCEDF1(R)
 ORDER DFHEAI,DFHEAIO,DSNCEDON,DSNAA
 MODE AMODE(31),RMODE(24)
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCEDON)
 ENTRY DSNCEDON
 NAME DSNCEDON(R)
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCSM31)
 ENTRY DSNCSM31
 ORDER DFHEAI,DFHEAIO,DSNCSM31,DSNAA
 MODE AMODE(31),RMODE(24)
 NAME DSNCSM31(R)
 INCLUDE DLOADLIB(DFHEAI)
 INCLUDE DLOADLIB(DFHEAIO)
 INCLUDE DSNLOAD(DSNCSM21)
 ENTRY DSNCSM21
 ORDER DFHEAI,DFHEAIO,DSNCSM21,DSNAA
 MODE AMODE(31),RMODE(24)
 NAME DSNCSM21(R)
//*
//*********************************************************************
//*        LOAD CICS CSD FILE WITH DB2 DEFINITIONS
//*        MINIMUM CICS ENVIRONMENT --> CICS/ESA 3.1.1
//*********************************************************************
//LOADCSD  EXEC  PGM=DFHCSDUP,REGION=2048K
//STEPLIB  DD DISP=SHR,DSN=VIHOST.CICS330.SDFHLOAD
//         DD DISP=SHR,DSN=CICS330.SDFHLOAD
//DFHCSD   DD DISP=SHR,DSN=VIHOST.CICS330.DFHCSD
//*DFHCSD   DD DISP=SHR,DSN=CICS330.DFHCSD
```

```
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
*********************************************************************
**********(DB2 CICS ATTACHMENT FACILITY DEFINITIONS)*****************
*********************************************************************
ADD    GROUP(DB2) LIST(DB2LIST)
*
DEFINE TRANSACTION(-DIS) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(-REC) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(-STA) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(-STO) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(DISC) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(DISP) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(DSNC) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(MODI) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(STOP) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
DEFINE TRANSACTION(STRT) GROUP(DB2) PROGRAM(DSNCCOM1) TWASIZE(1200)
*
DEFINE PROGRAM(DSNCCOM0) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
*
DEFINE PROGRAM(DSNCCOM1) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
DEFINE PROGRAM(DSNCCOM2) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
DEFINE PROGRAM(DSNCEDF1) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
DEFINE PROGRAM(DSNCEDON) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
DEFINE PROGRAM(DSNCEXT1) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
DEFINE PROGRAM(DSNCEXT2) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
DEFINE PROGRAM(DSNCMSG0) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
DEFINE PROGRAM(DSNCSTOP) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
DEFINE PROGRAM(DSNCSTRT) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
DEFINE PROGRAM(DSNCUEXT) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
DEFINE PROGRAM(DSNCSM21) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
DEFINE PROGRAM(DSNCSM31) GROUP(DB2) LANGUAGE(ASSEMBLER) CEDF(NO)
       EXECKEY(CICS) DATALOCATION(BELOW)
*
*********************************************************************
**********(DB2 CICS SAMPLE APPLICATION DEFINITIONS)******************
*********************************************************************
DEFINE TRANSACTION(D8CS) GROUP(DB2) PROGRAM(DSN8CC0)
DEFINE TRANSACTION(D8PP) GROUP(DB2) PROGRAM(DSN8CP6)
DEFINE TRANSACTION(D8PS) GROUP(DB2) PROGRAM(DSN8CP0)
DEFINE TRANSACTION(D8PT) GROUP(DB2) PROGRAM(DSN8CP3)
DEFINE TRANSACTION(D8PU) GROUP(DB2) PROGRAM(DSN8CP3)
*
DEFINE MAPSET(DSN8CCD) GROUP(DB2)
DEFINE MAPSET(DSN8CCG) GROUP(DB2)
DEFINE MAPSET(DSN8CPD) GROUP(DB2)
DEFINE MAPSET(DSN8CPE) GROUP(DB2)
DEFINE MAPSET(DSN8CPF) GROUP(DB2)
DEFINE MAPSET(DSN8CPG) GROUP(DB2)
DEFINE MAPSET(DSN8CPL) GROUP(DB2)
DEFINE MAPSET(DSN8CPN) GROUP(DB2)
```

```
DEFINE MAPSET(DSN8CPU) GROUP(DB2)
*
DEFINE PROGRAM(DSN8CC0) GROUP(DB2) LANGUAGE(COBOL)
DEFINE PROGRAM(DSN8CC1) GROUP(DB2) LANGUAGE(COBOL)
DEFINE PROGRAM(DSN8CC2) GROUP(DB2) LANGUAGE(COBOL)
DEFINE PROGRAM(DSN8CP0) GROUP(DB2) LANGUAGE(PLI)
DEFINE PROGRAM(DSN8CP1) GROUP(DB2) LANGUAGE(PLI)
DEFINE PROGRAM(DSN8CP2) GROUP(DB2) LANGUAGE(PLI)
DEFINE PROGRAM(DSN8CP3) GROUP(DB2) LANGUAGE(PLI)
DEFINE PROGRAM(DSN8CP6) GROUP(DB2) LANGUAGE(PLI)
DEFINE PROGRAM(DSN8CP7) GROUP(DB2) LANGUAGE(PLI)
DEFINE PROGRAM(DSN8CP8) GROUP(DB2) LANGUAGE(PLI)
DEFINE PROGRAM(DSNTIAC) GROUP(DB2) LANGUAGE(ASSEMBLER)
DEFINE PROGRAM(DSNTIA1) GROUP(DB2) LANGUAGE(ASSEMBLER)
********************************************************************
/*
```

A.2 Link Compatibility Requirements

When you are defining the links between the different systems, you must ensure that the definitions of the links for each system are compatible.

The following diagram shows the compatibility requirements that we used for the links between the Client and the Servers.

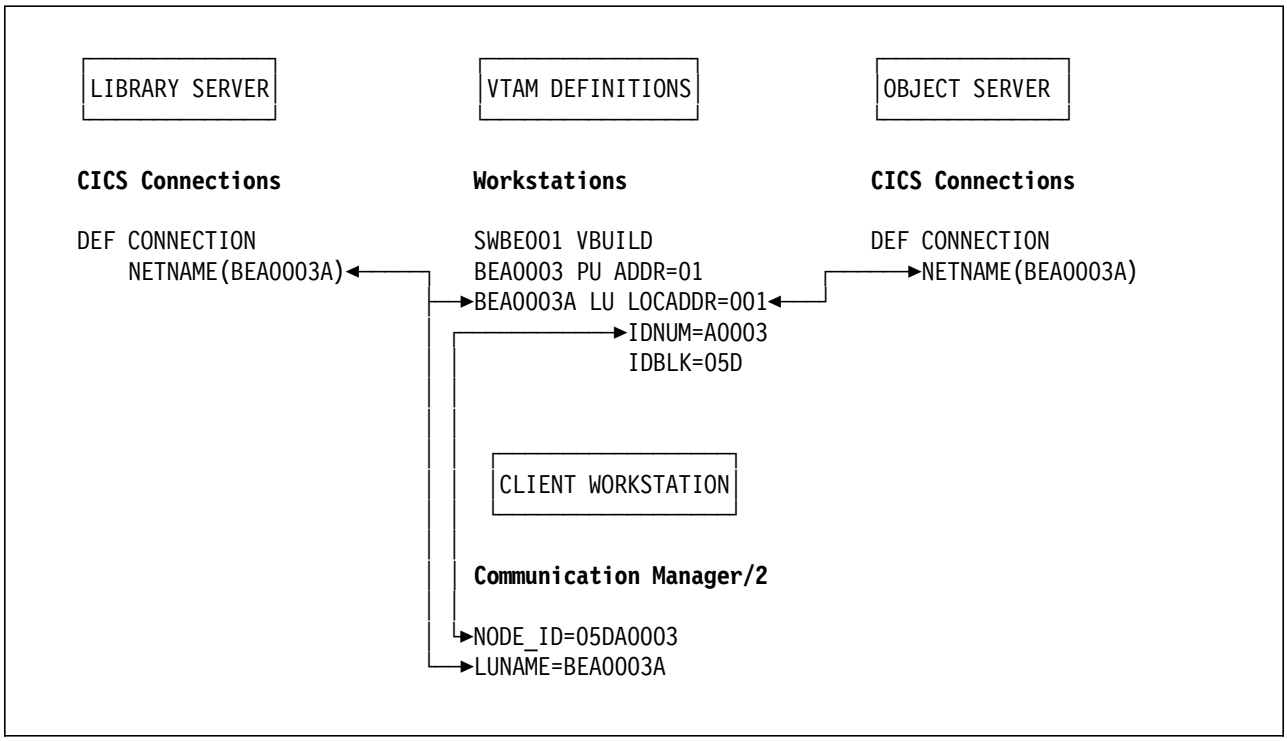
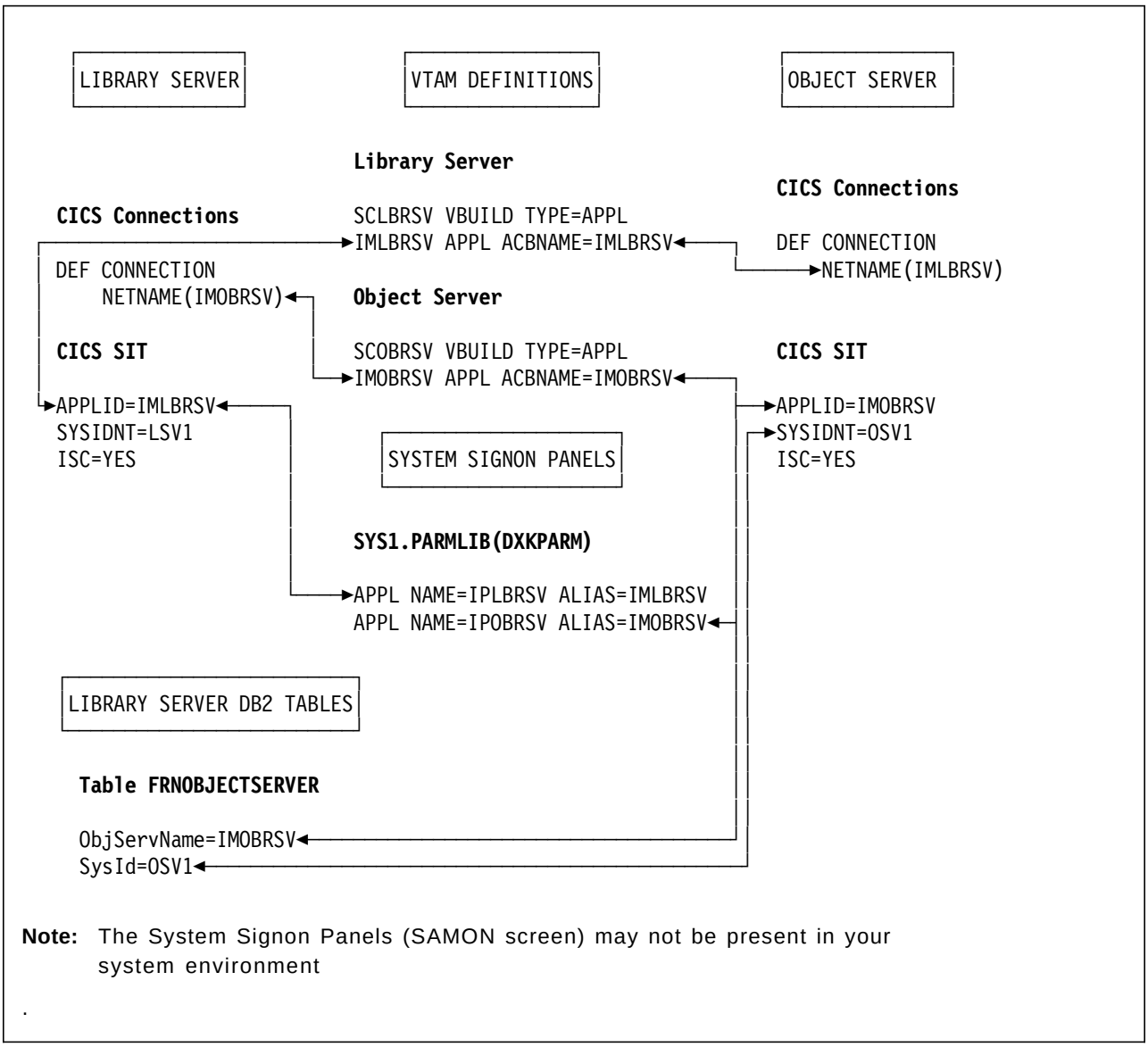


Figure 47. Client Link Compatible Definitions.

The following diagram shows the compatibility requirements that we used for the links between the Library Server and Object Server.



Note: The System Signon Panels (SAMON screen) may not be present in your system environment

.

Figure 48. Server Link Compatible Definitions

A.3 VisualInfo VTAM CICS Region Definition

Library Server

```
SCLBRSV  VBUILD  TYPE=APPL
*
IMLBRSV  APPL  EAS=1024,              VISUALINFO LIBRARY SERVER CICS  *
               ACBNAME=IMLBRSV,       APPLID FOR LIBRARY SERVER ACB   *
               PARSESS=YES,           PARALLEL SESSIONS               *
               MODETAB=APPCMODT,      MODELU62, MODE TABLE            *
               SONSCIP=YES,           SESSION OUTAGE NOTIFICATION     *
               VPACING=10,            APPL PACING                     *
               AUTH=(ACQ,PASS,VPACE)  CICS CAN ACQUIRE & PASS TMLS
```

Object Server

```
SCOBRSV  VBUILD  TYPE=APPL
*
IMOBRSV  APPL  EAS=1024,              VISUALINFO OBJECT SERVER CICS   *
               ACBNAME=IMOBRSV,       APPLID FOR OBJECT SERVER ACB    *
               PARSESS=YES,           PARALLEL SESSIONS               *
               MODETAB=APPCMODT,      MODELU62, MODE TABLE            *
               SONSCIP=YES,           SESSION OUTAGE NOTIFICATION     *
               VPACING=10,            APPL PACING                     *
               AUTH=(ACQ,PASS,VPACE)  CICS CAN ACQUIRE & PASS TMLS
```

A.4 Modetab and Logmode Definitions

APPCMODT Mode Table

```
APPCMODT MODETAB
*
LU62APPB   MODEENT LOGMOD=LU62APPB,                                    *
                   TYPE=X'001',                                        *
                   FMPROF=X'13',                                       *
                   TSPROF=X'07',                                       *
                   PRIPROT=X'B0',                                      *
                   SECPROT=X'B0',                                      *
                   COMPROT=X'50B9',                                    *
                   SSNDPAC=X'0A',                                      *
                   SRCVPAC=X'0A',                                      *
                   RUSIZE=X'0000',                                     *
                   PSNDPAC=X'0A',                                      *
                   PSERVIC=X'060200000000000000002C00'
*
ISCLU62  MODEENT LOGMOD=ISCLU62,                                       *
                   TYPE=X'000',                                        *
                   FMPROF=X'13',                                       *
                   TSPROF=X'07',                                       *
                   PRIPROT=X'B0',                                      *
                   SECPROT=X'B0',                                      *
                   COMPROT=X'50B1',                                    *
                   SSNDPAC=X'0A',                                      *
                   SRCVPAC=X'0A',                                      *
                   RUSIZE=X'8C8C',                                     *
                   PSNDPAC=X'0A',                                      *
                   PSERVIC=X'060200000000000000002C00'
*
* SNA SERVICE MANAGER
SNASVCMG MODEENT LOGMOD=SNASVCMG
*
*                  MODEEND
                   END
```

A.5 VTAM Definition for the Workstations

The following example is the PU definition for the workstations that we used in this project.

