

Advantage™ VISION:Transact® for CICS®

Installation Guide

Release 7.8



Computer Associates®

B02571-1E

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this documentation for their own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation "as is" without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

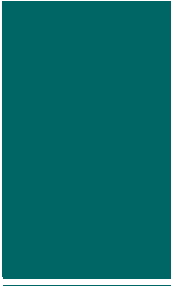
The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227- 19(c)(1) and (2) or DFARS Section 252.227-7013(c)(1)(ii) or applicable successor provisions.

© 2004 Computer Associates International, Inc.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.



Contents

Chapter 1: Introduction

About this Book	1-1
Audience	1-1
z/OS SMP/E Facility	1-2
License Management Program	1-2
Installation Process	1-2
System Tape	1-3
Installation Overview	1-3
Licensing Requirements	1-4
Coding and Integrating Your Licensing Key	1-4
Load System Tape	1-6
SMP/E Setup and the Basic Installation	1-6
Setups	1-7
Documentation	1-7
Contacting Total License Care (TLC)	1-8
Contacting Computer Associates	1-8

Chapter 2: Installation Instructions

Using the Installation Check List	2-2
VISION:Transact Installation Check List	2-2
Step 1 – Copying the Installation Tape Files	2-4
Reviewing the Installation Tape Contents	2-4
Copying the First File	2-5
Copying the Remaining files	2-5
Verifying the VISION:Transact Data Sets	2-6
Step 2 – Product Tailoring Steps (Optional)	2-7
Tailoring the Generator Component Parameter Modules	2-7
Tailoring the CICS Interface Modules	2-8
Tailor the Execution – Time Error and Trace Messages (Optional)	2-9
Tailor the BMS Map for the Error Screen (Optional)	2-10
Tailor the Toolkit Routines (Optional)	2-10
Step 3 – Performing SMP/E Installation Steps	2-11
SMPJOB01	2-13
SMPJOB02	2-15
SMPJOB03	2-16
SMPJOB04	2-16
SMPJOB05	2-16
ZRECEIVE	2-17
ZAPPLY	2-17
ZACCEPT	2-18
Step 4 – Setting Up VISION:Transact	2-20
Define the Executor to CICS	2-20
Creating Working Copies of the Operational Data Sets (Optional)	2-20
Setting up the Generator DB2 Interface Module FIVESQL (Optional)	2-21
Verifying Basic System Installation (Optional)	2-24
Step 5 – Setting Up the ISPF Workbench (Optional)	2-25
Allocating ISPF Workbench Run-Time Libraries	2-25
Allocation Requirements	2-25
Additional Information Concerning Load Libraries	2-28
Using the LIBDEF Feature	2-28
Additional Information on ISPF FILE Allocations	2-28
The List Data Set and Internal Work Files	2-29
The List Data Set	2-29

Internal Work Files	2-30
Invoking ISPF Workbench.	2-31
Adding an Option to a Standard ISPF Primary Menu	2-31
Using the ISPSTART Command.	2-33
Using Other ISPF Workbench Subsections	2-34
Optional Setup for ISPF Workbench	2-35
Preprocessing Your Panel Library	2-35
Allocating the ISPF Log Data Set	2-36
Allocating a New Panel Library	2-36
Running the Preprocess Utility	2-36
Copying the Unprocessed Panels	2-37
Allocating the New Library to ISPLIB.	2-37
Modifying Panels After Preprocessing	2-37
Customizing Job Submission Skeletons	2-38
Step 6 – Setting up the File Definition Conversion Utilities (Optional)	2-39

Chapter 3: Maintenance and Support

Maintenance – Installing the PTFs and APARs	3-2
Support – Problem Reporting	3-3
VISION:Transact Generator Problem Reporting	3-3
VISION:Transact Executor Problem Reporting	3-4
ISPF Workbench Problem Reporting	3-4
Panel Identification	3-4
Unexpected Error Panel	3-5
Problems and User Code	3-5

Appendix A: SMP/E Install JCL Samples

COPYTAPE	A-2
SMPJOB01	A-6
SMPJOB02	A-11
SMPJOB03	A-16
SMPJOB04	A-17
SMPJOB05	A-18
ZACCEPT	A-19
ZAPPLY	A-20
ZRECEIVE	A-21
ZREJECT	A-22
ZRESTORE	A-23

Appendix B: Customization and Setup JCL Samples

CLASM1	B-2
QSCOBEX	B-3
QSCOBLL	B-4
QSCOBPL	B-5
QSDB2BND	B-6
QSDB2EX	B-6
QSINQEX	B-7
QSRESEX	B-8
QSRESLL	B-9
QSRESPL	B-10
TRACICEM	B-10
TRACICS1	B-11
TRACICS2	B-12
TRACICS3	B-14
TRACICTM	B-15
TRASM1	B-15
TRBMSE	B-16
TRCOPY1	B-18
TRCOPY2	B-19
TRCSDUP	B-20
TRDB2A	B-21
TRDB2B	B-23
TRIVP	B-23
WBCOPY1	B-25
WBCOPY2	B-27

Appendix C: VISION:Transact Parameter Modules

CEEUOPT	C-2
FIVEPARM	C-2
CICS Instructions	C-3
IMS Instructions	C-4
Parameter Description	C-4
MARKLIBP	C-17
M5LEPARM	C-20

Appendix D: Sample ISPF Startup CLIST

Appendix E: ISPF Workbench Invocation Panels

XSR@PRIM.....	E-1
M9PRIM.....	E-2

Appendix F: Skeleton and User Panel Listings

M9GCTPU2.....	F-1
M9GCTSBG.....	F-2
M9GCTPU1.....	F-5
M9GCTSFG.....	F-6

Index

Introduction

Thank you for choosing Advantage™ VISION:Transact® for CICS® 7.8 (hereinafter also referred to as VISION:Transact or Transact). Before you install the software, read this chapter for important information.

This book describes how to install VISION:Transact for CICS. Any questions regarding the installation should be directed to Computer Associates Technical Support. For more information, see [Contacting Computer Associates on page 1-8](#).

About this Book

Read this chapter to acquire an understanding of the elements and processes that comprise the installation of VISION:Transact at your site.

Read the subsequent chapters in this book before starting the installation process so that you can have all of the details regarding the installation, customization, and maintenance of VISION:Transact.

Audience

The System Programming Group is usually responsible for software product installation and maintenance because of their SMP/E (System Modification Program Extended) knowledge. This book assumes a working knowledge of the SMP/E Facility and its processes.

A basic standalone SMP/E installation and maintenance approach is presented. For the knowledgeable SMP/E user, there is enough information provided in this book, and the supplied JCL and Control Statements, to allow integration with any site specific SMP/E standards. For the SMP/E novice, this book and the SMP/E documentation should provide enough of the information and concepts you need to complete the SMP/E installation process.

z/OS SMP/E Facility

Starting with release 7.8 of VISION:Transact, the installation and maintenance is managed by and under the control of the z/OS® SMP/E facility as provided by IBM. This process differs significantly from previous releases (7.5 and prior) of VISION:Transact.