```
*****************************************************************    00001000
SWBE001  VBUILD  TYPE=SWNET,                                        +00002000
              MAXGRP=1,                                           +00003000
              MAXNO=1                                              00004000
*                                                                  00005000
BEA0003  PU    ADDR=01,                                           +00006000
              IDBLK=05D,IDNUM=A0003,                              +00007000
              ANS=CONT,DISCNT=NO,                                 +00008000
              IRETRY=NO,ISTATUS=ACTIVE,                           +00009000
              MAXDATA=4105,MAXOUT=1,                              +00010000
              MAXPATH=1,                                          +00011000
              PUTYPE=2,SECNET=NO,                                 +00012000
              MODETAB=POKMODE,DLOGMOD=DYNRMT,                     +00013000
              USSTAB=USSRDYN,LOGAPPL=SCGVAMP,                     +00014000
              PACING=1,VPACING=2                                   00015000
*                                                                  00016000
BEA0003A LU    LOCADDR=001,DLOGMOD=LU62APPB,SSCPFM=FSS             00017000
BEA0003B LU    LOCADDR=002                                         00018000
BEA0003C LU    LOCADDR=003                                         00019000
BEA0003D LU    LOCADDR=004                                         00020000
BEA0003E LU    LOCADDR=005                                         00021000
BEA0003F LU    LOCADDR=006,DLOGMOD=LU62APPB,SSCPFM=FSS             00022000
BEA0003G LU    LOCADDR=0,DLOGMOD=LU62APPB                          00023000
BEA0003H LU    LOCADDR=0,DLOGMOD=LU62APPA                          00024000
*                                                                  00025000
```

A.6 Sample Session Definitions

A.6.1 Library Server to Object Server

```
*-------------------------------------------------------------------*
* CONNECTION definition for HBLS to Object Server                   *
*-------------------------------------------------------------------*
  DEF CONNECTION(COS1) GROUP(LS2OS)
           DESCRIPTION(CONNECTION to OBJECT SERVER 1)
           NETNAME(IMOBRSV)
           ACCESSMETHOD(VTAM)
           PROTOCOL(APPC)
           AUTOCONNECT(NO)
           INSERVICE(YES)
*-------------------------------------------------------------------*
* SESSIONS   definition for HBLS to Object Servers                  *
*-------------------------------------------------------------------*
  DEF SESSIONS(SOS1) GROUP(LS2OS)
           DESCRIPTION(SESSION to OBJECT SERVER 1   )
           CONNECTION(COS1)
           MODENAME(ISCLU62)
           PROTOCOL(APPC)
           MAXIMUM(4,2)
           RECEIVESIZE(4096)
           AUTOCONNECT(YES)
```

A.6.2 Library Server to Client

```
*--------------------------------------------------------------------*
* CONNECTION definition for HBLS to application client workstation   *
*--------------------------------------------------------------------*
  DEF CONNECTION(C003) GROUP(LS2CLNT)
           DESCRIPTION(Sample CONNECTION to application client)
           NETNAME(BEA0003A)
           ACCESSMETHOD(VTAM)
           PROTOCOL(APPC)
           AUTOCONNECT(YES)
           INSERVICE(YES)
  DEF CONNECTION(C007) GROUP(LS2CLNT)
           DESCRIPTION(Sample CONNECTION to application client)
           NETNAME(BEA0007A)
           ACCESSMETHOD(VTAM)
           PROTOCOL(APPC)
           AUTOCONNECT(YES)
           INSERVICE(YES)
*--------------------------------------------------------------------*
* SESSIONS   definition for HBLS to application client workstation   *
*--------------------------------------------------------------------*
  DEF SESSIONS(S003) GROUP(LS2CLNT)
           DESCRIPTION(Sample SESSIONS to application client)
           CONNECTION(C003)
           MODENAME(LU62APPB)
           PROTOCOL(APPC)
           MAXIMUM(4,2)
           SENDSIZE(4096)
           RECEIVESIZE(1920)
           AUTOCONNECT(YES)
  DEF SESSIONS(S007) GROUP(LS2CLNT)
           DESCRIPTION(Sample SESSIONS to application client)
           CONNECTION(C007)
           MODENAME(LU62APPB)
           PROTOCOL(APPC)
           MAXIMUM(4,2)
           SENDSIZE(4096)
           RECEIVESIZE(1920)
           AUTOCONNECT(YES)
```

A.6.3 Object Server to Library Server

```
*--------------------------------------------------------------------*
* CONNECTION definition for HBOS to HBLS                             *
*--------------------------------------------------------------------*
  DEF CONNECTION(CLS1) GROUP(OS2LS)
           DESCRIPTION(CONNECTION to LIBRARY SERVER 1)
           NETNAME(IMLBRSV)
           ACCESSMETHOD(VTAM)
           PROTOCOL(APPC)
           AUTOCONNECT(YES)
           INSERVICE(YES)
*--------------------------------------------------------------------*
* SESSIONS   definition for HBOS to HBLS                             *
*--------------------------------------------------------------------*
  DEF SESSIONS(SLS1) GROUP(OS2LS)
           DESCRIPTION(SESSION to LIBRARY SERVER 1   )
           CONNECTION(CLS1)
           MODENAME(ISCLU62)
```

```
            PROTOCOL(APPC)
            MAXIMUM(4,2)
            SENDSIZE(4096)
            RECEIVESIZE(4096)
            AUTOCONNECT(YES)
```

A.6.4 Object Server to Client

```
*--------------------------------------------------------------------*
* CONNECTION definition for HBOS to application client workstation   *
*--------------------------------------------------------------------*
  DEF CONNECTION(CO03) GROUP(OS2CLNT)
            DESCRIPTION(Sample CONNECTION to application client)
            NETNAME(BEA0003A)
            ACCESSMETHOD(VTAM)
            PROTOCOL(APPC)
            AUTOCONNECT(YES)
            INSERVICE(YES)
  DEF CONNECTION(CO07) GROUP(OS2CLNT)
            DESCRIPTION(Sample CONNECTION to application client)
            NETNAME(BEA0007A)
            ACCESSMETHOD(VTAM)
            PROTOCOL(APPC)
            AUTOCONNECT(YES)
            INSERVICE(YES)
*--------------------------------------------------------------------*
* SESSIONS   definition for HBLS to application client workstation   *
*--------------------------------------------------------------------*
  DEF SESSIONS(SO03) GROUP(OS2CLNT)
            DESCRIPTION(Sample SESSIONS to application client)
            CONNECTION(CO03)
            MODENAME(LU62APPB)
            PROTOCOL(APPC)
            MAXIMUM(4,2)
            SENDSIZE(4096)
            RECEIVESIZE(1920)
            AUTOCONNECT(YES)
  DEF SESSIONS(SO07) GROUP(OS2CLNT)
            DESCRIPTION(Sample SESSIONS to application client)
            CONNECTION(CO07)
            MODENAME(LU62APPB)
            PROTOCOL(APPC)
            MAXIMUM(4,2)
            SENDSIZE(4096)
            RECEIVESIZE(1920)
            AUTOCONNECT(YES)
```

A.7 CICS Tables

A.7.1 Library Server SIT

```
SIT      TITLE 'DFHSIT - CICS LIB SVR SYSTEM INITIALIZATION TABLE'
         DFHSIT TYPE=CSECT,
               ADI=30,           XRF(B) - Alternate delay interval
               AIEXIT=DFHZATDX,  Autoinstall user program name
               AILDELAY=0,       Delete delay period for AI TCTTEs
               AIQMAX=100,       Maximum no. of terminals queued for AI
               AIRDELAY=700,     Restart delay period for AI TCTTEs
               AKPFREQ=200,      Activity keypoint frequency
               AMXT=32,          MAX NO. OF ACTIVE TASKS EQU MXT= VALUE
               APPLID=IMLBRSV, <<VTAM APPL identifier
               AUTCONN=0,        Autoconnect delay
               AUXTR=OFF,        Auxiliary trace option
               AUXTRSW=NO,       Auxiliary trace autoswitch facility
               BMS=(FULL,,UNALIGN,DDS), Basic Mapping Support options
               CDSASZE=1M,       CICS DSA Below 16MB line
               CICSSVC=216,      The CICS SVC number
               CLSDSTP=NOTIFY,   Notification for ISSUE PASS command
               CLT=,             The command list table option/suffix
               CMXT=(4,4,4,4,4,4,4,4,4,4), <<max. no. of tasks classes
               CSCS=64K,         CICS DSA cushion below 16MB
               CSDACC=READWRITE, CSD access
               CSDBKUP=STATIC,   Backuptype of CSD (STATIC or DYNAMIC)
               CSDBUFND=,        Number of data buffers for the CSD
               CSDBUFNI=,        Number of index buffers for the CSD
               CSDDISP=,         CSD Disposition for dynamic allocation
               CSDDSN=,          CSD datasetname for dynamic allocation
               CSDFRLOG=NO,      Journal id. for CSD forward recovery
               CSDJID=NO,        Journal id. for CSD auto. journaling
               CSDLSRNO=1,       The VSAM LSR pool number for the CSD
               CSDRECOV=NONE,    CSD recoverable file option
               CSDSTRNO=2,       CSD Number of strings
               CWAKEY=USER,      CWA storage key
               DATFORM=MMDDYY,   CSA date format
               DBP=1$,           Required version of DBP with DLI=NO
               DBUFSZ=500,       Dynamic backout buffer size
               DCT=10,         <<Dest. control table option/suffix
               DDIR=YES,         DL/I DMB directory option/suffix
               DFLTUSER=CICSUSER, Default user
               DIP=NO,           Batch data interchange program
               DISMACP=YES,      Disable macro programs
               DLDBRC=NO,        DL/I DBRC support - CICS local DL/I
               DLI=NO,           DL/1 option
               DLIOLIM=100,      Number of errors per DL/I data base
               DLIRLM=NO,        DL/I IRLM option/name
               DLLPA=NO,         Use IMS/VS modules from LPA option
               DLMON=NO,         DL/I Data Base Monitor option
               DLTHRED=1,        DL/I number of threads (CICS-DLI)
               DLXCPVR=NO,       Page-fix ISAM/OSAM buffers for DL/I
               DMBPL=4,          DMB pool size in 1024-byte blocks
               DTRPGM=DFHDYP,    Dynamic transaction routing program
               DUMP=YES,         Dump option
               DUMPDS=AUTO,      CICS dump data set opening option
               DUMPSW=NO,        Dump data set autoswitch option
               DURETRY=30,       SDUMP total retry time (in seconds)
```

```
ECDSASZE=8M,    <<Size of CICS DSA above 16MB line
ECSCS=256K,       Extended Cics DSA cushion size
ENQPL=2,          Max.control.blk.space (in 1K blocks)
EODI=EO,          End-of-data indicator for seq. devices
ERDSASZE=8M,    <<Size of R/O  DSA above 16MB line
ERSCS=256K,       Extended DSA storage cushion size
ESMEXITS=NOINSTLN, External security manager exits
EUDSASZE=8M,    <<Size of USER DSA above 16MB line
EUSCS=256K,       Extended User DSA cushion size
FCT=NO,           File control table option/suffix
FEPI=NO,          Front-End Programming Interface
FLDSEP='    ',    End-of-field separator characters
FLDSTRT=' ',      Field start character for builtin fn
GMTEXT='WELCOME TO VI LIBRARY SERVER (IMLBRSV)',
GMTRAN=CSGM,      Initial transaction
GRPLIST=LSV1LIST,  << list name of CSD for start up
GTFTR=OFF,        GTF trace option
HPO=NO,           VTAM High Performance Option (HPO)
ICP=COLD,         Interval control pgm. start option
ICV=1000,         Region exit interval (milliseconds)
ICVR=5000,        Runaway task interval (milliseconds)
ICVTSD=500,       Terminal scan delay interval (  "  )
INITPARM=,        Initialisation parms for programs
INTTR=ON,         CICS internal trace option
IRCSTRT=NO,       Interregion communication start
ISC=YES,        <<Intersystem communication option
ISRDELAY=30,      Timeout value for SNT Table
JCT=NO,         <<Journal control table option/suffix
JESDI=30,         JES delay interval for XRF alternate
LGNMSG=NO,        Extract VTAM logon data
LPA=NO,           Use-LPA option for CICS/user modules
MAXSMIR=999,      Max. no. of suspended mirror tasks
MCT=NO,           Monitoring cntl.table option/suffix
MN=OFF,           CICS monitoring option
MNEVE=OFF,        Monitoring event class option
MNEXC=OFF,        Monitoring exception class option
MNPER=OFF,        Monitoring performance class option
MROBTCH=1,        Number of MRO requests to batch
MROLRM=NO,        Long-running mirror task option
MSGCASE=MIXED,    System console MSG level option
MSGLVL=1,         System console MSG level option
MXT=32,           Maximum number of tasks in CICS
NATLANG=E,        List of national languages
OPERTIM=120,      Write to operator timeout (seconds)
OPNDLIM=10,       OPNDST/CLSDST request limit
PARMERR=INTERACT, System init. parameter errors option
PDI=30,           Primary delay interval - XRF active
PDIR=NO,        <<DL/I PSB directory option/suffix
PGCHAIN=,         BMS CHAIN command
PGCOPY=,          BMS COPY command
PGPURGE=,         BMS PURGE command
PGRET=,           BMS RETURN command
PISCHD=YES,       Program isolation scheduling option
PLTPI=I1,       <<Program list table PI option/suffix
PLTSD=S1,       <<Program list table SD option/suffix
PRGDLAY=0,        BMS purge delay interval
PRINT=NO,         Print key option
PRTYAGE=32768,    Dispatcher priority ageing value
PSBCHK=NO,        PSB resource checking required
```

```
PSBPL=4,          PSB pool size in 1024-byte blocks
PVDELAY=30,       Timeout value for LUIT Table
RAMAX=256,        Max. I/O area for RECEIVE ANY
RAPOOL=2,         Max. RECEIVE ANY Request Parm.Lists
RENTPGM=PROTECT,  Reentrant program write protection
RESP=FME,         Logical unit response type
RMTRAN=CSGM,      XRF alternate recovery transaction
RST=NO,           Recovery service table (XRF-DBCTL)
SEC=YES,          External security manager option
SECPRFX=NO,       Security prefix
SKRPA1=,          SKR PA1 PAGE RETRIEVAL CMD
SKRPA2=,          SKR PA2 PAGE RETRIEVAL CMD
SKRPA3=,          SKR PA3 PAGE RETRIEVAL CMD
SKRPF1=,          SKR PF1 PAGE RETRIEVAL CMD
SKRPF2=,          SKR PF2 PAGE RETRIEVAL CMD
SKRPF3=,          SKR PF3 PAGE RETRIEVAL CMD
SKRPF4=,          SKR PF4 PAGE RETRIEVAL CMD
SKRPF5=,          SKR PF5 PAGE RETRIEVAL CMD
SKRPF6=,          SKR PF6 PAGE RETRIEVAL CMD
SKRPF7=,          SKR PF7 PAGE RETRIEVAL CMD
SKRPF8=,          SKR PF8 PAGE RETRIEVAL CMD
SKRPF9=,          SKR PF9 PAGE RETRIEVAL CMD
SKRPF10=,         SKR PF10 PAGE RETRIEVAL CMD
SKRPF11=,         SKR PF11 PAGE RETRIEVAL CMD
SKRPF12=,         SKR PF12 PAGE RETRIEVAL CMD
SKRPF13=,         SKR PF13 PAGE RETRIEVAL CMD
SKRPF14=,         SKR PF14 PAGE RETRIEVAL CMD
SKRPF15=,         SKR PF15 PAGE RETRIEVAL CMD
SKRPF16=,         SKR PF16 PAGE RETRIEVAL CMD
SKRPF17=,         SKR PF17 PAGE RETRIEVAL CMD
SKRPF18=,         SKR PF18 PAGE RETRIEVAL CMD
SKRPF19=,         SKR PF19 PAGE RETRIEVAL CMD
SKRPF20=,         SKR PF20 PAGE RETRIEVAL CMD
SKRPF21=,         SKR PF21 PAGE RETRIEVAL CMD
SKRPF22=,         SKR PF22 PAGE RETRIEVAL CMD
SKRPF23=,         SKR PF23 PAGE RETRIEVAL CMD
SKRPF24=,         SKR PF24 PAGE RETRIEVAL CMD
SKRPF25=,         SKR PF25 PAGE RETRIEVAL CMD
SKRPF26=,         SKR PF26 PAGE RETRIEVAL CMD
SKRPF27=,         SKR PF27 PAGE RETRIEVAL CMD
SKRPF28=,         SKR PF28 PAGE RETRIEVAL CMD
SKRPF29=,         SKR PF29 PAGE RETRIEVAL CMD
SKRPF30=,         SKR PF30 PAGE RETRIEVAL CMD
SKRPF31=,         SKR PF31 PAGE RETRIEVAL CMD
SKRPF32=,         SKR PF32 PAGE RETRIEVAL CMD
SKRPF33=,         SKR PF33 PAGE RETRIEVAL CMD
SKRPF34=,         SKR PF34 PAGE RETRIEVAL CMD
SKRPF35=,         SKR PF35 PAGE RETRIEVAL CMD
SKRPF36=,         SKR PF36 PAGE RETRIEVAL CMD
SPCTR=(1,2),      Level(s) of special tracing required
SPOOL=NO,         System spooling interface option
SRBSVC=215,       HPO Type 6 SVC number
SRT=YES,          System recovery table option/suffix
START=AUTO,       CICS system initialization option
STARTER=YES,      Starter ($ and #) suffixes option
STATRCD=OFF,      statistics recording status
STGPROT=NO,       Storage protection facility
STGRCVY=NO,       Storage recovery option
STNTR=1,          Level of standard tracing required
```

```
            SUBTSKS=0,        Number of concurrent mode TCBs
            SUFFIX=10,      <<Suffix of this SIT
            SYSIDNT=LSV1,   <<Local system identifier
            SYSTR=ON,         Master system trace flag
            TAKEOVR=MANUAL,   XRF alternate takeover option
            TBEXITS=,         Transaction backout exit programs
            TCAM=NO,          TCAM option
            TCP=YES,          Terminal control program option/suffix
            TCT=YES,          Terminal control table option/suffix
            TCTUAKEY=USER,    TCT user area storage key
            TCTUALOC=BELOW,   TCT user area below 16MB
            TD=(3,3),       <<Transient data buffers and strings
            TRAP=OFF,         F.E. global trap exit option
            TRTABSZ=16,       Internal trace table size in 1K bytes
            TS=(,3,3),      <<Temporary storage buffers and strings
            TSMGSET=4,        # of entries for pointers to TS MSGset
            TST=NO,           Temporary storage table option/suffix
            UDSASZE=3M,     <<Size of USER DSA below 16MB line
            USCS=64K,         USER DSA cushion below 16MB
            USERTR=ON,        Master user trace flag
            VTAM=YES,         VTAM access method option
            WRKAREA=512,      Common work area (CWA) size in bytes
            XAPPC=NO,         RACF class APPCLU required
            XCMD=NO,          SPI resource security class name: RACF
            XDCT=NO,          DCT resource security class name: RACF
            XFCT=NO,          FCT resource security class name: RACF
            XJCT=NO,          JCT resource security class name: RACF
            XLT=NO,           Transaction list table option/suffix
            XPCT=NO,          PCT resource security class name: RACF
            XPPT=NO,          PPT resource security class name: RACF
            XPSB=NO,          PSB resource security class name: RACF
            XRF=NO,           Extended recovery feature (XRF) option
            XRFSOFF=NOFORCE,  XRF -  Re-sign on after takeover
            XRFSTME=5,        XRF -  sign off timeout value
            XTRAN=NO,         Transaction security class name:  RACF
            XTST=NO,          TST resource security class name: RACF
      END   DFHSITBA
```

A.7.2 Object Server SIT