License Management Program

VISION:Transact uses the Computer Associates License Management Program (LMP), which provides a standardized and automated approach to the tracking of licensed software.

Installation Process

VISION:Transact is delivered on a tape cartridge. An LMP Product key certificate contains your execution key for each CPU licensed at your site. Other identifying information is provided on the external tape cartridge label.

Save all output generated during the installation, along with the system tape, for future reference.

The basic SMP/E setup and installation process is identical for all users. The first file on the system tape contains JCL for a job that transfers all the remaining system tape files to disk data sets. Once the system tape files are transferred to disk, you will have all of the elements that you need to prepare and complete the installation, customizing, and maintenance processes.

System Tape

The VISION:Transact system tape supplied for the z/OS environment is a standard labeled tape cartridge containing 14 files. The following table shows the order and content of the 14 files on the tape.

File	Description
1	SMP/E control library
2	Source and JCL library
3	Generator load library
4	CICS Executor load library
5	IMS Executor load library
6	CICS Toolkit load library
7	IMS Toolkit load library
8	CICS Toolkit source library
9	IMS Toolkit source library
10	Workbench for ISPF load library
11	Workbench for ISPF panel library
12	Workbench for ISPF messages library
13	Workbench for ISPF CLIST library
14	Workbench for ISPF JCL skeleton library

Installation Overview

The VISION:Transact installation is divided into the following sections:

- [Licensing Requirements on page 1-4](#)
- [Coding and Integrating Your Licensing Key on page 1-4](#)
- [Load System Tape on page 1-6](#)
- [SMP/E Setup and the Basic Installation on page 1-6](#)
- [Setups on page 1-7](#)

Licensing Requirements

VISION:Transact interfaces with the Computer Associates Licensing System using the CA Common Services™ CAIRIM and its CA-LMP facility, which is used to track licensed software.

For more information regarding the CA Common Services CAIRIM and its CA-LMP facility, refer to the *CA Common Services for z/OS and OS/390 Getting Started*.

CA-LMP (License Management Program) is a standardized and automated approach for tracking licensed software. CA-LMP is provided as an integral part of CAIRIM, and is required for VISION:Transact to initialize properly.

If CAIRIM has not already been installed on your system, you must install it before you install and use VISION:Transact. Refer to the *CA Common Services for z/OS and OS/390 Getting Started* for information about installing CAIRIM.

Note: Once CAIRIM has been installed or maintained at GenLevel 9212 or above, CA-LMP support will be available for all Computer Associates products that support CA-LMP.

Coding and Integrating Your Licensing Key

The first task for the installation of VISION:Transact is to get your Computer Associates Licensing Key information coded and integrated into the CAIRIM CA-LMP facility. This is a standard function for all Computer Associates software products. You must add a record with your VISION:Transact CA-LMP Execution Key information, as provided on the key certificate, to the KEYS member in the CAIRIM parameter data set, at the OPTLIB DD statement.

The CA-LMP key certificate you received with VISION:Transact contains the following information:

Field	Description
Product Name	The trademarked or registered product name as licensed for the designated site and the CPUs.
Product Code	A two-character code for the VISION:Transact System.
Supplement	The reference number of your license for VISION:Transact, which may be in the format nnnnnn - nnn.
CPU ID	The code identifying the specific CPU on which VISION:Transact is to be installed.

Field	Description
Execution Key	An encrypted code required by CA-LMP for VISION:Transact initialization. This is also referred to as the LMP Key.
Expiration Date	The date (ddmmmyy) your license for VISION:Transact expires.
Technical Contact	The name of the technical contact at your site who is responsible for the installation and maintenance of this licensed copy of VISION:Transact. This is the person to whom Computer Associates addresses all CA-LMP correspondence.
MIS Director	The name of the Director of MIS (or the person who performs this function at your site). If a person's name is omitted from the certificate, you should supply the actual certificate when correcting and verifying it.
CPU Location	The address of the building containing the CPU on which VISION:Transact is installed.

You must add the CA-LMP execution key information, as provided on the key certificate, to the CAIRIM parameters to ensure that VISION:Transact initializes properly. To define a CA-LMP execution key to the CAIRIM parameters, modify the KEYS member in the OPTLIB data set.

The parameter structure for member KEYS is:

```
PROD(pp) DATE(ddmmmyy) CPU(tttt-mmmm/ssssss) LMPCODE(kkkkkkkkkkkkkk)
```

pp	The two-character required product code for VISION:Transact for CICS is 3I.
ddmmmyy	The CA-LMP licensing agreement expiration date (for example, 31JAN04).
tttt-mmmm	The CPU type and model (for example, 3090-0600) on which the product is to run; required. If the CPU type and/or model are less than four characters, insert blank spaces for the unused characters.
ssssss	The serial number of the CPU on which the product is to run; required.
kkkkkkkkkkkkkk	The execution key needed to run the product; required. The CA-LMP execution key can be found on the key certificate that was shipped with the product.

Here is an example of the parameter entry for the CA-LMP:

```
PROD(3I) DATE(31JAN04) CPU(3090-0600/315109) LMPCODE(5149K01131R08ES)
```

For more information about defining the CA-LMP execution keys using the CAIRIM parameters, refer to the *Unicenter TNG Framework for z/OS Installation and Maintenance Guide*.

Load System Tape

File 1 on the system tape is a PDS that contains the JCL (COPYTAPE) for a job that transfers all the system tape files to disk data sets. The only JCL that needs to be prepared by the installer is an IEBCOPY job to copy the contents of file 1 to a PDS. This JCL is shown in [Figure 2-2 on page 2-5](#) as part of the first step of the installation process.

Once the first system tape file is loaded to disk, you review, tailor, and submit the JCL in member COPYTAPE. This transfers system tape files 2-14 to disk data sets. The main consideration here is the high-level index name assigned to the disk data sets.

[Appendix A, “SMP/E Install JCL Samples”](#) contains an alphabetic list of all the JCL members used to perform the SMP/E installation and perform product maintenance.

[Appendix B, “Customization and Setup JCL Samples”](#) contains an alphabetic list of all the JCL members used to perform the product tailoring and setup. A detailed list of the delivered JCL members follows the alphabetic list.

Once all the system tape files are copied to disk data sets, all the elements (JCL, control statements, source, load modules, etc.) needed for the VISION:Transact installation process are available to the installer. At this point, you can save the system tape; you only need the disk data sets to complete the installation process.

See [Chapter 2, “Installation Instructions”](#) for more information.