```
SIT   TITLE 'DFHSIT - CICS OBJ SVR  SYSTEM INITIALIZATION TABLE'
      DFHSIT TYPE=CSECT,
            ADI=30,           XRF(B) - Alternate delay interval
            AIEXIT=DFHZATDX,  Autoinstall user program name
            AILDELAY=0,       Delete delay period for AI TCTTEs
            AIQMAX=100,       Maximum no. of terminals queued for AI
            AIRDELAY=700,     Restart delay period for AI TCTTEs
            AKPFREQ=200,      Activity keypoint frequency
            AMXT=32,          MAX NO. OF ACTIVE TASKS EQU MXT= VALUE
            APPLID=IMOBRSV, <<VTAM APPL identifier
            AUTCONN=0,        Autoconnect delay
            AUXTR=OFF,        Auxiliary trace option
            AUXTRSW=NO,       Auxiliary trace autoswitch facility
            BMS=(FULL,,UNALIGN,DDS), Basic Mapping Support options
            CDSASZE=1M,       CICS DSA Below 16MB line
            CICSSVC=216,      The CICS SVC number
            CLSDSTP=NOTIFY,   Notification for ISSUE PASS command
            CLT=,             The command list table option/suffix
            CMXT=,            Maximum no. of tasks in classes (10)
```

```
CSCS=64K,          CICS DSA cushion below 16MB
CSDACC=READWRITE, CSD access
CSDBKUP=STATIC,   Backuptype of CSD (STATIC or DYNAMIC)
CSDBUFND=,         Number of data buffers for the CSD
CSDBUFNI=,         Number of index buffers for the CSD
CSDDISP=,          CSD Disposition for dynamic allocation
CSDDSN=,           CSD datasetname for dynamic allocation
CSDFRLOG=NO,       Journal id. for CSD forward recovery
CSDJID=NO,         Journal id. for CSD auto. journaling
CSDLSRNO=1,        The VSAM LSR pool number for the CSD
CSDRECOV=NONE,     CSD recoverable file option
CSDSTRNO=2,        CSD Number of strings
CWAKEY=USER,       CWA storage key
DATFORM=MMDDYY,    CSA date format
DBP=1$,            Required version of DBP with DLI=NO
DBUFSZ=500,        Dynamic backout buffer size
DCT=10,          <<Dest. control table option/suffix
DDIR=YES,          DL/I DMB directory option/suffix
DFLTUSER=CICSUSER, Default user
DIP=NO,            Batch data interchange program
DISMACP=YES,       Disable macro programs
DLDBRC=NO,         DL/I DBRC support - CICS local DL/I
DLI=NO,            DL/1 option
DLIOLIM=100,       Number of errors per DL/I data base
DLIRLM=NO,         DL/I IRLM option/name
DLLPA=NO,          Use IMS/VS modules from LPA option
DLMON=NO,          DL/I Data Base Monitor option
DLTHRED=1,         DL/I number of threads (CICS-DLI)
DLXCPVR=NO,        Page-fix ISAM/OSAM buffers for DL/I
DMBPL=4,           DMB pool size in 1024-byte blocks
DTRPGM=DFHDYP,     Dynamic transaction routing program
DUMP=YES,          Dump option
DUMPDS=AUTO,       CICS dump data set opening option
DUMPSW=NO,         Dump data set autoswitch option
DURETRY=30,        SDUMP total retry time (in seconds)
ECDSASZE=8M,     <<Size of CICS DSA above 16MB line
ECSCS=256K,        Extended Cics DSA cushion size
ENQPL=2,           Max.control.blk.space (in 1K blocks)
EODI=E0,           End-of-data indicator for seq. devices
ERDSASZE=8M,     <<Size of R/O  DSA above 16MB line
ERSCS=256K,        Extended DSA storage cushion size
ESMEXITS=NOINSTLN, External security manager exits
EUDSASZE=8M,     <<Size of USER DSA above 16MB line
EUSCS=256K,        Extended User DSA cushion size
FCT=YES,           File control table option/suffix
FEPI=NO,           Front-End Programming Interface
FLDSEP='     ',    End-of-field separator characters
FLDSTRT=' ',       Field start character for builtin fn
GMTEXT='Welcome to VisualInfo Object Server (IMOBRSV)'
GMTRAN=CSGM,       Initial transaction
GRPLIST=OSV1LIST, <<List name of CSD for startup
GTFTR=OFF,         GTF trace option
HPO=NO,            VTAM High Performance Option (HPO)
ICP=COLD,          Interval control pgm. start option
ICV=1000,          Region exit interval (milliseconds)
ICVR=5000,         Runaway task interval (milliseconds)
ICVTSD=500,        Terminal scan delay interval (  ″  )
INITPARM=,         Initialisation parms for programs
INTTR=ON,          CICS internal trace option
```

```
IRCSTRT=NO,       Interregion communication start
ISC=YES,        <<Intersystem communication option
ISRDELAY=30,      Timeout value for SNT Table
JCT=YES,          Journal control table option/suffix
JESDI=30,         JES delay interval for XRF alternate
LGNMSG=NO,        Extract VTAM logon data
LPA=NO,           Use-LPA option for CICS/user modules
MAXSMIR=999,      Max. no. of suspended mirror tasks
MCT=NO,           Monitoring cntl.table option/suffix
MN=OFF,           CICS monitoring option
MNEVE=OFF,        Monitoring event class option
MNEXC=OFF,        Monitoring exception class option
MNPER=OFF,        Monitoring performance class option
MROBTCH=1,        Number of MRO requests to batch
MROLRM=NO,        Long-running mirror task option
MSGCASE=MIXED,    System console MSG level option
MSGLVL=1,         System console MSG level option
MXT=32,           Maximum number of tasks in CICS
NATLANG=E,        List of national languages
OPERTIM=120,      Write to operator timeout (seconds)
OPNDLIM=10,       OPNDST/CLSDST request limit
PARMERR=INTERACT, System init. parameter errors option
PDI=30,           Primary delay interval - XRF active
PDIR=NO,        <<DL/I PSB directory option/suffix
PGCHAIN=,         BMS CHAIN command
PGCOPY=,          BMS COPY command
PGPURGE=,         BMS PURGE command
PGRET=,           BMS RETURN command
PISCHD=YES,       Program isolation scheduling option
PLTPI=I2,       <<Program list table PI option/suffix
PLTSD=S2,       <<Program list table SD option/suffix
PRGDLAY=0,        BMS purge delay interval
PRINT=NO,         Print key option
PRTYAGE=32768,    Dispatcher priority ageing value
PSBCHK=NO,        PSB resource checking required
PSBPL=4,          PSB pool size in 1024-byte blocks
PVDELAY=30,       Timeout value for LUIT Table
RAMAX=256,        Max. I/O area for RECEIVE ANY
RAPOOL=2,         Max. RECEIVE ANY Request Parm.Lists
RENTPGM=PROTECT,  Reentrant program write protection
RESP=FME,         Logical unit response type
RMTRAN=CSGM,      XRF alternate recovery transaction
RST=NO,           Recovery service table (XRF-DBCTL)
SEC=YES,          External security manager option
SECPRFX=NO,       Security prefix
SKRPA1=,          SKR PA1 PAGE RETRIEVAL CMD
SKRPA2=,          SKR PA2 PAGE RETRIEVAL CMD
SKRPA3=,          SKR PA3 PAGE RETRIEVAL CMD
SKRPF1=,          SKR PF1 PAGE RETRIEVAL CMD
SKRPF2=,          SKR PF2 PAGE RETRIEVAL CMD
SKRPF3=,          SKR PF3 PAGE RETRIEVAL CMD
SKRPF4=,          SKR PF4 PAGE RETRIEVAL CMD
SKRPF5=,          SKR PF5 PAGE RETRIEVAL CMD
SKRPF6=,          SKR PF6 PAGE RETRIEVAL CMD
SKRPF7=,          SKR PF7 PAGE RETRIEVAL CMD
SKRPF8=,          SKR PF8 PAGE RETRIEVAL CMD
SKRPF9=,          SKR PF9 PAGE RETRIEVAL CMD
SKRPF10=,         SKR PF10 PAGE RETRIEVAL CMD
SKRPF11=,         SKR PF11 PAGE RETRIEVAL CMD
```

```
SKRPF12=,         SKR PF12 PAGE RETRIEVAL CMD
SKRPF13=,         SKR PF13 PAGE RETRIEVAL CMD
SKRPF14=,         SKR PF14 PAGE RETRIEVAL CMD
SKRPF15=,         SKR PF15 PAGE RETRIEVAL CMD
SKRPF16=,         SKR PF16 PAGE RETRIEVAL CMD
SKRPF17=,         SKR PF17 PAGE RETRIEVAL CMD
SKRPF18=,         SKR PF18 PAGE RETRIEVAL CMD
SKRPF19=,         SKR PF19 PAGE RETRIEVAL CMD
SKRPF20=,         SKR PF20 PAGE RETRIEVAL CMD
SKRPF21=,         SKR PF21 PAGE RETRIEVAL CMD
SKRPF22=,         SKR PF22 PAGE RETRIEVAL CMD
SKRPF23=,         SKR PF23 PAGE RETRIEVAL CMD
SKRPF24=,         SKR PF24 PAGE RETRIEVAL CMD
SKRPF25=,         SKR PF25 PAGE RETRIEVAL CMD
SKRPF26=,         SKR PF26 PAGE RETRIEVAL CMD
SKRPF27=,         SKR PF27 PAGE RETRIEVAL CMD
SKRPF28=,         SKR PF28 PAGE RETRIEVAL CMD
SKRPF29=,         SKR PF29 PAGE RETRIEVAL CMD
SKRPF30=,         SKR PF30 PAGE RETRIEVAL CMD
SKRPF31=,         SKR PF31 PAGE RETRIEVAL CMD
SKRPF32=,         SKR PF32 PAGE RETRIEVAL CMD
SKRPF33=,         SKR PF33 PAGE RETRIEVAL CMD
SKRPF34=,         SKR PF34 PAGE RETRIEVAL CMD
SKRPF35=,         SKR PF35 PAGE RETRIEVAL CMD
SKRPF36=,         SKR PF36 PAGE RETRIEVAL CMD
SPCTR=(1,2),      Level(s) of special tracing required
SPOOL=NO,         System spooling interface option
SRBSVC=215,       HPO Type 6 SVC number
SRT=YES,          System recovery table option/suffix
START=AUTO,       CICS system initialization option
STARTER=YES,      Starter ($ and #) suffixes option
STATRCD=OFF,      statistics recording status
STGPROT=NO,       Storage protection facility
STGRCVY=NO,       Storage recovery option
STNTR=1,          Level of standard tracing required
SUBTSKS=0,        Number of concurrent mode TCBs
SUFFIX=20,      <<Suffix of this SIT
SYSIDNT=OSV1,   <<Local system identifier
SYSTR=ON,         Master system trace flag
TAKEOVR=MANUAL,   XRF alternate takeover option
TBEXITS=,         Transaction backout exit programs
TCAM=NO,          TCAM option
TCP=YES,          Terminal control program option/suffix
TCT=YES,          Terminal control table option/suffix
TCTUAKEY=USER,    TCT user area storage key
TCTUALOC=BELOW,   TCT user area below 16MB
TD=(3,3),       <<Transient data buffers and strings
TRAP=OFF,         F.E. global trap exit option
TRTABSZ=16,       Internal trace table size in 1K bytes
TS=(,3,3),      <<Temporary storage buffers and strings
TSMGSET=4,        # of entries for pointers to TS MSGset
TST=NO,           Temporary storage table option/suffix
UDSASZE=3M,     <<Size of USER DSA below 16MB line
USCS=64K,         USER DSA cushion below 16MB
USERTR=ON,        Master user trace flag
VTAM=YES,         VTAM access method option
WRKAREA=512,      Common work area (CWA) size in bytes
XAPPC=NO,         RACF class APPCLU required
XCMD=YES,         SPI resource security class name: RACF
```

```
         XDCT=YES,         DCT resource security class name: RACF
         XFCT=YES,         FCT resource security class name: RACF
         XJCT=YES,         JCT resource security class name: RACF
         XLT=NO,           Transaction list table option/suffix
         XPCT=YES,         PCT resource security class name: RACF
         XPPT=YES,         PPT resource security class name: RACF
         XPSB=YES,         PSB resource security class name: RACF
         XRF=NO,           Extended recovery feature (XRF) option
         XRFSOFF=NOFORCE,  XRF -  Re-sign on after takeover
         XRFSTME=5,        XRF -  sign off timeout value
         XTRAN=YES,        Transaction security class name:  RACF
         XTST=YES          TST resource security class name: RACF
END      DFHSITBA
```

Appendix B. Optional OAM and Customization

B.1 OSR Application Plan

The OSR application plan must be created if you plan to store objects without starting the OAM address space.

It consists of the following steps:

1. Creating views for dummy object storage groups
2. Create OSR application plans

B.1.1 Creating Views for Dummy Object Storage Groups

Create a job to prevent the next job, CBRIBIND, from failing. Job CBRIBIND assumes all 100 storage groups are being used and performs binds and grants to the object tables previously created. CBRIBIND as supplied, fails if all 100 object storage groups and their dedicated tables have not been created. Therefore, if you are not using all 100 storage groups, you must use one of the following procedures before running CBRIBIND:

1. First option:
   - Create a dummy object storage group table, for example, GROUPnn. The dummy storage group is allocated by modifying the supplied JCL for CBRIALC0. The database is created by modifying the JCL for CBRISQL0.
   - Views must be created to point to the GROUPNN table for any object storage group that has not been created.
2. Second option:

   Do not create a new group, but point the undefined ones to an existing table (for example, GROUP02) by copying job CBRISQL0 into the new job and make the following changes:
   - Delete all SQL statements from the first CREATE DATABASE statement up to, but not including, the first CREATE VIEW statement.
   - Provide VIEWS for the undefined groups by referring to the the existing GROUP.
   - Provide GRANTS for the undefined groups to EKCCICS.

```
CREATE VIEW
       GROUP05.V_OSM_OBJ_DIR
     AS SELECT ALL * FROM
       GROUP02.OSM_OBJ_DIR;
CREATE VIEW
       GROUP05.V_OSM_04K_OBJ_TBL
     AS SELECT ALL * FROM
       GROUP02.OSM_04K_OBJ_TBL;
CREATE VIEW
       GROUP05.V_OSM_32K_OBJ_TBL
     AS SELECT ALL * FROM
       GROUP02.OSM_32K_OBJ_TBL;
GRANT ALL ON
       GROUP05.V_OSM_OBJ_DIR
     TO EKCCICS;
GRANT ALL ON
       GROUP05.V_OSM_04K_OBJ_TBL
     TO EKCCICS;
GRANT ALL ON
       GROUP05.V_OSM_32K_OBJ_TBL
     TO EKCCICS;
 ...
 ...
 ...
 ...
CREATE VIEW
       GROUP99.V_OSM_OBJ_DIR
     AS SELECT ALL * FROM
       GROUP02.OSM_OBJ_DIR;
CREATE VIEW
       GROUP99.V_OSM_04K_OBJ_TBL
     AS SELECT ALL * FROM
       GROUP02.OSM_04K_OBJ_TBL;
CREATE VIEW
       GROUP99.V_OSM_32K_OBJ_TBL
     AS SELECT ALL * FROM
       GROUP02.OSM_32K_OBJ_TBL;
GRANT ALL ON
       GROUP99.V_OSM_OBJ_DIR
     TO EKCCICS;
GRANT ALL ON
       GROUP99.V_OSM_04K_OBJ_TBL
     TO EKCCICS;
GRANT ALL ON
       GROUP99.V_OSM_32K_OBJ_TBL
     TO EKCCICS;
```

For further information, refer to Appendix B, page 293 of the *OAM Planning, Installation and Storage Administration Guide for Object Support*.

B.1.2 Creating OSR Application Plans

Job CBRIBIND creates the plan CBRIDBS. After providing all of the views in the previous step, you can run this job with the following modifications to the JCL:

- Change the DB2 system name
- Add a JOBLIB statement to run program DSN in batch, for example:

  ```
  //JOBLIB    DD DSN=DSN310.SDSNLOAD,DISP=SHR
  ```

- Change the DBRMLIB statement to your system DBRM library for example:

  ```
  //DBRMLIB  DD  DSN=DSN310.DBRMLIB.DATA,DISP=SHR
  ```

Job CBRIGRNT grants run authority on plan CBRIDBS to PUBLIC. Besides the changes mentioned previously, you might have to modify the following as well:

- Change the PLAN name
- Change the LIB parameter to your DB2 runtime library

For further information refer to Chapter 3, page 103 of the *OAM Planning, Installation and Storage Administration Guide for Object Support*.

B.2 Defining Optical Libraries

B.2.1 Defining 9246 REAL Library

To define a REAL library and a device, use ISMF and select option 10 LIBRARY MANAGEMENT from the main menu.

The following panels describe the necessary definitions.