SMP/E Setup and the Basic Installation

This portion of the installation process uses the JCL and control statements in the PDS from file 1 on the system tape, TRANSACT.SMPE.SMPCNTL. There are jobs that define and allocate the following items:

- SMP/E target and distribution libraries for VISION:Transact
- An SMP/E CSI (Consolidated Software Inventory) library for tracking the activities
- SMP/E work data sets

The SMP/E setup has the following steps:

1. The elements from the indirect libraries, unloaded from the VISION:Transact system tape, are stored into the SMP/E work data sets using the RECEIVE operation. These elements include all the latest PTFs and APARs for VISION:Transact.

2. The SMP/E APPLY processing is performed to update the target libraries based on the Modification Control Statements (MCS) and a SMPJCLIN job stream. Essentially, the elements from the SMP/E work data sets are transferred to the target libraries. These elements include all the latest PTFs.
3. The SMP/E ACCEPT processing is performed to update the distribution libraries based on the Modification Control Statements (MCS) and a SMPJCLIN job stream. The elements from the SMP/E work data sets are transferred to the distribution libraries. This includes all the latest PTFs.

At this point, the SMP/E setup and the basic installation are complete. The target and distribution libraries are synchronized.

You should examine the APARs that are supplied with the product and APPLY those that you choose to include. You should also apply all PTFs that are supplied with the product. (These PTFs correct problems or deficiencies in the product that were discovered after the product code was frozen.)

The SMP/E concept is to APPLY to the target library and test the APAR/PTF. If you are not satisfied with the tests, you can RESTORE the modified target elements to their previous state from the distribution libraries. If the modifications perform as expected, you permanently ACCEPT the modifications into your distribution libraries. There is no direct method for undoing modifications once the ACCEPT processing is run.

The VISION:Transact SMP/E installation approach uses Indirect Libraries rather than Relative Files. The use of Indirect Libraries may require some adjustment for systems programmers more familiar with Relative Files.

See [Chapter 2, “Installation Instructions”](#) for more information.

Setups

VISION:Transact setup includes defining the product to CICS, allocating and initializing system files, setting up the Definition Processor, and additional optional steps. This process also includes the VISION:Transact IVP (Installation Verification Process).

See [Chapter 2, “Installation Instructions”](#) for more information.

Documentation

The documentation for VISION:Transact is located on the CD included with your installation package. This CD also contains the Readme file, which we recommend you review prior to installing this product. See the *Advantage VISION:Transact for CICS Getting Started* guide for details on use of this CD.

Contacting Total License Care (TLC)

TLC is available Monday-Friday 7 am - 9 pm Eastern Time in North America and 7 am - 7 pm United Kingdom time. Additionally, 24-hour callback service is available for after hours support. Contact TLC for all your licensing requirements.

Be prepared to provide your site ID for product activation.

To activate your product, use one of the following:

Location	Phone	email
North America:	800-338-6720 (toll free) 631-342-5069	help@licensedesk.cai.com
Europe:	00800-1050-1050	euro.tlc@ca.com
If your company or local phone service does not provide international access, please call your local Computer Associates office and have them route you to the above number.		
Australia:	1-800-224-852	
New Zealand:	0-800-224-852	
Asia Pacific:	800-224-852	
Brazil:	55-11-5503-6100	
Japan:	Not available	JPNTLC@ca.com

Contacting Computer Associates

For technical assistance with this product, access the Computer Associates web site at ca.com for a complete list of locations and phone numbers for Technical Support. Technical Support is available 24 hours a day, seven days a week.

Installation Instructions

This chapter explains how to install VISION:Transact and verify the installation. When you finish, you will have installed and set up the following components, included on the VISION:Transact installation tape:

- The VISION:Transact Generator — generates screen maps and object code
- The VISION:Transact Executor — provides runtime execution services
- The VISION:Transact Toolkit — collection of useful service functions
- The ISPF Workbench — source code preparation and maintenance facility

Note: Read all of this chapter before you begin the installation.

This chapter describes:

- Using the VISION:Transact installation check list.
- Copying the installation tape.
- Performing the SMP/E portion of the installation.
- Setting up VISION:Transact.
- Defining the VISION:Transact Executor to CICS
- Setting up the ISPF Workbench
- Setting up the COBOL, DB2, VISION:Inquiry, and VISION:Results file definition conversion utilities

Note: Dataset references throughout this chapter use the default high-level qualifier delivered on the product tape. When you assign your own high-level qualifier, make the appropriate adjustment.

Using the Installation Check List

This section provides you with the installation checklist.

- This check list summarizes the steps involved in the VISION:Transact installation process. Carefully review this check list and become familiar with each step before proceeding.
- You can use this check list to document and track your progress as you proceed with the actual VISION:Transact installation.

Note: The green underlined text steps (Step 1, Step2, ...) in the checklist are hypertext links to the section by the same title. When you are reading this document online, click the green underlined text to go to the text.

VISION:Transact Installation Check List

[Step 1 — Copying the Installation Tape Files on page 2-4](#)

- _____ Copy the SMP control library file from tape to disk. See [Figure 2-2](#).
- _____ Allocate and load remaining disk files using provided JCL (COPYTAPE).
- _____ Verify the VISION:Transact data sets. See [Figure 2-3](#).

[Step 2 — Product Tailoring Steps \(Optional\) on page 2-7](#)

- _____ Modify the MARKLIBP WORKLIB member per your requirements; use WORKLIB job CLASM1 to assemble and linkedit your modifications. (Optional if no modifications made.)
- _____ Modify the FIVEPARM and M5LEPARM WORKLIB members per your requirements; use WORKLIB job TRASM1 to assemble and linkedit your modifications. (Optional if no modifications made.)
- _____ Modify the CEEUOPT, MZCICS, and MZCOMM WORKLIB members per your requirements; use WORKLIB jobs TRACICS1 and TRACICS2 to assemble and linkedit your modifications. (Optional if no modifications made.)
- _____ Modify the FIVCM078 and FIVCT078 WORKLIB members to tailor the execution-time error and trace messages; use WORKLIB jobs TRACICEM and TRACICTM respectively to assemble and linkedit your modifications. (Optional if no modifications made.)
- _____ Modify the FIVCE07 WORKLIB member to tailor the error screen display when a CICS 3270 application fails; use WORKLIB job TRBMSE to compile, assemble, and linkedit the error display map set with your modifications. (Optional if no modifications made.)
- _____ Modify the Toolkit as described in the *Advantage VISION:Transact for CICS Toolkit Reference Guide*. (Optional if no modifications made.)

[Step 3 — Performing SMP/E Installation Steps on page 2-11](#)

Note: You need to modify the JCL in the jobs referenced below to conform to your requirements and conventions.

VISION:Transact Installation Check List

- _____ Allocate the SMP/E datasets and initialize the CSI (SMPJOB01).
- _____ Define the SMP/E Global, Target, and Distribution zones and related datasets (SMPJOB02).
- _____ Receive the product SYSMOD definitions (SMPJOB03).
- _____ Apply the product SYSMOD to the Target Zone (SMPJOB04).
- _____ Accept the product SYSMOD to the Distribution Zone (SMPJOB05).
- _____ Use jobs ZRECEIVE, ZAPPLY, and ZACCEPT any PTFs (members PTFxxxxxx) to install any PTFs shipped with the product.
- _____ Review the APARs (members APARxxxxxx) shipped with the product and use the jobs ZRECEIVE, ZAPPLY, and ZACCEPT to install the APARs you select. (Optional)

Step 4 – Setting Up VISION:Transact on page 2-20

- _____ Define the VISION:Transact runtime programs to CICS using WORKLIB job TRCSDUP.
- _____ Create working copies of the operational data sets.
- _____ Preprocess, assemble, link, and bind FIVESQL module for DB2 support using WORKLIB jobs TRDB2A and TRDB2B. (Optional if no DB2 support required.)
- _____ Add the appropriate DD statement to the DFHRPL concatenation in your CICS startup JCL for the VISION:Transact runtime library (CICSLIB).
- _____ Verify basic system installation. (Optional)

Step 5 – Setting Up the ISPF Workbench (Optional) on page 2-25

- _____ Allocate ISPF Workbench runtime data sets.
- _____ Set up the ISPF Workbench start method.
- _____ Preprocess the panel library. (Optional)
- _____ Customize job submission skeletons. (Optional)

Step 6 – Setting up the File Definition Conversion Utilities (Optional) on page 2-39

- _____ Bind the DB2 utility DBRM. (Optional)
- _____ Relink the COBOL utility with the CA-Panvalet or CA-Librarian interface. (Optional)
- _____ Relink the VISION:Results utility with the CA-Panvalet or CA-Librarian interface. (Optional)

Step 1 — Copying the Installation Tape Files

The first step of the installation process is to copy the installation tape files to disk.

Reviewing the Installation Tape Contents

Each file on the tape has the data set name TRANSACT.CICS.FILE*n*, where *n* is the file number on the tape. Review the files on the installation tape, as seen below.

File	Contents
1	SMP/E control library.
2	Work library (Setup source, macros, and JCL)
3	Generator load library
4	CICS Executor load library
5	IMS Executor load library
6	CICS Toolkit load library
7	IMS Toolkit load library
8	CICS Toolkit source library
9	IMS Toolkit source library
10	Workbench for ISPF load library
11	Workbench for ISPF panel library
12	Workbench for ISPF message library
13	Workbench for ISPF skeleton library
14	Workbench for ISPF CLIST library

Figure 2-1 VISION:Transact Installation Tape Files

Copying the First File

Use the JCL shown in [Figure 2-2](#) to copy the first file from the tape to the PDS named TRANSACT.TR078.SMPCNTL.

```
//          JOB . . .
//*
//*****
//* THIS JOB WILL LOAD THE FIRST FILE ON THE INSTALLATION TAPE (THE *
//* SMP/E CONTROL FILE) FROM TAPE TO DISK. MODIFY THE JOB STATEMENT *
//* STATEMENT, ADD ANY REQUIRED TAPE SETUP STATEMENTS, AND CHANGE ALL *
//* lowercase ENTRIES IN THE JCL TO APPROPRIATE VALUES FOR YOUR *
//* INSTALLATION. *
//*****
//* DELETE PREVIOUS SMP/E LIBRARY. *
//*****
//STEP1 EXEC PGM=IEFBR14
//DELDD DD DSN=transact.smpe.SMPCNTL,DISP=(MOD,DELETE), <-----
//        UNIT=sysda,SPACE=(TRK,1) <-----
//*****
//* ALLOCATE NEW SMP/E LIBRARY. *
//*****
//STEP2 EXEC PGM=IEBCOPY,REGION=2M
//SYSPRINT DD SYSOUT=*
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//IN DD DSN=VISION.TRANSACT.FILE1,
//        UNIT=tape, <-----
//        DISP=(OLD,KEEP), <-----
//        VOL=(PRIVATE,RETAIN,SER=nnnnnn), <-----
//        LABEL=(1,SL)
//OUT DD DSN=transact.smpe.SMPCNTL, <-----
//        DISP=(,CATLG,DELETE), <-----
//        UNIT=sysda, <-----
//        VOL=SER=vvvvvv, <-----
//        SPACE=(TRK,(15,5,10)),
//        DCB=(LRECL=80,BLKSIZE=27920,RECFM=FB,DSORG=PO)
//SYSIN DD *
// COPY INDD=IN,OUTDD=OUT
//
```

Figure 2-2 Copying the First File

Copying the Remaining files

Note: Green text indicates a [hypertext link](#). Clicking on “Figure” takes you to that figure. Clicking on a chapter or appendix title, takes you to that chapter or appendix.

After you copy the JCL library, use member COPYTAPE in TRANSACT.TR078.SMPCNTL to copy all of the other files on the tape. You complete the remainder of the installation by modifying and submitting the copied JCL members. Each JCL member consists of one or more in-stream procedures followed by one or more executions of the procedures.

1. Add appropriate JOB and tape setup statements.
2. Customize the PROC parameters on the EXEC statements, following the PEND statement.

Verifying the VISION:Transact Data Sets

The 14 data sets use 600 tracks on a 3390 device. When you finish copying the tape, verify that you have created the following data sets with your selected high-level qualifiers.

DSN	Dir. Blocks	No. of 3390 tracks	DCB Information	Tape File No.
TRANSACT.TR078.CICSLIB	5	15 U	0 27998	4
TRANSACT.TR078.CICSTKL	10	15 U	0 27998	6
TRANSACT.TR078.CICSTKS	15	15 FB	80 27920	11
TRANSACT.TR078.GENLIB	50	180 U	0 27998	3
TRANSACT.TR078.IMSLIB	5	15 U	0 27998	5
TRANSACT.TR078.IMSTKL	10	15 U	0 27998	3
TRANSACT.TR078.IMSTKS	15	15 FB	80 27920	9
TRANSACT.TR078.SMPCNTL	10	15 FB	80 27920	1
TRANSACT.TR078.WORKLIB	10	90 FB	80 27920	2
TRANSACT.WB061.CLIST	10	20 FB	80 27920	14
TRANSACT.WB061.LOADLIB	40	60 U	0 27998	10
TRANSACT.WB061.MSGS	40	15 FB	80 27920	12
TRANSACT.WB061.PANELS	250	120 FB	80 27920	11
TRANSACT.WB061.SKELS	5	10 FB	80 27920	13
Total		600		

Figure 2-3 VISION:Transact Installation Data Sets

Step 2 — Product Tailoring Steps (Optional)

VISION:Transact contains parameters, options, settings, messages, and screens that can be tailored for each installation. The default values are in modules that can be modified, assembled, and link-edited as needed.