```
                    OPTICAL LIBRARY APPLICATION SELECTION
COMMAND ===>

TO PERFORM LIBRARY OPERATIONS, SPECIFY:

  CDS NAME            ===> 'SMS1.BASE.SCDS1'
                                (1 to 44 character data set name or 'ACTIVE')
  LIBRARY NAME        ===> REAL9246  (For Optical Library List, fully or
                                      partially specified or * for all)
  LIBRARY DEVICE TYPE ===> 9246      (For Optical Library List, fully or
                                      partially specified or * for all)
SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

 1   LIST    - Generate a list of Libraries
 2   DISPLAY - Display a Library
 3   DEFINE  - Define a Library
 4   ALTER   - Alter a Library

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
   RESPECIFY SORT CRITERIA           ===> N  (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 49. Library Application Selection Panel

```
                              9246 LIBRARY DEFINE
COMMAND ===>

SCDS NAME:    SMS1.BASE.SCDS1
LIBRARY NAME: LIBDUMMY

TO DEFINE LIBRARY, SPECIFY:

  DESCRIPTION ===> REAL 9246 OPTICAL LIBRARY
              ===>

  LIBRARY TYPE           ===> REAL             (REAL or PSEUDO)

  THE FOLLOWING FIELDS ARE FOR REAL LIBRARY TYPE ONLY:

  ONLINE STATUS          ===> N                  (Y or N)
  CURRENT PATH           ===> PRIMARY             (PRIMARY or ALTERNATE)
  PRIMARY CTC ADDRESS    ===> 320                (Valid CTC address)
  PRIMARY PORT ADDRESS   ===> 1                  (1 or 2)
  ALTERNATE CTC ADDRESS  ===> 340                (Valid CTC address)
  ALTERNATE PORT ADDRESS ===> 1                  (1 or 2)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 50. Library Define Panel

```
                     OPTICAL DRIVE APPLICATION SELECTION
COMMAND ===>

TO PERFORM DRIVE OPERATIONS, SPECIFY:

  CDS NAME          ===> 'SMS1.BASE.SCDS1'
                              (1 to 44 character data set name or 'ACTIVE')
  DRIVE NAME        ===> DRIV9246    (For Optical Drive List, fully or
                                      partially specified or * for all)
  DRIVE DEVICE TYPE ===> 9247        (For Optical Drive List, fully or
                                      partially specified or * for all)
SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

 1   LIST    - Generate a list of Drives
 2   DISPLAY - Display a Drive
 3   DEFINE  - Define a Drive
 4   ALTER   - Alter a Drive

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
   RESPECIFY SORT CRITERIA           ===> N  (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 51. Drive Application Selection Panel

```
                                9247 DRIVE DEFINE
COMMAND ===>

SCDS NAME:  SMS1.BASE.SCDS1
DRIVE NAME: DRIV9246

TO DEFINE DRIVE, SPECIFY:

  DESCRIPTION ===> Drive 9247 for library REAL9246
              ===>

  LIBRARY NAME  ===> REAL9246           (1 to 8 characters)
  DRIVE TYPE    ===> STDALONE           (LIBRARY or STDALONE)
  DRIVE NUMBER  ===>                    (0 to 3 or blank for STDALONE)
  ONLINE STATUS ===> N                  (Y or N)
  CTC ADDRESS   ===> 320                (Valid CTC address)
  SCSI ADDRESS  ===> 0                  (0 to 7)




 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 52. Drive Define Panel

B.2.2 3995 Optical Library

B.2.2.1 REAL Library

To define a REAL library and a device, use ISMF and select option 10 LIBRARY MANAGEMENT from the main menu.

The following panels describe the necessary definitions.

Although you have to define at least one library, the following panels describe the necessary definitions for one library. If an extension unit is attached to the main unit, another library must be defined using the same procedure. You have to repeat this again for the second library.

You must also define at least one pseudo drive for operator accessibility.

```
                    OPTICAL LIBRARY APPLICATION SELECTION
COMMAND ===>

TO PERFORM LIBRARY OPERATIONS, SPECIFY:

  CDS NAME            ===> 'SMS1.BASE.SCDS1'
                               (1 to 44 character data set name or 'ACTIVE')
  LIBRARY NAME        ===> REAL3995  (For Optical Library List, fully or
                                      partially specified or * for all)
  LIBRARY DEVICE TYPE ===> 3995-132  (For Optical Library List, fully or
                                      partially specified or * for all)
SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

 1   LIST    - Generate a list of Libraries
 2   DISPLAY - Display a Library
 3   DEFINE  - Define a Library
 4   ALTER   - Alter a Library

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
   RESPECIFY SORT CRITERIA           ===> N  (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 53. Library Application Selection Panel

```
                              3995 LIBRARY DEFINE
COMMAND ===>

SCDS NAME:    SMS1.BASE.SCDS1
LIBRARY NAME: REAL3995
MODEL NUMBER: 132

TO DEFINE LIBRARY, SPECIFY:

  DESCRIPTION ===> REAL 3995 DEFINITION
              ===>

  LIBRARY TYPE  ===> REAL           (REAL or PSEUDO)

  THE FOLLOWING FIELDS ARE FOR REAL LIBRARY TYPE ONLY:

  ONLINE STATUS       ===> N          (Y or N)
  CTC ADDRESS         ===> 320        (Valid CTC address)   (Model 131 or 132)
  CONTROLLING LIBRARY ===>            (Library Name)        (Model 111 or 112)



 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 54. Library Define Panel

```
                    OPTICAL DRIVE APPLICATION SELECTION
COMMAND ===>

 TO PERFORM LIBRARY OPERATIONS, SPECIFY:

   CDS NAME            ===> 'SMS1.BASE.SCDS1'
                                (1 to 44 character data set name or 'ACTIVE')
   LIBRARY NAME        ===> R3995DRV  (For Optical Library List, fully or
                                       partially specified or * for all)
   LIBRARY DEVICE TYPE ===> 3995-132  (For Optical Library List, fully or
                                       partially specified or * for all)
 SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

  1   LIST    - Generate a list of Libraries
  2   DISPLAY - Display a Library
  3   DEFINE  - Define a Library
  4   ALTER   - Alter a Library

 IF OPTION 1 CHOSEN ABOVE,
    RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
    RESPECIFY SORT CRITERIA           ===> N  (Y or N)

  F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
  F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 55. Drive Application Selection Panel

```
                                3995 DRIVE DEFINE
 COMMAND ===>

 SCDS NAME:    SMS1.BASE.SCDS1
 DRIVE NAME:   R3995DRV
 MODEL NUMBER: 132

 TO DEFINE DRIVE, SPECIFY:

   DESCRIPTION ===> DUMMY 3995 DRIVE for REAL3995 LIBRARY
               ===>

   LIBRARY NAME      ===> REAL3995             (1 to 8 characters)
   DRIVE NUMBER      ===> 1                    (1 to 4 for LIBRARY, 5 for
                                                OPERATOR ACCESSIBLE)
   ONLINE STATUS     ===> N                    (Y or N)

 THE FOLLOWING FIELD IS FOR OPERATOR ACCESSIBLE DRIVE TYPE ONLY:

   REAL LIBRARY NAME ===>                      (1 to 8 characters)


  F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
  F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 56. Drive Define Panel

B.2.2.2 PSEUDO Library

Use ISMF to define a pseudo library for operator accessibility. Select option 10 LIBRARY MANAGEMENT from the main menu.

The following panels show the required definitions.

```
                    OPTICAL LIBRARY APPLICATION SELECTION
COMMAND ===>

TO PERFORM LIBRARY OPERATIONS, SPECIFY:

  CDS NAME            ===> 'SMS1.BASE.SCDS1'
                                (1 to 44 character data set name or 'ACTIVE')
  LIBRARY NAME        ===> DUM3995   (For Optical Library List, fully or
                                      partially specified or * for all)
  LIBRARY DEVICE TYPE ===> 3995-132  (For Optical Library List, fully or
                                      partially specified or * for all)
SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

 1   LIST    - Generate a list of Libraries
 2   DISPLAY - Display a Library
 3   DEFINE  - Define a Library
 4   ALTER   - Alter a Library

IF OPTION 1 CHOSEN ABOVE,
   RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
   RESPECIFY SORT CRITERIA           ===> N  (Y or N)

 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 57. Library Application Selection Panel

```
                              3995 LIBRARY DEFINE
COMMAND ===>

SCDS NAME:    SMS1.BASE.SCDS1
LIBRARY NAME: DUM3995
MODEL NUMBER: 132

TO DEFINE LIBRARY, SPECIFY:

  DESCRIPTION ===> DUMMY 3995 DEFINITION
              ===>

  LIBRARY TYPE  ===> PSEUDO         (REAL or PSEUDO)

  THE FOLLOWING FIELDS ARE FOR REAL LIBRARY TYPE ONLY:

  ONLINE STATUS       ===>            (Y or N)
  CTC ADDRESS         ===>            (Valid CTC address)   (Model 131 or 132)
  CONTROLLING LIBRARY ===>            (Library Name)        (Model 111 or 112)



 F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
 F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 58. Library Define Panel

```
                    OPTICAL DRIVE APPLICATION SELECTION
COMMAND ===>

 TO PERFORM LIBRARY OPERATIONS, SPECIFY:

   CDS NAME            ===> 'SMS1.BASE.SCDS1'
                                (1 to 44 character data set name or 'ACTIVE')
   LIBRARY NAME        ===> D3995DRV  (For Optical Library List, fully or
                                       partially specified or * for all)
   LIBRARY DEVICE TYPE ===> 3995-132  (For Optical Library List, fully or
                                       partially specified or * for all)
 SELECT ONE OF THE FOLLOWING OPTIONS  ===> 3

  1   LIST    - Generate a list of Libraries
  2   DISPLAY - Display a Library
  3   DEFINE  - Define a Library
  4   ALTER   - Alter a Library

 IF OPTION 1 CHOSEN ABOVE,
    RESPECIFY VIEW CRITERIA           ===> N  (Y or N)
    RESPECIFY SORT CRITERIA           ===> N  (Y or N)

  F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
  F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 59. Drive Application Selection Panel

```
                                3995 DRIVE DEFINE
 COMMAND ===>

 SCDS NAME:    SMS1.BASE.SCDS1
 DRIVE NAME:   D3995DRV
 MODEL NUMBER: 132

 TO DEFINE DRIVE, SPECIFY:

   DESCRIPTION ===> DUMMY 3995 DRIVE for DUM3995 LIBRARY
               ===>

   LIBRARY NAME      ===> DUM3995             (1 to 8 characters)
   DRIVE NUMBER      ===> 5                    (1 to 4 for LIBRARY, 5 for
                                                OPERATOR ACCESSIBLE)
   ONLINE STATUS     ===> N                    (Y or N)

 THE FOLLOWING FIELD IS FOR OPERATOR ACCESSIBLE DRIVE TYPE ONLY:

   REAL LIBRARY NAME ===> REAL3995             (1 to 8 characters)


  F1=HELP    F2=SPLIT   F3=END     F4=RETURN  F5=        F6=        F7=UP
  F8=DOWN    F9=SWAP    F10=LEFT   F11=RIGHT  F12=CURSOR
```

Figure 60. Drive Define Panel

Appendix C. Table Descriptions

C.1 Library Server Table

Table 13 (Page 1 of 2). Library Server Table Jobs Description

TableSpace Name	Table Name	Unique Index Name(s)	Index Name(s)	Description
ACCDTS	FRNACCESSCODES	ACCD1IXU ACCD2IXU		Access Codes Table
ACLSTS	FRNACCESSLIST	ACLS1IXU		Access List Table
ATTRTS	FRNATTRDEFS	ATTR1IXU		Attribute Definition Table
CKOTTS	FRNCHECKEDOUT	CKOT1IXU	CKOT2IX	Checked Out Table
CLATTS	FRNCLASSATTRS	CLAT1IXU CLAT2IXU		Class Attributes Table
CLDFTS	FRNCLASSDEFS	CLDF1IXU		Class Definitions Table
CNTLTS	FRNCNTL	CNTL1IXU		Control Table
COLLTS	FRNCOLLNAME	COLL1IXU COLL2IXU		Collection Names Table
DESCTS	FRNDESCRIPTIONS	DESC1IXU		Description Table
EVNTTS	FRNEVENTS		EVNT1IX	Events Table
FORPTS	FRNFOREIGNPARTSNMS	FORP1IXU FORP2IXU		Foreign Parts Names Table
GRAVTS	FRNGRAVEYARD			Graveyard Table
ITEMTS	FRNITEMS	ITEM1IXU ITEM2IXU		Items Table
LINKTS	FRNLINKS	LINK3IXU	LINK1IX LINK2IX	Links Table
LKEYTS	FRNLICENSEKEY			License Key Table
LVIOTS	FRNLICVIOLATIONS			License Violations Table
NAMETS	FRNNAMES	NAME1IXU	NAME2IX	Names Table
NLSKTS	FRNNLSKEYWORDS	NLSK1IXU NLSK2IXU		NLS Keywords Table
NLSLTS	FRNNLSLANGUAGES	NLSL1IXU NLSL2IXU		NLS Languages Table
OBJSTS	FRNOBJECTSERVER	OBJS1IXU OBJS2IXU		Object Server Table
PARTTS	FRNPARTS	PART1IXU		Parts Table
PRIVTS	FRNPRIVILEGES	PRIV1IXU PRIV2IXU		Privileges Table
PROPTS	FRNPROPERTIES	PROP1IXU		Properties Table
PTGPTS	FRNPATRONGROUP	PTGP1IXU		
PTRNTS	FRNPATRONS	PTRN1IXU PTRN2IXU		Patron Group Table

Table 13 (Page 2 of 2). Library Server Table Jobs Description				
TableSpace Name	**Table Name**	**Unique Index Name(s)**	**Index Name(s)**	**Description**
QRYXTS	FRNQUERYEXPRS	QRYX1IXU QRYX2IXU		Query Expressions Table
SQRYTS	FRNSTATICQUERIES	SQRY1IXU		Static Queries Table
VWATTS	FRNVIEWATTRS	VWAT1IXU VWAT2IXU		View Attributes Table
VWDFTS	FRNVIEWDEFS	VWDF1IXU		View Definitions Table
WPITTS	FRNWIPITEMS	WPIT1IXU	WPIT2IXU	Work-in-Process Items Table
WPSUTS	FRNWIPSUSPITEMS	WPSU1IXU		Suspended Items Table

Note: Detailed information for each table can be found in the *ImagePlus VisualInfo Application Programming Reference, Volume 3: Common Data Structures and Database Tables*.

C.2 Folder Manager Table

Table 14. Folder Manager Table Description				
TableSpace Name	**Table Name**	**Unique Index Name(s)**	**View Name(s)**	**Description**
AVT01TS	AVT00001	IXT00001, AVT01IXU	ICV00001	Contents Index Class
AVT02TS	AVT00002	IXT00002, AVT02IXU	ICV00002	Workbasket Index Class
AVT03TS	AVT00003	IXT00003, AVT03IXU	ICV00003	Workflow Index Class
AVT04TS	AVT00004	IXT00004	ICV00004	Information about index classes defined in the system
AVT05TS	AVT00005	IXT00005	ICV00005	Information about index class views defined in the system
AVT06TS	AVT00006	IXT00006	ICV00006	NOINDEX Index class
AVT07TS	AVT00007	IXT00007	ICV00007	Information about search result folders

Appendix D. Sample RDO Definitions

D.1 FFST RDO Definitions

```
*****************************************************************      00160000
*                                                               *      00170000
*   FFST/MVS  - CICS CSD Definitions                            *      00180000
*                                                               *      00190000
*****************************************************************      00200000
*                                                                      00210000
*                                                                      00220000
* FFST/MVS PPT PCT CSD Start                                           00230000
*                                                                      00240000
 DELETE GROUP(EPWFFST)                                                00250002
*                                                                      00260000
 DEFINE PROGRAM(EPWTRUEI) GROUP(EPWFFST) LANGUAGE(ASSEMBLER)          00270000
 DEFINE PROGRAM(EPWTRUET) GROUP(EPWFFST) LANGUAGE(ASSEMBLER)          00271000
 DEFINE PROGRAM(EPWTRUE)  GROUP(EPWFFST) LANGUAGE(ASSEMBLER)          00272000
        RESIDENT(YES)                                                 00272100
 DEFINE PROGRAM(EPWCIVP)  GROUP(EPWFFST) LANGUAGE(ASSEMBLER)          00273000
 DEFINE TRANSACTION(EIVP) PROGRAM(EPWCIVP) GROUP(EPWFFST)             00274001
 DEFINE TRANSACTION(EPWT) PROGRAM(EPWTRUET) GROUP(EPWFFST)            00274101
*                                                                     00274203
* APAR PN45724                                                        00274303
*                                                                     00274403
*DEFINE PROGRAM(EPWCGUEI) GROUP(EPWFFST) LANGUAGE(ASSEMBLER)         00274503
*DEFINE PROGRAM(EPWCGUET) GROUP(EPWFFST) LANGUAGE(ASSEMBLER)         00274603
*DEFINE PROGRAM(EPWCGUE)  GROUP(EPWFFST) LANGUAGE(ASSEMBLER)         00274703
*       RESIDENT(YES)                                                00274803
*DEFINE TRANSACTION(GUET) PROGRAM(EPWCGUET) GROUP(EPWFFST)           00274903
*                                                                     02170000
*                                                                     02173000
* FFST/MVS CSD End                                                    02180003
*                                                                     02190000
```

D.2 PL/I Definition

```
PPTPL    TITLE 'DFHPPT - FOR PL/I TRANSIENT LIBRARY'                  00010204
*                                                                     00010301
         PRINT NOGEN                                                  00010401
         DFHPPT TYPE=INITIAL,                                        X00010501
               SUFFIX=PL                                              00010602
*                                                                     00010701
********************************************************************* 00010801
* ./       ADD   LIST=ALL                                             00011000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBCCLA      Complex String Directo 00020003
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBCCRA      Non-comlex str dir     00030000
*                                                                     00040000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBEOCA      ON-CODE builtin        00050000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFEFCA      Print COUNT Tables     00060000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFESMA      Error Message Module   00070000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFESNA      Error Message Module   00080000
*                                                                     00090000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFKCSA      PLIDUMP routines       00100000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFKMRA      ....                   00110000
```

```
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFKPTA        ....                00120000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFKTBA          ....              00130000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFKTCA            ....            00140000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFKTRA              ....          00150000
*                                                                       00160000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFPGDA      Storage Management     00170000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFPMRA      REPORT module          00180000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMFSTVA      Stream Transmitter     00190000
*                                                                       00200000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMOMGF       Message formatter      00210000
*                                                                       00220000
*   Language tables for Upper Case English                              00230000
*                                                                       00240000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBLANU      Language Appendage     00250000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBLNTU      Language Table         00260000
*                                                                       00270000
*   Message modules for Upper Case English                              00280000
*                                                                       00290000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMOLMSU      Messages Text File     00300000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMOLM2U      Messages Text File     00310000
*                                                                       00320000
*---------------------------------------------------------------------  00330000
*   The following entries are optional and may be used only if          00340000
*   those particular languages have been installed on your system.      00350000
*---------------------------------------------------------------------  00360000
*                                                                       00370000
*   Language tables for Mixed Case English                              00380000
*                                                                       00390000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBLANA      Language Appendage     00400000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBLNTA      Language Table         00410000
*                                                                       00420000
*   Message modules for Mixed Case English                              00430000
*                                                                       00440000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMOLMSA      Messages Text File     00450000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMOLM2A      Messages Text File     00460000
*                                                                       00470000
*   Language tables for Japanese                                        00480000
*                                                                       00490000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBLANN      Language Appendage     00500000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMBLNTN      Language Table         00510000
*                                                                       00520000
*   Message modules for Japanese                                        00530000
*                                                                       00540000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMOLMSN      Messages Text File     00550000
         DFHPPT TYPE=ENTRY,PROGRAM=IBMOLM2N      Messages Text File     00560000
********************************************************************** 00561001
*                                                                       00570001
         DFHPPT TYPE=FINAL                                              00580001
         END                                                            00590001
```