If no modifications are required, you can bypass this step. You can also perform this step later, even after you have begun to use the product if this best meets your needs. However, if you know what modifications you prefer or require, making these modifications now will preserve them within the SMP/E framework and they will not need to be lost by some future APAR or PTF that may be applied.

Tailoring the Generator Component Parameter Modules

The Generator component of VISION:Transact contains three parameter modules that can be tailored to your requirements and preferences.

The three parameter modules are:

FIVEPARM	VISION:Transact primary parameter module
M5LEPARM	VISION:Transact Language Environment parameter module
MARKLIBP	VISION:Transact COMLIB sub-component parameter module

These parameter modules are listed in [Appendix C, “VISION:Transact Parameter Modules”](#). The source code for the parameter modules are in the WORKLIB PDS.

Before you modify any of these modules, create a backup copy of the original source code for future reference.

Once you review and modify the parameter modules, use the JCL member TRASM1 in the WORKLIB PDS to assemble and link-edit the FIVEPARM and M5LEPARM modules into the VISION:Transact GENLIB load library. Use JCL member CLASM1 in the WORKLIB PDS to assemble and link-edit the MARKLIBP module into the GENLIB load library. You only need to run the job steps that correspond to the modules being changed. For each step that you choose to run, specify the WORKLIB PDS name, the source member name, and the load library name. The FIVEPARM, M5LEPARM, and MARKLIBP modules are independently loaded at run time and, therefore, can be stored in load libraries that are separate from the VISION:Transact GENLIB library. If used, you must concatenate the load library before the VISION:Transact GENLIB library and COMLIB load libraries. In this way, you can tailor several different versions of these modules to satisfy varying requirements for separate user groups.

Tailoring the CICS Interface Modules

The VISION:Transact CICS interface modules are delivered in source in the WORKLIB dataset so that you can customize them if necessary. You only need to perform this step if you modify the delivered source by changing the source code members per the comments at the beginning of each source code module. Three source code members may be modified in this way.

The first source code member that may be considered for modification is CEEUOPT. This member contains the run-time Language Environment parameters used by VISION:Transact to interface with CICS. The source code for CEEUOPT is listed in [Appendix C, “VISION:Transact Parameter Modules”](#).

The second source code member that may be customized is MZCICS. This module serves as the VISION:Transact CICS interface for 3270 Terminal programs and contains the executable code that invokes CICS services for these applications. The customizable settings in this module are:

1. The trace output destinations
2. Whether the DL/I database interface should be enabled
3. Whether an error screen should be displayed when any field becomes invalid during application execution. A field may become invalid due to arithmetic overflow, assignment from a missing data segment or another invalid field, or the field contains bad content (e.g. bad packed decimal data).

The relevant source code in member MZCICS is shown below. Change the SETB operand values from 1 to 0 or vice versa as appropriate for your site. A value of 1 means YES and a value of 0 means NO.

```
*****;RL78 00120008
*      ;RL78 00130008
* I N T E R F A C E   C U S T O M I Z A T I O N S      ;RL78 00140008
*      ;RL78 00150008
*****;RL78 00160008
*      ;RL78 00170008
* CUSTOMIZE TRACE OUTPUT DESTINATIONS                ;RL78 00180008
*      ;RL78 00190008
&TSQUEUE SETB 1      USE TEMPORARY STORAGE QUEUE? 1=YES, 0=NO ;RL78 00200008
&TDQUEUE SETB 1      USE TRANSIENT DATA QUEUE? 1=YES, 0=NO  ;RL78 00210008
*      (CSD DEFINE FOR TDQUEUE REQUIRED IF YES) ;RL78 00220008
&SCREEN SETB 1       USE 3270 SCREEN? 1=YES, 0=NO ;RL78 00230008
*      ;RL78 00240008
* CUSTOMIZE DL/I DATABASE AVAILABILITY (RSM102) ;RL78 00250008
*      ;RL78 00260008
&DLICALL SETB 1      DL/I DATABASE SYSTEM AVAILABLE? 1=YES, 0=NO ;RL78 00270008
*      ;RL78 00280008
* CUSTOMIZE DISPLAY OF ERROR SCREEN WHEN INVALID FIELDS (RSM105) ;RL78 00290008
*      ;RL78 00300008
&INVFLD SETB 0       ERROR DISPLAY IF INVALID FIELDS? 1=YES, 0=NO ;RL78 00310008
*****;RL78 00320008
```

Figure 2-4 Relevant source code in member MZCICS

The third source member that may be customized is MZCOMM. This module serves as the VISION:Transact CICS interface for COMMAREA programs and contains the executable code that invokes CICS services for these applications. The customizable settings in this module are:

1. The trace output destinations
2. Whether the DL/I database interface should be enabled.

The relevant source code for member MZCOMM is shown below: Change the SETB operand values from 1 to 0 or vice versa as appropriate for your site. A value of 1 means YES and a value of 0 means NO.

```
*****;RL78 0026000
* ;RL78 0027000
* I N T E R F A C E   C U S T O M I Z A T I O N S ;RL78 0028000
* ;RL78 0029000
*****;RL78 0030000
* ;RL78 0031000
* CUSTOMIZE TRACE OUTPUT DESTINATIONS ;RL78 0032000
* ;RL78 0033000
&TSQUEUE SETB 1 USE TEMPORARY STORAGE QUEUE? 1=YES, 0=NO ;RL78 0034000
&TDQUEUE SETB 1 USE TRANSIENT DATA QUEUE? 1=YES, 0=NO ;RL78 0035000
* (CSD DEFINE REQUIRED FOR TDQUEUE IF YES) ;RL78 0036000
* ;RL78 0037000
* CUSTOMIZE DL/I DATABASE SYSTEM AVAILABILITY ;RL78 0038000
* ;RL78 0039000
&DLICALL SETB 1 DL/I DATABASE SYSTEM AVAILABLE? 1=YES, 0=NO ;RL78 0040000
* ;RL78 0041000
*****;RL78 0042000
```

Figure 2-5 Relevant source code for member MZCOMM

If modifications are made to any of these source code members, the corresponding job in the WORKLIB data set must be run:

1. TRACICS1 if modifications are made to either CEEUOPT or MZCICS
2. TRACICS2 if modifications are made to either CEEUOPT or MZCOMM

A return code of 4 in the assembler steps for MZCICS and MZCOMM is expected — the job completed successfully.

Tailor the Execution — Time Error and Trace Messages (Optional)

All error and trace messages issued during the online execution of VISION:Transact-generated applications are provided in source so that you can customize them.

Because the error and trace messages are installed already, you need only perform these steps if you modify the error or trace message module source. The following table identifies the source member names for the various error message or trace message modules along with the corresponding JCL member names that you can use to install any customizations. The source members and the JCL members are in the WORKLIB PDS.

Source Member	Type	Environment	JCL Member
FIVCM078	Error	CICS	TRACICEM
FIVCT078	Trace	CICS	TRACICTM

Tailor the BMS Map for the Error Screen (Optional)

When a VISION:Transact-generated application fails during execution, an error screen displays on the originating terminal. The error screen is also sent to an error destination terminal if one is defined to the VISION:Transact Generator using FIVEPARM (ERRDEST keyword).

The BMS map specifies a 3270 device with 24 lines and 80 columns and is included in the VISION:Transact CICS Executor library. This step is only necessary if you want to tailor the error screen. The VISION:Transact screen definition source for the error screen is provided in member FIVCE07 of the WORKLIB PDS. Modify the source code in this member to tailor the error screen. Use the member TRBMSE in the WORKLIB PDS to assemble and link-edit the tailored BMS map for the screen.

Tailor the Toolkit Routines (Optional)

The VISION:Transact Toolkit is a set of subroutines that enable you to perform some useful functions that are not readily available using native VISION:Transact programming operations. The Toolkit programs are delivered as both Assembler source code programs and as ready-to-use CICS program modules.

You can optionally tailor these Toolkit programs to conform to your particular requirements. See Chapter 2 of the *Advantage VISION:Transact for CICS Toolkit Reference Guide* for instructions on how to perform this tailoring, if you so choose.

You are now ready to proceed with the SMP/E installation steps.

Step 3 — Performing SMP/E Installation Steps

At this point, you need to run eight job streams to establish and define the SMP/E CSI and zones, and install the basic VISION:Transact. These job streams are located in the PDS data set loaded from File 1 of the installation tape, TRANSACT.TR078.SMPCNTL. The following members contain these eight job streams for the basic VISION:Transact installation.

Member Name	Function of the Job within this member
SMPJOB01	Allocate the SMP/E CSI, the SMP/E work datasets, and all of the associated datasets for the target and distribution libraries.
SMPJOB02	Define the SMP/E global, target, and distribution zones.
SMPJOB03	Receive the MCS (Modification Control Statements) and product SYSMOD into the SMP/E global zone.
SMPJOB04	Apply the product SYSMOD (modules and elements) to the target libraries.
SMPJOB05	Accept the product SYSMOD (modules and elements) to the distribution libraries.
ZRECEIVE	Receive the APAR and PTF SYSMODS into the SMP/E global zone.
ZAPPLY	Apply the APAR and PTF SYSMODS to the target libraries.
ZACCEPT	Accept the APAR and PTF SYSMODS to the distribution libraries.

The following members are additional members in the TRANSACT.TR078.SMPCNTL dataset that are referenced by the above jobs. These are control statements and APAR/PTF SYSMODS.

Member Name	Description
CDDX078	The JCLIN (IEBCOPY statements) for the target and distribution libraries.
SMPMCS0	MCS member - the header and copyright information for VISION:Transact.
SMPMCS1C	MCS member - the CICS Executor component elements.
SMPMCS1G	MCS member - the Generator component elements.
SMPMCS1I	MCS member - the IMS Executor component elements.

SMPMCS2	MCS member - the COMLIB component elements.
SMPMCS3	MCS member - the ISPF Workbench component elements.
SMPMCS4	MCS member - the SAS/C runtime library elements.
SMPMCS5C	MCS member - the CICS Toolkit component elements.
SMPMCS5I	MCS member - the IMS Toolkit component elements.
TR00nnn	APAR and PTF SYSMODS for the Generator or Executor components
WB00nnn	APAR and PTF SYSMODS for the ISPF Workbench component

SMPJOB01

Tailor and run job SMPJOB01 from the TRANSACT.TR078.SMPCNTL data set. In this job, you allocate all the data sets needed by SMP/E to manage, control and maintain VISION:Transact and its components. This includes the SMP/E CSI, the associated work data sets, and the distribution and target libraries.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming and allocation conventions.
- Change the IDCAMS control statements and JCL in the DEFINE and INIT steps (end of job stream) to match the dataset names specified with the SET command above.
- Add any necessary dataset allocation parameters (UNIT, VOLUMES, STORCLASS, etc.) to the IDCAMS control statements in the DEFINE step (at end of job stream).

Any existing or previously defined data sets of the same names are deleted before the new data sets are allocated.

The following data sets are allocated (shown with the default high-level qualifier):

TRANSACT.SMPE.CSI
TRANSACT.SMPE.CSI.DATA
TRANSACT.SMPE.CSI.INDEX
TRANSACT.SMPE.SMPLOG
TRANSACT.SMPE.SMPLOGA
TRANSACT.SMPE.SMPMTS
TRANSACT.SMPE.SMPPTS
TRANSACT.SMPE.SMPSCDS
TRANSACT.SMPE.SMPSTS
TRANSACT.TR078.TRDCICS
TRANSACT.TR078.TRDGEN
TRANSACT.TR078.TRDIMS
TRANSACT.TR078.TRDTKCL
TRANSACT.TR078.TRDTKCS
TRANSACT.TR078.TRDTKIL

TRANSACT.TR078.TRDTKIS
TRANSACT.TR078.TRDWORK
TRANSACT.TR078.TRTCICS
TRANSACT.TR078.TRTGEN
TRANSACT.TR078.TRTIMS
TRANSACT.TR078.TRTTKCL
TRANSACT.TR078.TRTTKCS
TRANSACT.TR078.TRTTKIL
TRANSACT.TR078.TRTTKIS
TRANSACT.TR078.TRTWORK
TRANSACT.WB061.WBDCLIST
TRANSACT.WB061.WBDLOAD
TRANSACT.WB061.WBDMSG
TRANSACT.WB061.WBDPANEL
TRANSACT.WB061.WBDSKELS
TRANSACT.WB061.WBTCLIST
TRANSACT.WB061.WBTLOAD
TRANSACT.WB061.WBTMSG
TRANSACT.WB061.WBTPANEL
TRANSACT.WB061.WBTSKELS

SMPJOB02

Tailor and run job SMPJOB02 from the TRANSACT.TR078.SMPCNTL data set. In this job, you define the VISION:Transact global, distribution, and target zones in the CSI. This is the information needed by SMP/E to manage, control, and maintain VISION:Transact.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming and conventions.
- Change the CSI dataset qualifiers in the ZONEINDEX entry to match the SET statement parameters.
- Change the DA names for the DDDEF entries beginning with 'SMP' and with 'TR' and with 'WB' to the dataset names that were allocated in SMPJOB01.
- Find the two occurrences of the DDDEF entries for RESLIB and change the DA names to your IMS system residence library. If you do not have the IMS Database product from IBM, remove all references to the RESLIB entry.
- Find the two occurrences of the DDDEF entries for DSNLOAD and change the DA names to your DB2 system load library. If you do not have the DB2 Database product from IBM, remove all references to the DSNLOAD entry.
- Find the occurrence of the DDDEF entry for SDFHLOAD and change the DA name to your CICS system load library.
- Find the occurrence of the DDDEF entry for SDFHMAC and change the DA name to your CICS macro library.
- Find the occurrence of the DDDEF entry for SCEEMAC and change the DA name to your Language Environment Assembler macro library.