D.3 Additional PL/I RDO Definition

```
 DEFINE PROGRAM(IBMEPGDA) GROUP(PLIMISS) LANGUAGE(ASSEMBLER)            00270001
        RELOAD(NO)   RESIDENT(NO) STATUS(ENABLED) CEDF(NO)              00271001
 DEFINE PROGRAM(IBMEPMRA) GROUP(PLIMISS) LANGUAGE(ASSEMBLER)            00272001
        RELOAD(NO)   RESIDENT(NO) STATUS(ENABLED) CEDF(NO)              00273001
```

D.4 C/370 PL/I RDO Definition

```
************************************************************************ 00010000
*  copied from EDC.V2R2M0.SEDCJCL1(EDCCCSD)                            * 00020000
************************************************************************ 00030000
*                                                                      * 00040000
************************************************************************ 00050000
*    LICENSED MATERIALS - PROPERTY OF IBM.                             * 00060000
*                                                                      * 00070000
*    5688-188 (C) COPYRIGHT IBM CORP. 1988, 1993                       * 00080000
*    ALL RIGHTS RESERVED                                               * 00090000
*    US GOVERNMENT USERS RESTRICTED RIGHTS - USE,                      * 00100000
*    DUPLICATION OR DISCLOSURE RESTRICTED BY GSA                       * 00110000
*    ADP SCHEDULE CONTRACT WITH IBM CORP.                              * 00120000
*                                                                      * 00130000
*    SEE COPYRIGHT INSTRUCTIONS                                        * 00140000
************************************************************************ 00150000
************************************************************************ 00160000
*                                                                      * 00170000
*   C/370 - CICS CSD Definitions                                       * 00180000
*                                                                      * 00190000
************************************************************************ 00200000
*                                                                        00210000
*                                                                        00220000
* C/370 CSD Start                                                       00230000
*                                                                       00240000
* C/370 specific run time routines (V2R2)                               00250000
*                                                                       00260000
 DEFINE PROGRAM(EDCXV)    GROUP(EDC) LANGUAGE(ASSEMBLER)                00270000
 DEFINE PROGRAM(EDC$LCNM) GROUP(EDC) LANGUAGE(ASSEMBLER)                00280000
 DEFINE PROGRAM(EDCUCSNM) GROUP(EDC) LANGUAGE(ASSEMBLER)                00290000
*                                                                       00300000
* C/370 compatibility locales                                           00310000
*                                                                       00320000
 DEFINE PROGRAM(EDC$FRAN) GROUP(EDC) LANGUAGE(ASSEMBLER)                00330000
 DEFINE PROGRAM(EDC$GERM) GROUP(EDC) LANGUAGE(ASSEMBLER)                00340000
 DEFINE PROGRAM(EDC$ITAL) GROUP(EDC) LANGUAGE(ASSEMBLER)                00350000
 DEFINE PROGRAM(EDC$SPAI) GROUP(EDC) LANGUAGE(ASSEMBLER)                00360000
 DEFINE PROGRAM(EDC$S370) GROUP(EDC) LANGUAGE(ASSEMBLER)                00370000
 DEFINE PROGRAM(EDC$UK)   GROUP(EDC) LANGUAGE(ASSEMBLER)                00380000
 DEFINE PROGRAM(EDC$USA)  GROUP(EDC) LANGUAGE(ASSEMBLER)                00390000
*                                                                       00400000
* C/370 POSIX-style locales                                             00410000
*                                                                       00420000
 DEFINE PROGRAM(EDC$DAEE) GROUP(EDC) LANGUAGE(ASSEMBLER)                00430000
 DEFINE PROGRAM(EDC$DAEY) GROUP(EDC) LANGUAGE(ASSEMBLER)                00440000
 DEFINE PROGRAM(EDC$DCEO) GROUP(EDC) LANGUAGE(ASSEMBLER)                00450000
 DEFINE PROGRAM(EDC$DCEY) GROUP(EDC) LANGUAGE(ASSEMBLER)                00460000
 DEFINE PROGRAM(EDC$DDEB) GROUP(EDC) LANGUAGE(ASSEMBLER)                00470000
 DEFINE PROGRAM(EDC$DDEY) GROUP(EDC) LANGUAGE(ASSEMBLER)                00480000
 DEFINE PROGRAM(EDC$EJEX) GROUP(EDC) LANGUAGE(ASSEMBLER)                00490000
 DEFINE PROGRAM(EDC$EKEK) GROUP(EDC) LANGUAGE(ASSEMBLER)                00500000
 DEFINE PROGRAM(EDC$EKEY) GROUP(EDC) LANGUAGE(ASSEMBLER)                00510000
 DEFINE PROGRAM(EDC$ELES) GROUP(EDC) LANGUAGE(ASSEMBLER)                00520000
 DEFINE PROGRAM(EDC$ESEJ) GROUP(EDC) LANGUAGE(ASSEMBLER)                00530000
 DEFINE PROGRAM(EDC$ESEY) GROUP(EDC) LANGUAGE(ASSEMBLER)                00540000
 DEFINE PROGRAM(EDC$EUEA) GROUP(EDC) LANGUAGE(ASSEMBLER)                00550000
 DEFINE PROGRAM(EDC$EUEY) GROUP(EDC) LANGUAGE(ASSEMBLER)                00560000
 DEFINE PROGRAM(EDC$FBEO) GROUP(EDC) LANGUAGE(ASSEMBLER)                00570000
```

```
 DEFINE PROGRAM(EDC$FBEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00580000
 DEFINE PROGRAM(EDC$FCEA) GROUP(EDC) LANGUAGE(ASSEMBLER)              00590000
 DEFINE PROGRAM(EDC$FCEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00600000
 DEFINE PROGRAM(EDC$FFEM) GROUP(EDC) LANGUAGE(ASSEMBLER)              00610000
 DEFINE PROGRAM(EDC$FFEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00620000
 DEFINE PROGRAM(EDC$FIEF) GROUP(EDC) LANGUAGE(ASSEMBLER)              00630000
 DEFINE PROGRAM(EDC$FIEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00640000
 DEFINE PROGRAM(EDC$FSEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              00650000
 DEFINE PROGRAM(EDC$FSEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00660000
 DEFINE PROGRAM(EDC$ISER) GROUP(EDC) LANGUAGE(ASSEMBLER)              00670000
 DEFINE PROGRAM(EDC$ISEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00680000
 DEFINE PROGRAM(EDC$ITEG) GROUP(EDC) LANGUAGE(ASSEMBLER)              00690000
 DEFINE PROGRAM(EDC$ITEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00700000
 DEFINE PROGRAM(EDC$JAEL) GROUP(EDC) LANGUAGE(ASSEMBLER)              00710000
 DEFINE PROGRAM(EDC$JAEU) GROUP(EDC) LANGUAGE(ASSEMBLER)              00720000
 DEFINE PROGRAM(EDC$JAEV) GROUP(EDC) LANGUAGE(ASSEMBLER)              00730000
 DEFINE PROGRAM(EDC$JAEX) GROUP(EDC) LANGUAGE(ASSEMBLER)              00740000
 DEFINE PROGRAM(EDC$NBEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              00750000
 DEFINE PROGRAM(EDC$NBEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00760000
 DEFINE PROGRAM(EDC$NNEA) GROUP(EDC) LANGUAGE(ASSEMBLER)              00770000
 DEFINE PROGRAM(EDC$NNEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00780000
 DEFINE PROGRAM(EDC$NOEE) GROUP(EDC) LANGUAGE(ASSEMBLER)              00790000
 DEFINE PROGRAM(EDC$NOEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00800000
 DEFINE PROGRAM(EDC$PTEA) GROUP(EDC) LANGUAGE(ASSEMBLER)              00810000
 DEFINE PROGRAM(EDC$PTEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00820000
 DEFINE PROGRAM(EDC$SVEF) GROUP(EDC) LANGUAGE(ASSEMBLER)              00830000
 DEFINE PROGRAM(EDC$SVEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00840000
 DEFINE PROGRAM(EDC$TREW) GROUP(EDC) LANGUAGE(ASSEMBLER)              00850000
*                                                                     00860000
* C/370 Code set converters                                          00870000
*                                                                     00880000
 DEFINE PROGRAM(EDCUAAEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              00890000
 DEFINE PROGRAM(EDCUABEL) GROUP(EDC) LANGUAGE(ASSEMBLER)              00900000
 DEFINE PROGRAM(EDCUABEN) GROUP(EDC) LANGUAGE(ASSEMBLER)              00910000
 DEFINE PROGRAM(EDCUABEU) GROUP(EDC) LANGUAGE(ASSEMBLER)              00920000
 DEFINE PROGRAM(EDCUABEV) GROUP(EDC) LANGUAGE(ASSEMBLER)              00930000
 DEFINE PROGRAM(EDCUABEX) GROUP(EDC) LANGUAGE(ASSEMBLER)              00940000
 DEFINE PROGRAM(EDCUACEL) GROUP(EDC) LANGUAGE(ASSEMBLER)              00950000
 DEFINE PROGRAM(EDCUACEN) GROUP(EDC) LANGUAGE(ASSEMBLER)              00960000
 DEFINE PROGRAM(EDCUACEU) GROUP(EDC) LANGUAGE(ASSEMBLER)              00970000
 DEFINE PROGRAM(EDCUACEV) GROUP(EDC) LANGUAGE(ASSEMBLER)              00980000
 DEFINE PROGRAM(EDCUACEX) GROUP(EDC) LANGUAGE(ASSEMBLER)              00990000
 DEFINE PROGRAM(EDCUEAEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01000000
 DEFINE PROGRAM(EDCUEAEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01010000
 DEFINE PROGRAM(EDCUEAI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01020000
 DEFINE PROGRAM(EDCUEBEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01030000
 DEFINE PROGRAM(EDCUEBEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01040000
 DEFINE PROGRAM(EDCUEBI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01050000
 DEFINE PROGRAM(EDCUECEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01060000
 DEFINE PROGRAM(EDCUECEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01070000
 DEFINE PROGRAM(EDCUECI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01080000
 DEFINE PROGRAM(EDCUEDEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01090000
 DEFINE PROGRAM(EDCUEDEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01100000
 DEFINE PROGRAM(EDCUEDI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01110000
 DEFINE PROGRAM(EDCUEEEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01120000
 DEFINE PROGRAM(EDCUEEEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01130000
 DEFINE PROGRAM(EDCUEEI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01140000
 DEFINE PROGRAM(EDCUEFEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01150000
 DEFINE PROGRAM(EDCUEFEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01160000
```

```
DEFINE PROGRAM(EDCUEFI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01170000
DEFINE PROGRAM(EDCUEGEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01180000
DEFINE PROGRAM(EDCUEGEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01190000
DEFINE PROGRAM(EDCUEGI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01200000
DEFINE PROGRAM(EDCUEHEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01210000
DEFINE PROGRAM(EDCUEHEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01220000
DEFINE PROGRAM(EDCUEHI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01230000
DEFINE PROGRAM(EDCUEIEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01240000
DEFINE PROGRAM(EDCUEIEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01250000
DEFINE PROGRAM(EDCUEII1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01260000
DEFINE PROGRAM(EDCUEJEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01270000
DEFINE PROGRAM(EDCUEJEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01280000
DEFINE PROGRAM(EDCUEJI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01290000
DEFINE PROGRAM(EDCUEKEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01300000
DEFINE PROGRAM(EDCUEKEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01310000
DEFINE PROGRAM(EDCUEKI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01320000
DEFINE PROGRAM(EDCUELAB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01330000
DEFINE PROGRAM(EDCUELAC) GROUP(EDC) LANGUAGE(ASSEMBLER)              01340000
DEFINE PROGRAM(EDCUELEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01350000
DEFINE PROGRAM(EDCUELEX) GROUP(EDC) LANGUAGE(ASSEMBLER)              01360000
DEFINE PROGRAM(EDCUELEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01370000
DEFINE PROGRAM(EDCUELI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01380000
DEFINE PROGRAM(EDCUEMEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01390000
DEFINE PROGRAM(EDCUEMEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01400000
DEFINE PROGRAM(EDCUEMI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01410000
DEFINE PROGRAM(EDCUENAB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01420000
DEFINE PROGRAM(EDCUENAC) GROUP(EDC) LANGUAGE(ASSEMBLER)              01430000
DEFINE PROGRAM(EDCUEOEA) GROUP(EDC) LANGUAGE(ASSEMBLER)              01440000
DEFINE PROGRAM(EDCUEOEB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01450000
DEFINE PROGRAM(EDCUEOEC) GROUP(EDC) LANGUAGE(ASSEMBLER)              01460000
DEFINE PROGRAM(EDCUEOED) GROUP(EDC) LANGUAGE(ASSEMBLER)              01470000
DEFINE PROGRAM(EDCUEOEE) GROUP(EDC) LANGUAGE(ASSEMBLER)              01480000
DEFINE PROGRAM(EDCUEOEF) GROUP(EDC) LANGUAGE(ASSEMBLER)              01490000
DEFINE PROGRAM(EDCUEOEG) GROUP(EDC) LANGUAGE(ASSEMBLER)              01500000
DEFINE PROGRAM(EDCUEOEH) GROUP(EDC) LANGUAGE(ASSEMBLER)              01510000
DEFINE PROGRAM(EDCUEOEI) GROUP(EDC) LANGUAGE(ASSEMBLER)              01520000
DEFINE PROGRAM(EDCUEOEJ) GROUP(EDC) LANGUAGE(ASSEMBLER)              01530000
DEFINE PROGRAM(EDCUEOEK) GROUP(EDC) LANGUAGE(ASSEMBLER)              01540000
DEFINE PROGRAM(EDCUEOEL) GROUP(EDC) LANGUAGE(ASSEMBLER)              01550000
DEFINE PROGRAM(EDCUEOEM) GROUP(EDC) LANGUAGE(ASSEMBLER)              01560000
DEFINE PROGRAM(EDCUEOER) GROUP(EDC) LANGUAGE(ASSEMBLER)              01570000
DEFINE PROGRAM(EDCUEOEX) GROUP(EDC) LANGUAGE(ASSEMBLER)              01580000
DEFINE PROGRAM(EDCUEOEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01590000
DEFINE PROGRAM(EDCUEOI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01600000
DEFINE PROGRAM(EDCUEREO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01610000
DEFINE PROGRAM(EDCUEREY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01620000
DEFINE PROGRAM(EDCUERI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01630000
DEFINE PROGRAM(EDCUESEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01640000
DEFINE PROGRAM(EDCUESI7) GROUP(EDC) LANGUAGE(ASSEMBLER)              01650000
DEFINE PROGRAM(EDCUEUAB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01660000
DEFINE PROGRAM(EDCUEUAC) GROUP(EDC) LANGUAGE(ASSEMBLER)              01670000
DEFINE PROGRAM(EDCUEVAB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01680000
DEFINE PROGRAM(EDCUEVAC) GROUP(EDC) LANGUAGE(ASSEMBLER)              01690000
DEFINE PROGRAM(EDCUEWEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01700000
DEFINE PROGRAM(EDCUEWI9) GROUP(EDC) LANGUAGE(ASSEMBLER)              01710000
DEFINE PROGRAM(EDCUEXAB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01720000
DEFINE PROGRAM(EDCUEXAC) GROUP(EDC) LANGUAGE(ASSEMBLER)              01730000
DEFINE PROGRAM(EDCUEXEL) GROUP(EDC) LANGUAGE(ASSEMBLER)              01740000
DEFINE PROGRAM(EDCUEXEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01750000
```

```
 DEFINE PROGRAM(EDCUEXEY) GROUP(EDC) LANGUAGE(ASSEMBLER)              01760000
 DEFINE PROGRAM(EDCUEXI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01770000
 DEFINE PROGRAM(EDCUEYAA) GROUP(EDC) LANGUAGE(ASSEMBLER)              01780000
 DEFINE PROGRAM(EDCUEYEA) GROUP(EDC) LANGUAGE(ASSEMBLER)              01790000
 DEFINE PROGRAM(EDCUEYEB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01800000
 DEFINE PROGRAM(EDCUEYEC) GROUP(EDC) LANGUAGE(ASSEMBLER)              01810000
 DEFINE PROGRAM(EDCUEYED) GROUP(EDC) LANGUAGE(ASSEMBLER)              01820000
 DEFINE PROGRAM(EDCUEYEE) GROUP(EDC) LANGUAGE(ASSEMBLER)              01830000
 DEFINE PROGRAM(EDCUEYEF) GROUP(EDC) LANGUAGE(ASSEMBLER)              01840000
 DEFINE PROGRAM(EDCUEYEG) GROUP(EDC) LANGUAGE(ASSEMBLER)              01850000
 DEFINE PROGRAM(EDCUEYEH) GROUP(EDC) LANGUAGE(ASSEMBLER)              01860000
 DEFINE PROGRAM(EDCUEYEI) GROUP(EDC) LANGUAGE(ASSEMBLER)              01870000
 DEFINE PROGRAM(EDCUEYEJ) GROUP(EDC) LANGUAGE(ASSEMBLER)              01880000
 DEFINE PROGRAM(EDCUEYEK) GROUP(EDC) LANGUAGE(ASSEMBLER)              01890000
 DEFINE PROGRAM(EDCUEYEL) GROUP(EDC) LANGUAGE(ASSEMBLER)              01900000
 DEFINE PROGRAM(EDCUEYEM) GROUP(EDC) LANGUAGE(ASSEMBLER)              01910000
 DEFINE PROGRAM(EDCUEYEO) GROUP(EDC) LANGUAGE(ASSEMBLER)              01920000
 DEFINE PROGRAM(EDCUEYER) GROUP(EDC) LANGUAGE(ASSEMBLER)              01930000
 DEFINE PROGRAM(EDCUEYES) GROUP(EDC) LANGUAGE(ASSEMBLER)              01940000
 DEFINE PROGRAM(EDCUEYEW) GROUP(EDC) LANGUAGE(ASSEMBLER)              01950000
 DEFINE PROGRAM(EDCUEYEX) GROUP(EDC) LANGUAGE(ASSEMBLER)              01960000
 DEFINE PROGRAM(EDCUEYI1) GROUP(EDC) LANGUAGE(ASSEMBLER)              01970000
 DEFINE PROGRAM(EDCUI1EA) GROUP(EDC) LANGUAGE(ASSEMBLER)              01980000
 DEFINE PROGRAM(EDCUI1EB) GROUP(EDC) LANGUAGE(ASSEMBLER)              01990000
 DEFINE PROGRAM(EDCUI1EC) GROUP(EDC) LANGUAGE(ASSEMBLER)              02000000
 DEFINE PROGRAM(EDCUI1ED) GROUP(EDC) LANGUAGE(ASSEMBLER)              02010000
 DEFINE PROGRAM(EDCUI1EE) GROUP(EDC) LANGUAGE(ASSEMBLER)              02020000
 DEFINE PROGRAM(EDCUI1EF) GROUP(EDC) LANGUAGE(ASSEMBLER)              02030000
 DEFINE PROGRAM(EDCUI1EG) GROUP(EDC) LANGUAGE(ASSEMBLER)              02040000
 DEFINE PROGRAM(EDCUI1EH) GROUP(EDC) LANGUAGE(ASSEMBLER)              02050000
 DEFINE PROGRAM(EDCUI1EI) GROUP(EDC) LANGUAGE(ASSEMBLER)              02060000
 DEFINE PROGRAM(EDCUI1EJ) GROUP(EDC) LANGUAGE(ASSEMBLER)              02070000
 DEFINE PROGRAM(EDCUI1EK) GROUP(EDC) LANGUAGE(ASSEMBLER)              02080000
 DEFINE PROGRAM(EDCUI1EL) GROUP(EDC) LANGUAGE(ASSEMBLER)              02090000
 DEFINE PROGRAM(EDCUI1EM) GROUP(EDC) LANGUAGE(ASSEMBLER)              02100000
 DEFINE PROGRAM(EDCUI1EO) GROUP(EDC) LANGUAGE(ASSEMBLER)              02110000
 DEFINE PROGRAM(EDCUI1ER) GROUP(EDC) LANGUAGE(ASSEMBLER)              02120000
 DEFINE PROGRAM(EDCUI1EX) GROUP(EDC) LANGUAGE(ASSEMBLER)              02130000
 DEFINE PROGRAM(EDCUI1EY) GROUP(EDC) LANGUAGE(ASSEMBLER)              02140000
 DEFINE PROGRAM(EDCUI7ES) GROUP(EDC) LANGUAGE(ASSEMBLER)              02150000
 DEFINE PROGRAM(EDCUI9EW) GROUP(EDC) LANGUAGE(ASSEMBLER)              02160000
*                                                                     02170000
* C/370 CSD End                                                       02180000
*                                                                     02190000
```