Make any other changes as appropriate for your installation.

SMPJOB03

Tailor and run job SMPJOB03 from the TRANSACT.TR078.SMPCNTL data set. In this job, you RECEIVE the Modification Control Statements (MCS) and VISION:Transact software system elements (SYSMODS) into the global zone and SMP/E data sets.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming conventions.

SMPJOB04

Tailor and run job SMPJOB04 from the TRANSACT.TR078.SMPCNTL data set. In this job, you APPLY the VISION:Transact software system elements (SYSMODS) into the target libraries.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming conventions.

SMPJOB05

Tailor and run job SMPJOB05 from the TRANSACT.TR078.SMPCNTL data set. In this job, you ACCEPT the VISION:Transact software system elements (SYSMODS) into the distribution libraries.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming conventions.

ZRECEIVE

Tailor and run job ZRECEIVE from the TRANSACT.TR078.SMPCNTL data set for each APAR and PTF SYSMOD you want to receive into your SMP/E global zone.

Note:

Prior to VISION:Transact Release 7.8, PTFs were known as SMs (System Modifications) or GSMs (General System Modifications). These types of patches apply to all systems and correct or enhance the software system.

Prior to VISION:Transact Release 7.8, APARs were known as RSMs (Restricted System Modifications). These types of patches only apply, if at all, to sites with unique requirements, or they change the way the product operates, as described in the product documentation.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming conventions.
- Change the member name for the SMPPTFIN DSN to the APAR or PTF member name you want to receive.

Note: After running ZRECEIVE, you should retain the SYSOUT dataset with ddname SMPRPT that the job produces. The SMPRPT listing provides a list of the APARs and PTFs received by ZRECEIVE. Use the items listed under the SYSMOD column to specify the APPLY SELECT parameters in the subsequent jobs ZAPPLY and ZACCEPT.

ZAPPLY

Tailor and run job ZAPPLY from the TRANSACT.TR078.SMPCNTL data set. In this job, you APPLY the VISION:Transact software system APAR and PTF SYSMODS into the target libraries.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming conventions.
- Change the "APPLY SELECT(NNNNNNN)" to name the APAR/PTF that is to be applied. Change only the 'NNNNNNN' portion of the statement.
- Apply APARS/PTFS one at a time, specifying only one item on each invocation of the job.

This job is used in one of two ways:

- During the installation process, to apply the initial PTFs and, optionally, apply APARs that are delivered with the system.
- After product installation, to apply maintenance to the product when necessary.

When you run this job during product installation, refer to the SMPRPT listing from the ZRECEIVE output. This listing gives you the names of the PTFs that need to be applied in ZAPPLY. Apply each PTF in a separate invocation of the JOB contained in ZAPPLY. All PTFs listed in the SMPRPT output are required; APARs are optional, and usually not needed. If you have questions about the APARs or PTFs, contact Technical Support (see [Contacting Computer Associates on page 1-8](#)).

Note: PTFs must be applied in ascending sequence, by the PTF name.

After ZAPPLY is complete, the target libraries contain the APARs and PTFs that were applied. If your operational libraries are a copy of the target zone libraries, your operational libraries (if they already exist) may need to be refreshed.

ZACCEPT

Tailor and run job ZACCEPT from the TRANSACT.TR078.SMPCNTL data set for each APAR and PTF SYSMOD you want to accept into the distribution libraries.

Instructions for tailoring this job are contained in JCL comments within the member. To tailor this job, you must:

- Supply a valid "JOB" JCL statement.
- Change the SET command parameters to reflect your dataset naming conventions.
- Change the "ACCEPT SELECT(NNNNNNN)" to name the APAR/PTF that is to be accepted. Change only the 'NNNNNNN' portion of the statement.
- Accept APARS and PTFS one at a time, specifying only one item on each invocation of the job.

This job is used in one of two ways:

1. During the installation process, to accept the initial PTFs and, optionally, APARs that are delivered with the system.
2. After product installation, to accept maintenance to the product when necessary.

When you run this job during product installation, refer to the list of PTFs that you applied in ZAPPLY, which in turn came from the SMPRPT listing from ZRECEIVE. This will be the list of PTFs that you accept in ZACCEPT. All PTFs applied in ZACCEPT must be accepted with ZACCEPT; APARs are optional. Accept each PTF in a separate invocation of the job contained in ZACCEPT.

Do not run the ZACCEPT job for any APARs that you think you might want to remove later on. Once an APAR is accepted, the only way to remove it is to re-install the entire product.

Note: PTFs must be applied in ascending sequence, by the PTF name.

After ZACCEPT is complete, both the distribution libraries and the target libraries contain the APARs and PTFs that were accepted. If your operational libraries are a copy of the distribution libraries, your operational libraries (if they already exist) may need to be refreshed.

Step 4 – Setting Up VISION:Transact

This step involves the following:

- Define the VISION:Transact Executor to CICS
- Create working copies of the operating libraries (Optional)
- Set up the DB2 interface module for DB2 database access (Optional)

Define the Executor to CICS

Because the generated VISION:Transact programs invoke Executor routines, and possibly Toolkit routines, you must update the CICS System Definition (CSD) to include definitions for these objects. Use the job TRCSDUP in the WORKLIB PDS to update your CSD. This update defines 22 program entries, one transaction entry, one mapset entry, and one trace output destination entry. The trace output destination may be removed from the CSD definitions if the Transient Data Queue was disabled as a trace destination during the tailoring in [Step 2 — Product Tailoring Steps \(Optional\) on page 2-7](#).