Appendix E. OS/2 Workstation Files

This workstation contains a VisualInfo client and the Configuration Server.

E.1 CONFIG.SYS for Workstation

```
IFS=C:\OS2\HPFS.IFS /CACHE:2048 /CRECL:24 /A:DE
SET SQLNETB=16
PROTSHELL=C:\OS2\PMSHELL.EXE
SET USER_INI=C:\OS2\OS2.INI
SET SYSTEM_INI=C:\OS2\OS2SYS.INI
SET OS2_SHELL=C:\OS2\CMD.EXE
SET AUTOSTART=PROGRAMS,TASKLIST,FOLDERS,CONNECTIONS
SET RUNWORKPLACE=C:\OS2\PMSHELL.EXE
SET RESTARTOBJECTS=STARTUPFOLDERSONLY
SET COMSPEC=C:\OS2\CMD.EXE
LIBPATH=D:\SQLLIB\DLL;C:\MUGLIB\DLL;C:\IBMLAN\NETLIB;
 C:\IBMCOM\DLL;.;E:\FTW;E:\FTW;V:\VIZ11;
 D:\SQLLIB\DCSLIB;D:\TOOLS;D:\FRNV1RO\DLL;C:\OS2\DLL;D:\LAD2\LADDLL;
 C:\CMLIB\DLL;C:\OS2\MDOS;C:\;C:\OS2\APPS\DLL;C:\TOOLKT21\DLL;
 C:\IBMCPP\DLL;X:\DLL;D:\PERF_30;D:\PCSOS2;
SET
PATH=D:\SQLLIB;C:\IBMLAN\NETPROG;C:\MUGLIB;E:\FTW;V:\VIZ11;E:\FTW;
 D:\FRNV1RO\DLL;D:\TOOLS;D:\FRNV1RO;C:\OS2;C:\CMLIB;C:\OS2\SYSTEM;
 C:\OS2\MDOS\WINOS2;C:\OS2\INSTALL;D:\LAD2;D:\LAD2\LADCODE;
 D:\LAD2\LADTOOLS;C:\;C:\OS2\MDOS;C:\OS2\APPS;C:\TOOLKT21\OS2BIN;
 C:\IBMCPP\BIN;X\BIN;D:\PERF_30;D:\PCSOS2;
SET
DPATH=D:\SQLLIB;C:\IBMLAN\NETPROG;C:\IBMLAN;
 C:\MUGLIB;C:\IBMCOM;E:\FTW;V:\VIZ11;E:\FTW;D:\TOOLS;
 D:\FRNV1RO\DLL;D:\FRNV1RO;C:\OS2;C:\CMLIB;C:\OS2\SYSTEM;
 C:\OS2\MDOS\WINOS2;C:\OS2\INSTALL;D:\SERVER;C:\;C:\OS2\BITMAP;
 C:\OS2\MDOS;C:\OS2\APPS;C:\IBMCPP\LOCALE;C:\IBMCPP\HELP;
 C:\IBMCPP\SYS;C:\IBM386FS;
SET PROMPT=$i·$p“
SET HELP=D:\SQLLIB;e:\ftw;v:\viz11;E:\FTW;
 D:\FRNV1RO\HELP;D:\FRNV1RO;C:\OS2\HELP;
 C:\OS2\HELP\TUTORIAL;C:\TOOLKT21\OS2HELP;
 C:\IBMCPP\HELP;X:\HELP;C:\CMLIB;
SET FTB1PATH=e:\ftw;v:\viz11
SET FTB1DIR=e:\ftw\WORK
SET FTBBASE=e:\ftw
REM *Query_for_OS/2_(Requester)* SET FTB1PATH=E:\FTW
REM *Query_for_OS/2_(Requester)* SET FTB1DIR=E:\FTW\WORK
REM *Query_for_OS/2_(Requester)* SET FTBBASE=E:\FTW
SET GLOSSARY=C:\OS2\HELP\GLOSS;
SET IPF_KEYS=SBCS
PRIORITY_DISK_IO=YES
FILES=40
DEVICE=C:\IBMCOM\PROTOCOL\LANPDD.OS2
DEVICE=C:\IBMCOM\PROTOCOL\LANVDD.OS2
DEVICE=C:\IBMCOM\LANMSGDD.OS2 /I:C:\IBMCOM
DEVICE=C:\IBMCOM\PROTMAN.OS2 /I:C:\IBMCOM
DEVICE=C:\OS2\INSTALL\IBMCSFLK.SYS C:\OS2\INSTALL\IBMCSFLK.LST
DEVICE=C:\OS2\TESTCFG.SYS
DEVICE=C:\OS2\DOS.SYS
```

```
DEVICE=C:\OS2\PMDD.SYS
BUFFERS=90
IOPL=YES
DISKCACHE=2048,LW,128,AC:CE
MAXWAIT=5
MEMMAN=SWAP,PROTECT
SWAPPATH=C: 4096 20480
BREAK=OFF
THREADS=2048
PRINTMONBUFSIZE=134,134,134
COUNTRY=001,C:\OS2\SYSTEM\COUNTRY.SYS
SET KEYS=ON
REM SET DELDIR=C:\DELETE,512;D:\DELETE,512;
 E:\DELETE,512;F:\DELETE,512;
BASEDEV=PRINT02.SYS
BASEDEV=IBM2FLPY.ADD
BASEDEV=IBM2SCSI.ADD /LED
BASEDEV=OS2DASD.DMD
BASEDEV=OS2SCSI.DMD
SET BOOKSHELF=D:\SQLLIB\BOOK;E:\FTW\MANUCHAR\ENU;
 C:\IBMLAN\BOOK;C:\OS2\BOOK;C:\TOOLKT21\BOOK;
 C:\IBMCPP\HELP;X:\HELP;C:\CMLIB\BOOK;
SET EPMPATH=C:\OS2\APPS;
REM DEVICE=C:\OS2\APPS\SASYNCDB.SYS
PROTECTONLY=NO
SHELL=C:\OS2\MDOS\COMMAND.COM C:\OS2\MDOS
FCBS=16,8
RMSIZE=640
DEVICE=C:\OS2\MDOS\VEMM.SYS
DOS=LOW,NOUMB
DEVICE=C:\OS2\MDOS\VDPX.SYS
DEVICE=C:\OS2\MDOS\VXMS.SYS /UMB
DEVICE=C:\OS2\MDOS\VDPMI.SYS
DEVICE=C:\OS2\MDOS\VCDROM.SYS
DEVICE=C:\OS2\MDOS\VWIN.SYS
REM DEVICE=C:\OS2\PCMCIA.SYS
REM DEVICE=C:\OS2\MDOS\VPCMCIA.SYS
DEVICE=C:\OS2\MDOS\VMOUSE.SYS
DEVICE=C:\OS2\POINTDD.SYS
DEVICE=C:\OS2\MOUSE.SYS
DEVICE=C:\OS2\COM.SYS
DEVICE=C:\OS2\MDOS\VCOM.SYS
rem DEVICE=D:\FRNV1R0\DLL\EXY24.SYS
CODEPAGE=437,850
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
BASEDEV=XGA.SYS
DEVICE=C:\OS2\XGARINGO.SYS
DEVINFO=SCR,VGA,C:\OS2\VIOTBL.DCP
SET VIDEO_DEVICES=VIO_XGA
SET VIO_XGA=DEVICE(BVHVGA,BVHXGA)
DEVICE=C:\OS2\MDOS\VVGA.SYS
DEVICE=C:\OS2\MDOS\VXGA.SYS
RUN=C:\OS2\EPW.EXE
RUN=C:\OS2\EPWROUT.EXE 1
RUN=C:\OS2\SYSTEM\LOGDAEM.EXE
DEVICE=C:\OS2\LOG.SYS
SET INCLUDE=D:\SQLLIB;D:\FRNV1R0\INCLUDE;D:\FRNV1R0\INC;
C:\IBMCPP\INCLUDE;C:\IBMCPP\IBMCLASS;X:\INCLUDE;X:\IBMCLASS;
SET LIB=D:\SQLLIB;D:\FRNV1R0\DB_LIB;D:\FRNV1R0\LIB;C:\IBMCPP\LIB;X:\LIB;
```

```
DEVICE=C:\OS2\EPWDD.SYS
RUN=C:\OS2\EPWDDR3.EXE
rem SET TMP=C:\
SET PROGREF21=CPGREF1.INF+CPGREF2.INF+CPGREF3.INF
SET PMREF=PMFUN.INF+PMGPI.INF+PMHOK.INF+PMMSG.INF+PMREL.INF+PMWIN.INF+PMWKP.INF
SET HELPNDX=EPMKWHLP.NDX+DDE4LRM.NDX+DDE4SCL.NDX+DDE4UIL.NDX+DDE4CCL.NDX+DDE4CLIB.NDX
SET TMP=C:\IBMCPP\TMP
rem DEVICE=C:\IBMCPP\SYS\DDE4XTRA.SYS
REM *VisualInfo - Stand-alone System* SET FRNROOT=D:\FRNV1R0
REM *VisualInfo - Stand-alone System* SET FRNDISP=YES
REM *VisualInfo - Stand-alone System* SET FRNDEFLANG=ENU;
REM *VisualInfo - Stand-alone System* SET FRNIP2=FRNOFIOS;FRNOFO2;FRNOSMSA
REM *VisualInfo - Stand-alone System* SET SB_ROOT=D:\FRNV1R0
REM *VisualInfo - Stand-alone System* SET DEBUG=0
REM *VisualInfo - Stand-alone System* SET DIAGNOSE=0
REM *VisualInfo - Stand-alone System* SET LBLS_DB=LIBSRVR2
REM *VisualInfo - Stand-alone System* SET FRNCACHEPAGE=3

SET TMP=C:\IBMCPP
REM *VisualInfo - Standard Client* SET FRNROOT=D:\FRNV1R0
REM *VisualInfo - Standard Client* SET FRNDISP=YES
REM *VisualInfo - Standard Client* SET FRNDEFLANG=ENU;
REM *VisualInfo - Standard Client* SET FRNIP2=FRNOFIOS;FRNOFO2;FRNOSMSA
REM *VisualInfo - Standard Client* SET SB_ROOT=D:\FRNV1R0
REM *VisualInfo - Standard Client* SET DEBUG=0
REM *VisualInfo - Standard Client* SET DIAGNOSE=0
REM *VisualInfo - Standard Client* SET LBLS_DB=LIBSRVR2
REM *VisualInfo - Standard Client* SET FRNCACHEPAGE=3
DEVICE=C:\CMLIB\DFTDD.SYS
DEVICE=C:\CMLIB\ACSLANDD.SYS
DEVICE=C:\CMLIB\CMKFMDE.SYS
SET CMPATH=C:\CMLIB
REM *VisualInfo - Selectable Components* SET FRNROOT=d:\FRNV1R0
REM *VisualInfo - Selectable Components* SET FRNDISP=YES
REM *VisualInfo - Selectable Components* SET FRNDEFLANG=ENU;
REM *VisualInfo - Selectable Components* SET FRNIP2=FRNOFIOS;FRNOFO2;FRNOSMSA
REM *VisualInfo - Selectable Components* SET SB_ROOT=d:\FRNV1R0
REM *VisualInfo - Selectable Components* SET DEBUG=0
REM *VisualInfo - Selectable Components* SET DIAGNOSE=0
REM *VisualInfo - Selectable Components* SET LBLS_DB=LIBSRVR2
REM *VisualInfo - Selectable Components* SET FRNCACHEPAGE=3
SET FRNROOT=D:\FRNV1R0
SET FRNDISP=YES
SET FRNDEFLANG=ENU;
SET FRNIP2=FRNOFIOS;FRNOFO2;FRNOSMSA
SET SB_ROOT=D:\FRNV1R0
SET DEBUG=0
SET DIAGNOSE=0
SET LBLS_DB=LIBSRVR2
SET FRNCACHEPAGE=3
PRIORITY=DYNAMIC
DEVICE=C:\CMLIB\T1P1NDD.SYS CFG=C:\CMLIB\5250.CFG
SET EHNL=2924
SET EHNP=D:\PCSOS2
IFS=D:\PCSOS2\EHNSFL0.DLL
IFS=D:\PCSOS2\EHNVPFS.IFS MON=1,2,3
RUN=D:\PCSOS2\UPDATEP2.EXE
DEVICE=D:\PCSOS2\EHNVPRDR.SYS
DEVICE=D:\PCSOS2\EHNPCPDD.SYS
```

```
DEVICE=D:\PCSOS2\EHNPCVDD.SYS
REM *Query_for_OS/2_(Standalone)* SET FTB1DIR=E:\FTW\WORK
REM *Query_for_OS/2_(Standalone)* SET FTB1PATH=E:\FTW
REM *Query_for_OS/2_(Standalone)* SET FTBBASE=E:\FTW
SET FTB1PRELOAD=30
SET DSSPATH=E:\FTW
RUN=C:\OS2\INSTALL\IBMCSFLK.EXE C:\OS2\INSTALL\IBMCSFLK.LST
rem     DEVICE=C:\ODDCS\SRVIFS.SYS
rem IFS=C:\ODDCS\SRVIFSC.IFS *
RUN=C:\IBMCOM\PROTOCOL\NETBIND.EXE
RUN=C:\IBMCOM\LANMSGEX.EXE
DEVICE=C:\IBMCOM\PROTOCOL\NETBEUI.OS2
DEVICE=C:\IBMLAN\NETPROG\RDRHELP.200
IFS=C:\IBMLAN\NETPROG\NETWKSTA.200 /I:C:\IBMLAN /N
DEVICE=C:\IBMCOM\PROTOCOL\NETBIOS.OS2
DEVICE=C:\IBMCOM\PROTOCOL\LANDD.OS2
DEVICE=C:\IBMCOM\PROTOCOL\LANDLLDD.OS2
DEVICE=C:\IBMCOM\MACS\IBMTOK.OS2
RUN=C:\IBMCOM\PROTOCOL\LANDLL.EXE
RUN=C:\IBMLAN\NETPROG\LSDAEMON.EXE
DEVICE=D:\CID\IMG\SRVIFS\SRVIFS.SYS
IFS=D:\CID\IMG\SRVIFS\SRVIFSC.IFS *
SET QRWDR=D:
SET QRWINST=D:\SQLLIB
```

E.2 Network Table

```
/* Network Table Generation Utility                                    */
/* DAA1006/USIBMSC.DAA1006                                             */
/* Thu Mar  9 15:46:53 1995                                            */


SERVER: HBOS0001 REMOTE APPN
            TP            = FRNO
            MODE          = LU62APPB
            LU_NAME       = USIBMSC.IMOBRSV
            SECURITY      = NONE
          SESSION_TIMEOUT = 3000
            SERVER_TYPE   = FRNOSMVS


SERVER: VIUSR1 REMOTE APPN
            TP            = FRNI
            MODE          = LU62APPB
            LU_NAME       = USIBMSC.IMLBRSV
            SECURITY      = NONE
          SESSION_TIMEOUT = 3000
            SERVER_TYPE   = FRNLSMVS


SERVER: FRNOCSVR LOCAL LAN_WIDE APPN
            TP            = FRNI
            PATH          = \FRNOLMLM.EXE
            START         = WINDOW
            ICON          = \FRNOLMLM.ICO
            PASSNAME
            SERVER_TYPE   = FRNCS
```

```
SERVER: FRNCACHE LOCAL LOCAL_ONLY
            TP            = FRNI
            PATH          = \FRNOLMLM.EXE
            START         = WINDOW
            ICON          = \FRNOLMLM.ICO
            PASSNAME
            SERVER_TYPE   = FRNCACHE


ALIAS: FRNONSVR FRNOCSVR


SERVER: FRNODAOS LOCAL LAN_WIDE APPN
            TP            = FRNI
```

E.3 OS/2 Communication Manager/2 Configuration File

```
  DEFINE_LOCAL_CP  FQ_CP_NAME(USIBMSC.DAA1006  )
                   CP_ALIAS(DAA1006 )
                   NAU_ADDRESS(INDEPENDENT_LU)
                   NODE_TYPE(EN)
                   NODE_ID(X'05DA1006')
                   NW_FP_SUPPORT(NONE)
                   HOST_FP_SUPPORT(YES)
                   HOST_FP_LINK_NAME(HOST0001)
                   MAX_COMP_LEVEL(NONE)
                   MAX_COMP_TOKENS(0);

  DEFINE_LOGICAL_LINK  LINK_NAME(HOST0001)
                       ADJACENT_NODE_TYPE(LEARN)
                       DLC_NAME(IBMTRNET)
                       ADAPTER_NUMBER(0)
                       DESTINATION_ADDRESS(X'400008210210')
                       ETHERNET_FORMAT(NO)
                       CP_CP_SESSION_SUPPORT(YES)
                       SOLICIT_SSCP_SESSION(YES)
                       NODE_ID(X'05DA1006')
                       ACTIVATE_AT_STARTUP(YES)
                       USE_PUNAME_AS_CPNAME(NO)
                       LIMITED_RESOURCE(USE_ADAPTER_DEFINITION)
                       LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                       MAX_ACTIVATION_ATTEMPTS(USE_ADAPTER_DEFINITION)
                       EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                       COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                       COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                       SECURITY(USE_ADAPTER_DEFINITION)
                       PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                       USER_DEFINED_1(USE_ADAPTER_DEFINITION)
                       USER_DEFINED_2(USE_ADAPTER_DEFINITION)
                       USER_DEFINED_3(USE_ADAPTER_DEFINITION);

  DEFINE_LOCAL_LU  LU_NAME(DAA1006I)
                   LU_ALIAS(DAA1006I)
                   NAU_ADDRESS(INDEPENDENT_LU);

  DEFINE_LOCAL_LU  LU_NAME(DAA1006J)
                   LU_ALIAS(DAA1006J)
```

```
                NAU_ADDRESS(INDEPENDENT_LU);

DEFINE_PARTNER_LU  FQ_PARTNER_LU_NAME(USIBMSC.IMLBRSV  )
                   PARTNER_LU_ALIAS(IMLBRSV)
                   PARTNER_LU_UNINTERPRETED_NAME(IMLBRSV )
                   MAX_MC_LL_SEND_SIZE(32767)
                   CONV_SECURITY_VERIFICATION(NO)
                   PARALLEL_SESSION_SUPPORT(YES);

DEFINE_PARTNER_LU  FQ_PARTNER_LU_NAME(USIBMSC.IMOBRSV  )
                   PARTNER_LU_ALIAS(IMOBRSV)
                   PARTNER_LU_UNINTERPRETED_NAME(IMOBRSV )
                   MAX_MC_LL_SEND_SIZE(32767)
                   CONV_SECURITY_VERIFICATION(NO)
                   PARALLEL_SESSION_SUPPORT(YES);

DEFINE_MODE  MODE_NAME(FRNOMVS )
             COS_NAME(#CONNECT)
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(1024)
             RECEIVE_PACING_WINDOW(8)
             MAX_NEGOTIABLE_SESSION_LIMIT(32767)
             PLU_MODE_SESSION_LIMIT(0)
             MIN_CONWINNERS_SOURCE(0)
             COMPRESSION_NEED(PROHIBITED)
             PLU_SLU_COMPRESSION(NONE)
             SLU_PLU_COMPRESSION(NONE);

DEFINE_MODE  MODE_NAME(FRNOSRCL)
             COS_NAME(#CONNECT)
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(4096)
             RECEIVE_PACING_WINDOW(8)
             MAX_NEGOTIABLE_SESSION_LIMIT(32767)
             PLU_MODE_SESSION_LIMIT(5)
             MIN_CONWINNERS_SOURCE(1)
             COMPRESSION_NEED(PROHIBITED)
             PLU_SLU_COMPRESSION(NONE)
             SLU_PLU_COMPRESSION(NONE);

DEFINE_MODE  MODE_NAME(FRNOLMCL)
             COS_NAME(#CONNECT)
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(4096)
             RECEIVE_PACING_WINDOW(8)
             MAX_NEGOTIABLE_SESSION_LIMIT(32767)
             PLU_MODE_SESSION_LIMIT(8)
             MIN_CONWINNERS_SOURCE(1)
             COMPRESSION_NEED(PROHIBITED)
             PLU_SLU_COMPRESSION(NONE)
             SLU_PLU_COMPRESSION(NONE);

DEFINE_MODE  MODE_NAME(FRNOSRSR)
             COS_NAME(#CONNECT)
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(4096)
             RECEIVE_PACING_WINDOW(8)
             MAX_NEGOTIABLE_SESSION_LIMIT(32767)
             PLU_MODE_SESSION_LIMIT(15)
```

```
                MIN_CONWINNERS_SOURCE(1)
                COMPRESSION_NEED(PROHIBITED)
                PLU_SLU_COMPRESSION(NONE)
                SLU_PLU_COMPRESSION(NONE);

DEFINE_MODE  MODE_NAME(#INTERCS)
                COS_NAME(#INTERSC)
                DEFAULT_RU_SIZE(YES)
                RECEIVE_PACING_WINDOW(7)
                MAX_NEGOTIABLE_SESSION_LIMIT(8)
                PLU_MODE_SESSION_LIMIT(8)
                MIN_CONWINNERS_SOURCE(4)
                COMPRESSION_NEED(REQUESTED)
                PLU_SLU_COMPRESSION(LZ9)
                SLU_PLU_COMPRESSION(LZ9);

DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                    DEFAULT_MODE_NAME(BLANK)
                    DEFAULT_LOCAL_LU_ALIAS(DAA1006I)
                    MAX_MC_LL_SEND_SIZE(32767)
                    DIRECTORY_FOR_INBOUND_ATTACHES(*)
                    DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                    DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                    DEFAULT_TP_CONV_SECURITY_RQD(NO)
                    MAX_HELD_ALERTS(10);

DEFINE_TP  TP_NAME(FRNI)
           PIP_ALLOWED(NO)
           FILESPEC(D:\FRNV1RO\FRNOLIIS.EXE)
           ICON_FILESPEC(D:\FRNV1RO\FRNOLIIS.ICO)
           CONVERSATION_TYPE(EITHER)
           CONV_SECURITY_RQD(NO)
           SYNC_LEVEL(EITHER)
           TP_OPERATION(QUEUED_AM_STARTED)
           PROGRAM_TYPE(VIO_WINDOWABLE)
           INCOMING_ALLOCATE_QUEUE_DEPTH(255)
           INCOMING_ALLOCATE_TIMEOUT(INFINITE)
           RECEIVE_ALLOCATE_TIMEOUT(INFINITE);

START_ATTACH_MANAGER;
```

Appendix F. General Installation Jobs and Tips

Below is a list of things that you can reference when you are installing the VisualInfo Host Based Library Server and Object Server.

1. Review the PSP bucket. PTF UN75318 must be installed prior to the customization of VisualInfo installation jobs.
2. Sample JCL to copy installation jobs to new data sets for customization. You can use IPOUPDTE to mass changes all the jobs.

   SMP/E Related Jobs

```
//*  STEPS:                                                          00010000
//*   DELETE  DELETE ALLOCATED DATA SET                              00010000
//*   ALLOC   ALLOCATE SMPCNTL DATA SET                              00010000
//*   LSSMPE  COPY HBLS SMP/E JOBS                                   00010000
//*   OSSMPE  COPY HBOS SMP/E JOBS                                   00010000
//*
//DELETE   EXEC PGM=IEFBR14
//SMPCNTL  DD   DSN=VIUSR1.NEW.SMPCNTL,DISP=(SHR,DELETE)
//*
//ALLOC    EXEC PGM=IEFBR14
//SMPCNTL  DD   DSN=VIUSR1.NEW.SMPCNTL,
//           DISP=(NEW,CATLG,DELETE),
//           DCB=(RECFM=FB,BLKSIZE=23440,LRECL=80),
//           SPACE=(CYL,(1,1,30)),VOL=SER=IMAGE1
//*
//LSSMPE   EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//PRODTAPE DD   DISP=SHR,DSN=SYSADM1.PRODTAPE.LIBSVR
//SMPCNTL  DD   DISP=SHR,DSN=VIUSR1.NEW.SMPCNTL
//SYSUT3   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSUT4   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSIN    DD   *
LIBSVR     COPY I=PRODTAPE,O=SMPCNTL
           SELECT MEMBER=((FRNACC,FRNLACC),(FRNACCCK,FRNLACCC))
           SELECT MEMBER=((FRNALLOC,FRNLALLO),(FRNAPP,FRNLAPP))
           SELECT MEMBER=((FRNAPPCK,FRNLAPPC),(FRNDDDEF,FRNLDDEF))
           SELECT MEMBER=((FRNPSTLK,FRNLPSTL),(FRNREC,FRNLREC))
/*
//OSSMPE   EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//PRODTAPE DD   DISP=SHR,DSN=SYSADM1.PRODTAPE.OBJSVR
//SMPCNTL  DD   DISP=SHR,DSN=VIUSR1.NEW.SMPCNTL
//SYSUT3   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSUT4   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSIN    DD   *
OBJSVR     COPY I=PRODTAPE,O=SMPCNTL
           SELECT MEMBER=((FRNACC,FRNOACC),(FRNACCCK,FRNOACCC))
           SELECT MEMBER=((FRNAPP,FRNOAPP),(FRNAPPCK,FRNOAPPC))
           SELECT MEMBER=(FRNOALLO,FRNODDEF,FRNOPSTL)
           SELECT MEMBER=((FRNREC,FRNOREC))
/*
```

   VisualInfo Installation Jobs

```
//*  STEPS:                                                          00010000
//*   DELETE  DELETE ALLOCATED DATA SETS                             00010000
//*   ALLOC   ALLOCATE DATA SETS                                     00010000
//*   LSDB2   COPY HBLS DB2 RELATED JOBS                             00010000
//*   LSCICS  COPY HBLS CICS RELATED JOBS                            00010000
//*   OSDB2   COPY HBOS DB2 RELATED JOBS                             00010000
//*   OSCICS  COPY HBOS CICS RELATED JOBS                            00010000
//*
//DELETE   EXEC PGM=IEFBR14
//INSTALL  DD   DSN=VIUSR1.NEW.INSTALL,DISP=(SHR,DELETE)
//TABLES   DD   DSN=VIUSR1.NEW.TABLES,DISP=(SHR,DELETE)
//LIBSVRIN DD   DSN=VIUSR1.NEW.LIBSVRIN,DISP=(SHR,DELETE)
//LIBSVRVS DD   DSN=VIUSR1.NEW.FRNVSAM1,DISP=(SHR,DELETE)
//*
//ALLOC    EXEC PGM=IEFBR14
//INSTALL  DD   DSN=VIUSR1.NEW.INSTALL,
//           DISP=(NEW,CATLG,DELETE),
//           DCB=(RECFM=FB,BLKSIZE=23440,LRECL=80),
//           SPACE=(CYL,(1,1,30)),VOL=SER=IMAGE1
//TABLES   DD   DSN=VIUSR1.NEW.TABLES,
//           DISP=(NEW,CATLG,DELETE),
//           DCB=(RECFM=FB,BLKSIZE=23440,LRECL=80),
//           SPACE=(CYL,(1,1,30)),VOL=SER=IMAGE1
//LIBSVRIN DD   DSN=VIUSR1.NEW.LIBSVRIN,
//           DISP=(NEW,CATLG,DELETE),
//           DCB=(RECFM=FB,BLKSIZE=23440,LRECL=80),
//           SPACE=(TRK,(5,5,5)),VOL=SER=IMAGE1
//LIBSVRVS DD   DSN=VIUSR1.NEW.FRNVSAM1,
//           DISP=(NEW,CATLG,DELETE),
//           DCB=(RECFM=VB,BLKSIZE=2554,LRECL=255),
//           SPACE=(TRK,(5,5,5)),VOL=SER=IMAGE1
//*
//LSDB2    EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//SFRNINS1 DD   DISP=SHR,DSN=VILIBSVR.SFRNINS1
//SFRNSMP1 DD   DISP=SHR,DSN=VILIBSVR.SFRNSMP1
//SFRNVSA1 DD   DISP=SHR,DSN=VILIBSVR.SFRNVSA1
//SFRNTBL1 DD   DISP=SHR,DSN=VILIBSVR.SFRNTBL1
//INSTALL  DD   DISP=SHR,DSN=VIUSR1.NEW.INSTALL
//TABLES   DD   DISP=SHR,DSN=VIUSR1.NEW.TABLES
//LIBSVRIN DD   DISP=SHR,DSN=VIUSR1.NEW.LIBSVRIN
//LIBSVRVS DD   DISP=SHR,DSN=VIUSR1.NEW.FRNVSAM1
//SYSUT3   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSUT4   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSIN    DD   *
INST1      COPY I=SFRNINS1,O=INSTALL
           SELECT MEMBER=(FRNDBCRT,FRNDBCR2,FRNDBLOD,FRNDBLD2)
           SELECT MEMBER=(FRNDBSYN,FRNPKBD1,FRNPKBD2)
           SELECT MEMBER=(FRNPLBND)
           SELECT MEMBER=(FRNDBALS,FRNDBAL2,FRNDBGRT,FRNDBGR2)
           SELECT MEMBER=(FRNCMSGC)
           SELECT MEMBER=(FRNICOVC)
INST2      COPY I=SFRNSMP1,O=INSTALL
           SELECT MEMBER=(FRNPLBCS,FRNRUNCS)
LSRCT      COPY I=SFRNTBL1,O=TABLES
           SELECT MEMBER=(FRNRCTU1)
ENINDX1    COPY I=SFRNINS1,O=INSTALL
           SELECT MEMBER=(FRNDBCR3,FRNDBLD3)
ENINDX2    COPY I=SFRNSMP1,O=INSTALL
```

```
           SELECT MEMBER=(FRNBDAVT)
ENINDX3A   COPY I=SFRNSMP1,O=LIBSVRIN
           SELECT MEMBER=(FRNICGEN)
ENINDX31   COPY I=SFRNSMP1,O=INSTALL
           SELECT MEMBER=(FRNICGEP)
ENINDX3B   COPY I=SFRNSMP1,O=LIBSVRIN
           SELECT MEMBER=(FRNICGN2)
ENINDX32   COPY I=SFRNSMP1,O=INSTALL
           SELECT MEMBER=(FRNICGP2)
ENINDX3C   COPY I=SFRNSMP1,O=LIBSVRIN
           SELECT MEMBER=(FRNICGN3)
ENINDX33   COPY I=SFRNSMP1,O=INSTALL
           SELECT MEMBER=(FRNICGP3)
ENINDX4    COPY I=SFRNVSA1,O=LIBSVRVS
           SELECT MEMBER=(FRNICOVR)
OPIXJOBA   COPY I=SFRNSMP1,O=LIBSVRIN
           SELECT MEMBER=(FRNSQ1GN,FRNSQ1PJ)
OPIXPROA   COPY I=SFRNSMP1,O=INSTALL
           SELECT MEMBER=(FRNSQ1CP,FRNSQ1PP)
OPIXJOBB   COPY I=SFRNSMP1,O=LIBSVRIN
           SELECT MEMBER=(FRNSQ2GN,FRNSQ2BJ)
OPIXPROB   COPY I=SFRNSMP1,O=INSTALL
           SELECT MEMBER=(FRNSQ2PP,FRNSQ2BP)
OPIXJOBC   COPY I=SFRNSMP1,O=LIBSVRIN
           SELECT MEMBER=(FRNSQ3PJ)
OPIXPROC   COPY I=SFRNSMP1,O=INSTALL
           SELECT MEMBER=(FRNSQ3PP)
/*
//LSCICS   EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//SFRNTBL1 DD   DISP=SHR,DSN=VILIBSVR.SFRNTBL1
//SFRNINS1 DD   DISP=SHR,DSN=VILIBSVR.SFRNINS1
//SFRNSMP1 DD   DISP=SHR,DSN=VILIBSVR.SFRNSMP1
//TABLES   DD   DISP=SHR,DSN=VIUSR1.NEW.TABLES
//INSTALL  DD   DISP=SHR,DSN=VIUSR1.NEW.INSTALL
//SYSUT3   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSUT4   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSIN    DD   *
TABLES     COPY I=SFRNTBL1,O=TABLES
           SELECT MEMBER=(FRNPLTU1,FRNPLTU2)
           SELECT MEMBER=(FRNDCTU1)
           SELECT MEMBER=(FRNCCSD1,FRNCCSD2)
DEFGROUP   COPY I=SFRNINS1,O=TABLES
           SELECT MEMBER=(FRNCSDUP)
SESSCONN   COPY I=SFRNSMP1,O=TABLES
           SELECT MEMBER=(FRNCNCLI)
/*
//OSDB2    EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
//SFRNSMP2 DD   DISP=SHR,DSN=VIOBJSVR.SFRNSMP2
//TABLES   DD   DISP=SHR,DSN=VIUSR1.NEW.TABLES
//SYSUT3   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSUT4   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSIN    DD   *
OSRCT      COPY I=SFRNSMP2,O=TABLES
           SELECT MEMBER=((FRNRCTU1,FRNRCTU0))
/*
//OSCICS   EXEC PGM=IEBCOPY
//SYSPRINT DD   SYSOUT=*
```

```
//SFRNSMP2 DD   DISP=SHR,DSN=VIOBJSVR.SFRNSMP2
//TABLES   DD   DISP=SHR,DSN=VIUSR1.NEW.TABLES
//INSTALL  DD   DISP=SHR,DSN=VIUSR1.NEW.INSTALL
//SYSUT3   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSUT4   DD   UNIT=SYSDA,SPACE=(TRK,(15))
//SYSIN    DD   *
DEFGROUP   COPY I=SFRNSMP2,O=INSTALL
           SELECT MEMBER=(FRNCSDUJ)
SESSCONN   COPY I=SFRNSMP2,O=TABLES
           SELECT MEMBER=((FRNCNCLI,FRNCNCLO))
           SELECT MEMBER=((FRNCCSD1,FRNOCSD1))
/*
```

3. Before starting library server tables, FRNDBCRT, you should read the Performance Considerations section in *Module C7: Installing the IBM ImagePlus VisualInfo Library Server for MVS/ESA*. You might want to change the PCTFREE value before submitting the FRNDBCRT job.

4. If C/370 is not able to be initialized during the CICS initialization, you must ensure that the EDCCICS module in the library listed in LINKLST or DFHRPL is the same level as the C/370 library used during library server postlink job. For example, if you use SYSP.C370.SEDCBASE for the Postlink, you must use EDCCICS module from SYSP.C370.SEDCLINK.

   The same rule should be applied to other runtime libraries.

5. The VisualInfo installation modules assume that CICS installation tasks are performed on working CICS regions. Therefore, you have to make sure your CICS regions are functional (i.e. DB2 and OAM are enabled) prior to modification.

6. If VisualInfo library server PLT initialization programs fail, you can can bring up the CICS region first and test the PLT initialization programs later. You can turn on the CICS auxtrace and test these initialization programs.

   To test the first initialization program, FRNMHBP1, you have define it as an transaction. For example, DEFINE TRAN(HBP1) GROUP(TEMP1) PROGRAM(FRNMHBP1) TASKDATALOC(BELOW) DE(test FRNMHBP1 program). You then enter HBP1 for the FRNMHBP1 program, FRN2 for the FRNMHBP2 program, and FRNT for the FRNMHBP3 program.

   If the FRNMHBP1 fails, the problem is related to the environment, which means that PL/1 or C/370 were not install correctly or the wrong level of runtime libraries was used.

7. If you are using a collection name that was never used before (see FRNDBLOD job) and you are using DFSMS/MVS, you can use the following to test your new collection name:

```
//OCDBTABS EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
  OSREQ STORE MY.COLLECT.NAME A1996JUN.JAMYNAME LENGTH(1048576)
  OSREQ QUERY MY.COLLECT.NAME A1996JUN.JAMYNAME
  LISTCAT ENTRIES('MY.COLLECT.NAME')
  OSREQ CHANGE MY.COLLECT.NAME A1996JUN.JAMYNAME RP(365)
  OSREQ QUERY MY.COLLECT.NAME A1996JUN.JAMYNAME
  OSREQ RETRIEVE MY.COLLECT.NAME A1996JUN.JAMYNAME COMPARE VIEW(PRIMARY)
  OSREQ DELETE MY.COLLECT.NAME A1996JUN.JAMYNAME
/*
```

   This sample is taken from the Appendix B, *DFSMS/MVS Object Access Method Planning, Installation, and Storage Administration Guide for Object Support*, SC26-4918.

   If any of OSREQ statement fails, you need to check your ACS routines, OAM storage groups, and ISMF definitions.

8. For the object Server, you can use any data base name on the configuration utility. OAM and ACS routines will take care it for you.

9. If you plan to modify your VisualInfo network table, FRNOLINT.TBL, on the workstation, you have to make sure the **=** column is aligned. You should not create a network table from scratch.
10. If your VTAM is not version 4, you can only use APPC instead of APPN.
11. If you have to choose between independent LU or dependent LU, it is recommended to use dependent LU.
12. If your client is successfully logged on to the library server, you should be able find the eye catcher, GREEN EGGS AND HAM, in your FFST trace details report. Otherwise, you have to look for the **sense data** in the FFST trace details report to determine the problem.
13. The easiest way to check your installation is to logon to the client as FRNADMIN and import a PC file. However, if you firmly believe that you have not make any mistakes and you are not able to logon or import, you can use VIBB package from the IMAGE tools disk and use chapter 3, First Steps in Using VIBB, *VisualInfo Building Blocks for REXX and OS/2 Command Line* to do another test. If it fails also, the problem is in one of your installation steps.
14. Although it is optional to create the index class, CLAIM, it is recommended to create it. Once it is created, you can use it for your testing before testing your creating index class JCL.
15. To test your creating index class JCL, you can issue FRII from the library server CICS region.
16. If you get a return code in an error message from the library server and you need to find the explanation, the APAR PN64441 can help you.
17. If you need to find return codes and reason codes for OAM, information APARs II05257 and II07284 contain this information.

Glossary

Access Control. Access control provides the capability to limit access to certain functions provided by the system and certain documents stored in the system.

Access Level. The level of authority a subject has when using a protected source, for example, authority to access a particular security level of information.

Access List. A collection of access rights for various users or groups to one object.

Advanced Search Settings Notebook. The advanced search settings notebook lets you create and modify existing advance search profiles. You see an advanced search setting notebook by selecting the new search entry in the fileroom search container or by selecting Open -> Settings on an existing search template. When you select an existing advanced search profile from the fileroom search container, a modified advanced search notebook appears that lets you enter new data values to use as search criteria for the advanced search.

Attribute. An attribute is a characteristic that can be a user-defined attribute associated with index classes or a system attribute that is created when you create an item. End-users see attributes as key fields.

Basic Search Dialog. The basic search dialog window lets you enter simple search criteria associated with a single index class or all classes. You see a basic search dialog by selecting basic search from the tools menu in the main container or by selecting the Fileroom menu in the fileroom search container.

Cache. The cache is a storage area used by the Client to manage request and response blocks and local copies of item parts. Local copies of item parts are stored in the cache catalog. You access the cache through the List Manager APIs.

Catalog. When you create an element, it does not belong in any list. Some elements might not logically belong in a list. A catalog organizes and stores these orphan elements. The catalog is an index your application uses to insert and remove keyed elements. Your application does not create, delete, open, or close the catalog. It uses a catalog handle to gain access to the catalog and the elements that it contains.

Client. (1) A user. (2) An OS/2 node when you install VisualInfo Client for OS/2. Clients let users, system administrators, application programmers, and others who has user ID′s, access and work with the VisualInfo system.

Client Application. An application written with the VisualInfo APIs to customize the user interface.

Client-Server. The model of interaction in distributed processing in which a program at one site sends a request to a program at another site and awaits a response. The requesting program is called a client, the answering program a server.

Collection. A collection is a group of objects with a similar set of management rules. Every object is in one collection, and all objects in a collection are in the same storage group.

Common Interchange Format. The common interchange format (cif) is a standard folder and document format used to copy folders and documents from one ImagePlus system to another. You can export folders and documents as cif files, and you can import cif files. You select the cif as an option on the export dialog and the import dialog.

Configuration Server. A component of the VisualInfo system that acts as an interface between clients and other components of the VisualInfo system. Each configuration server stores profile information about each client that links to it.

Content Class. A content class describes the physical format of an object stored in the Object Server.

Database. (1) A collection of data with a given structure for accepting, storing and providing, on demand, data for multiple users. (2) A collection of interrelated data organized to serve one or more applications.

Database Table. A named collection of data consisting of rows and columns.

Data model. The data model provides a logical view of the organization of data in a database. In the VisualInfo system, the data model provides your applications with many general document and folder management capabilities.

Document. A document in the VisualInfo system is a basic part of the Folder Manager data model and is similar to a paper document.

Element. An element is a data resource that the List Manager allocates for your application. An element is a single object that you want your application to store, retrieve, or manipulate. Elements can be image objects, spreadsheets, or any other objects in the VisualInfo system that identify tasks for your application program or user.

Export Dialog. The export dialog lets you specify user options for exporting documents and folders from the VisualInfo system to files. You see the export dialog by selecting export from the document, folder, and selected menus.

Fileroom. The fileroom is the permanent location for documents and folders in the VisualInfo system.

Folder. A folder in the VisualInfo system is a basic part of the Folder Manager that is similar to folders in a paper filing system and can contain other folders or documents.

Folder Management. (1) A part of the VisualInfo system that organizes document images into electronic file folders for convenient access and retrieval. (2) The use of a computer program to store, retrieve and manage information associated with an index number or other form of identification.

Group Name. (1) A generic name for a collection of input or output devices. (2) A single name identifying a group of users with the same attributes.

History. A history log holds a record of activities for a folder or document in workflow.

History Log Window. The history log window is contained in the library object window (low) for folders and documents when you request history using the process menu. The history log window contains history events for the item associated with the VisualInfo Client's work-in-process (workbasket and workflow) activities. History events implemented by your organization can also be displayed.

Image Object. An object that contains image data. Also see *object*.

Image Window. The image window is contained in the low. It can display a document or an icon representing an object of an externally supported content class. The VisualInfo Client uses Image Service functions to display the view window.

Import Dialog. The import dialog lets you select files to include in the VisualInfo library. You see an import dialog by selecting the import option from the tools menu in the main container.

Index. (1) A list of the contents of a file or a document, together with keys or references for locating the contents. (2) In &vi,. specific information about a document used for folder management and routing information used for workflow management.

Index Class. The index class definition specifies attributes common to a set of items. For example, an index class for reports could have attributes such as author, data, or subject. When you create an item, your application must assign an index class and supply the attribute values required by that class. You can create separate index classes for reports, spreadsheets, claim forms, or other kinds of documents that you use.

Index Class View. An index class view is a subset of all the attributes associated with an index class. You can use index class views for access control, update control, and searches. For example, a personnel index class can include attributes for sensitive information. An index class view for a user set might include attributes such as address and telephone number but not attributes for salary or performance ratings.

When the system creates an index class, it defines a default index view that contains all attributes defined for the index class. The system administrator can create as many other index class views that are necessary to meet requirements for all users. Index class views can restrict attributes as read-only or read-write.

Item. An item is an independent entity created by the Client interface. The Client lets your application index items and specify features such as security on an item-by-item basis.

Item Part. A part is associated with an item that is typically stored on an Object Server and has a content class describing the physical format.

Library Object Window. The library object window (low) of the VisualInfo Client contains information windows associated with the workbasket, folder, or document you are displaying.

The information windows that can appear are:

- Image window
- Table of content (toc) window
- Index form window
- Note log window
- History log window

You see a low by selecting a workbasket from the workbasket container, a document or folder from a workbasket or folder, or as the result of a search.

List. A list is a data structure the List Manager allocates in its file space for your application. Elements in a list reside in contiguous positions. Your application inserts or removes elements from lists. Your application must explicitly create, open, delete, and close lists. A list name identifies a list. If your application does not explicitly provide a list name when it creates the list, the List Manager assigns a list name and returns it to the application.

Logon Dialog. The logon dialog appears when you start the VisualInfo Client. The **Ip2Logon** function also displays the logon dialog. You enter a user ID and password and select the desired Library Server for the session with the logon dialog.

Management Class. A management class is a set of rules that applies to moving objects among storage classes.

Network Table. A text file created in the VisualInfo system by the installation program that records the system specific configuration information. Each node in the network must have a table that identifies the node and lists which nodes in the system it can connect to.

Node. (1) In a network, a point at which one or more functional units connect channels or data circuits. (2) A computer in a network. It is the end point of a link or a junction that is common to two or more links in the network.

Node Name. The name assigned to a node in a VisualInfo system. Node names are symbolic names that have some meaning to you.

Note. A note can be added to a document or a folder.

Note Log Window. The note log window is contained in the low for folders and documents. The note log window contains user-written notes associated with the displayed item. The note log window is implemented as a Presentation Manager edit window.

Object. An object is a stored stream of bits. An object can be an image, spreadsheet, word processing file, note, annotation, or event associated with a document. The VisualInfo system also considers an object to be a document and an item part.

Object Server. The component of the VisualInfo system that contains the object for your end-user applications.

Print Dialog. The print dialog lets you specify user options for the printer profile you want to use, pages of documents you want to print, and parts of documents and folders to you want to print. You see the print dialog by selecting the print option from the Document and selected menus.

Priority. (1) A rank assigned to a task that determines its precedence in receiving resources. (2) The relative significance of one job to another in competing for the allocation of resources.

Privilege. A privilege is a capability that the system administrator gives to a user to either access or perform certain tasks on objects stored in the system.

Query. A request for data from a database or file based on specific conditions.

Retention Period. The retention period is the length of time that objects are to be stored on the Object Server.

Scanner Window. The scanner window lets you specify scanner operational controls. You see the scanner window by selecting the basic scan or advanced scan options from the tools menu in the main container. A view window also might appear, based on user preferences, when the scanner window appears. The format of the scanner window can vary depending upon the scanner used.

Storage Class. Storage classes are categories that logically group storage media available for storing objects.

Storage Group. A storage group is a set of one or more storage volumes containing one or more storage classes.

Table of Contents Window. The table of contents (toc) window is contained in the low. It displays the table of contents for a workbasket or a folder. A low cannot display both a view window and a toc window at the same time.

View Window. The view window displays an image of scanned pages during the scanning process. You see a view window by selecting the basic scan or advanced scan option from the tools menu in the main container.

Workbasket. A workbasket is a container that holds folders and documents to be processed, usually in priority order.

Workflow. A workflow is a sequence of workbaskets that a document or folder travels through while it is being processed.

Workflow Management. In a VisualInfo system a function that automatically distributes documents needing processing to system users.

List of Abbreviations

ACDS	Active Control data Set
ACS	Automatic Class Selection
AD/CYCLE	Application Development Cycle
AIX	Advanced Interactive Executive
API	Application Program Interface
APPC	Advanced Program to Program Communication
APPN	Advanced Peer to Peer Networking
AVT	Attributes Value Table
CDS	Control Data Set
CICS	Customer Information Control System
CM/2	Communications Manager/2
COMMDS	Command Data Set
CSD	Corrective Service Diskette
C/370	C Language for a 370/390 host
DASD	Direct Access Storage Device
DB/2	Database 2
DB2/2	Database 2 for Operating System 2
DCT	CICS Destination Control Table
DLL	Dynamic Link Library
ESA	Enterprise Systems Architecture
FFST	First Failure Support Technology
IBM	International Business Machines Corporation
ISMF	Interactive Storage Management Facility
ISPF/PDF	Interactive System Productivity Facility / Program Development Facility
ITSO	International Technical Support Organization
JCL	Job Control Language
JES2	Job Entry System/2
JES3	Job Entry System/3
LAN	Local Area Network
LCS	Library Control System
MVS	Multiple Virtual Storage (IBM System 370 & 390)
OAM	Object Access Method
OSMC	OAM Storage Management Component
OSR	Object Storage and Retrieval
OS/2	Operating System 2
OTIS	OAM thread isolation support
PCT	CICS Program Control Table
PLT	CICS Program List Table
PL/I	Programming Language One
PPT	CICS Program Properties Table
PSP	Preventive Service Planning
PTF	Program Temporary Fix
RACF	Records Access and Control Facility
RCT	Resource Control Table
RDO	CICS Resource Definition Online
RJE	Remote Job Entry
SAA	Systems application architecture
SCDS	Source Control Data Set
SIT	CICS Systems Initialization Table
SMP/E	System Modification Program Extended
SMS	Storage Management Subsystem
SQL	Structured Query Language
SYSADM	System Administrator
TSO/E	Time Sharing Option Extended
VI/OS2	VisualInfo for Operating System/2
VI	VisualInfo
VI/MVS	VisualInfo MVS/ESA
VSAM	Virtual Storage Access Method
VTAM	Virtual Telecommunications Access Method

Index

Numerics

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

ITSO Technical Bulletin Evaluation RED000

International Technical Support Organization
VisualInfo MVS/ESA Installation
and Customization Guide
June 1995

Publication No. GG24-4488-00

Your feedback is very important to help us maintain the quality of ITSO Bulletins. **Please fill out this questionnaire and return it using one of the following methods:**

- Mail it to the address on the back (postage paid in U.S. only)
- Give it to an IBM marketing representative for mailing
- Fax it to: Your International Access Code + 1 914 432 8246
- Send a note to REDBOOK@VNET.IBM.COM

Please rate on a scale of 1 to 5 the subjects below.
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction	____		
Organization of the book	____	Grammar/punctuation/spelling	____
Accuracy of the information	____	Ease of reading and understanding	____
Relevance of the information	____	Ease of finding information	____
Completeness of the information	____	Level of technical detail	____
Value of illustrations	____	Print quality	____

Please answer the following questions:

a) If you are an employee of IBM or its subsidiaries:

Do you provide billable services for 20% or more of your time? Yes____ No____

Are you in a Services Organization? Yes____ No____

b) Are you working in the USA? Yes____ No____

c) Was the Bulletin published in time for your needs? Yes____ No____

d) Did this Bulletin meet your needs? Yes____ No____

If no, please explain:

_____

_____

What other topics would you like to see in this Bulletin?

_____

_____

What other Technical Bulletins would you like to see published?

_____

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

Name	Address
Company or Organization	
Phone No.	

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM International Technical Support Organization
Department FGC, Location BME
6710 Rockledge Drive
Bethesda, MD
USA 20817-1824

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold
Along Line

Printed in U.S.A.