Creating Working Copies of the Operational Data Sets (Optional)

At this point in the installation process, the SMP/E target and distribution libraries contain identical copies of the VISION:Transact system. You may use the target libraries as your operational libraries or you may make working copies of these libraries for your operational use. If you choose to make working copies of the libraries required for operational use, the following steps are used to create the working copies of these libraries:

1. For the Generator and Executor components, make a copy of the VISION:Transact GENLIB, CICSLIB and CICSTKL load libraries for operational use. Use either job TRCOPY1 or TRCOPY2 from the TRANSACT.TR078.WORKLIB (default name) data set. Both jobs use IEBCOPY to copy the target GENLIB, CICSLIB, and CICSTKL load libraries to operational copies. The difference between the two jobs is that TRCOPY1 re-creates the operational copies of the libraries by deleting and re-allocating them before the target libraries are copied. Job TRCOPY2 copies the target libraries to existing libraries, with the REPLACE option, and then compresses the existing copy of the libraries.
2. For the ISPF Workbench component, make a copy of the VISION:Transact CLIST, LOADLIB, MSGS, PANELS, and SKELS libraries. Use either job WBCOPY1 or WBCOPY2 from the TRANSACT.TR078.WORKLIB (default name) data set. Both jobs use IECOPY to copy the target CLIST, LOADLIB, MSGS, PANELS, and SKELS libraries to operational copies. The difference

between the two jobs is that WBCOPY1 re-creates the operational copies of the libraries by deleting and re-allocating them before the target libraries are copied. Job WBCOPY2 copies the target libraries to existing libraries, with the REPLACE option, and then compresses the existing copies of the load libraries.

Setting up the Generator DB2 Interface Module FIVESQL (Optional)

Only customers that expect to use VISION:Transact for applications that access DB2 databases need to perform this step.

The FIVESQL module contains the SQL statements used by the Generator to access the DB2 catalog. The Generator accesses the DB2 catalog to verify that the DB2 column names in your file definition are defined to DB2 and to construct the appropriate code sequence to map the DB2 table columns to your VISION:Transact application.

This module must go through the DB2 application program preparation process described in the IBM DATABASE 2™ Application Programming Guide manuals. The FIVESQL source code module supplied in the WORKLIB PDS is assembled to generate source statements that become the plan required by DB2 to compile VISION:Transact applications that access DB2 tables.

The FIVESQL module, as distributed, assumes that the view name to access the SYSCOLUMNS table of the DB2 catalog is SYSIBM.SYSCOLUMNS and that the view name to access the SYSSYNONYMS table of the DB2 catalog is SYSIBM.SYSSYNONYMS.

As necessary:

- Change the SQL statements containing these names to conform to your installation's naming convention before proceeding with the preparation process.
- Consult with your DB2 database administrator before proceeding with the program preparation process.

Note: During the link edit step, some linkage editors could issue warning messages and a condition code 4. You can ignore these. For example, message IEW2646W issued by Binder is a warning about RMODE Conflicts when the IBM Module DSNELI is included. The module FIVESQL is correctly linked, and the message can be ignored.

Use the JCL member TRDB2A in the WORKLIB PDS to assemble and prepare the FIVESQL module. As necessary, confer with your DB2 database administrator before proceeding with this process. You only need to prepare the FIVESQL modules that correspond to the attach facilities that you will use.

After the required FIVESQL module has been preprocessed, compiled, and link-edited, the application plan generated by the DB2 preprocessor must undergo an additional process called binding.

Invoke the BIND function with either DB2I (DB2 Interactive) using the TSO terminal monitor program with the appropriate control statements, or using batch JCL. Note that the VISION:Transact Generator component does not require repeatable read isolation. Specify cursor stability isolation when binding the plans to allow greater concurrent access to your DB2 tables. Refer to the *IBM DATABASE2 Application Programming Guide* for your environment for more information on the bind process. You may need to confer with your DB2 database administrator before proceeding with the bind process.

You can choose either of the two methods to perform the bind process for the prepared FIVESQL module that will be used to Attach to DB2. [Figure 2-6](#), [Figure 2-7](#), and [Figure 2-8](#), respectively, display the panels when doing the bind using the DB2I (DB2 Interactive) facility. Alternatively, you can use the JCL member TRDB2B in the WORKLIB PDS to bind the plan names in a batch job. If at any time you make changes to the FIVESQL module, you need to repeat the preparation and bind process. Be aware that you must use a BIND REPLACE action, not REBIND, when you perform a new bind.

```
COMMAND ==> 5                                DB2I PRIMARY OPTION MENU          SSID:

Select one of the following DB2 functions and press ENTER.

 1 SPUFI                      (Process SQL statements)
 2 DCLGEN                     (Generate SQL and source language declarations)
 3 PROGRAM PREPARATION        (Prepare a DB2 application program to run)
 4 PRECOMPILE                  (Invoke DB2 precompiler)
 5 BIND/REBIND/FREE           (BIND, REBIND, or FREE plans or packages)
 6 RUN                         (RUN an SQL program)
 7 DB2 COMMANDS                (Issue DB2 commands)
 8 UTILITIES                   (Invoke DB2 utilities)
 D DB2I DEFAULTS              (Set global parameters)
 X EXIT                       (Leave DB2I)

PRESS:                                END to exit          HELP for more information
```

Figure 2-6 Using DB2I to Perform the BIND Function (Panel 1)

```
COMMAND ==> 1                                BIND/REBIND/FREE          SSID:

Select one of the following and press ENTER:

 1 BIND PLAN                   (Add or replace an application plan)
 2 REBIND PLAN                  (Rebind existing application plan or plans)
 3 FREE PLAN                    (Erase application plan or plans)
```

Figure 2-7 Using DB2I to Perform the BIND Function (Panel 2)

```

4 BIND PACKAGE          (Add or replace a package)
5 REBIND PACKAGE        (Rebind existing package or packages)
6 REBIND TRIGGER PACKAGE (Rebind existing trigger package or packages)
7 FREE PACKAGE          (Erase a package or packages)
PRESS:  ENTER to process  END to exit          HELP for more information

```

Figure 2-7 Using DB2I to Perform the BIND Function (Panel 2)

```

                                BIND PLAN                                SSID:
COMMAND ==>

Enter DBRM data set name(s):
1 MEMBER ..... ==> FIVESQL
2 PASSWORD ..... ==>
3 LIBRARY ..... ==> DBRMLIB.DATA
4 ADDITIONAL DBRMS? ..... ==> NO      (YES to include more DBRMs)

Enter options as desired:
5 PLAN NAME ..... ==> FIVESQL      (Required to create a plan)
6 CHANGE CURRENT DEFAULTS? .. ==> NO (NO or YES)
7 ENABLE/DISABLE CONNECTIONS? ==> NO (NO or YES)
8 INCLUDE PACKAGE LIST?..... ==> NO (NO or YES)
9 OWNER OF PLAN (AUTHID)..... ==>   (Leave blank for your primaryID)
10 QUALIFIER ..... ==>             (For tables, views, and aliases)
11 CACHESIZE ..... ==>             (Blank, or value 0-4096)
12 ACTION ON PLAN ..... ==> REPLACE (REPLACE or ADD)
13 RETAIN EXECUTION AUTHORITY. ==> YES (YES to retain user list)
14 CURRENT SERVER ..... ==>        (Location name)
15 INCLUDE PATH?..... ==> NO      (NO or YES)

PRESS:  ENTER to process  END to save and exit  HELP for more information

```

Figure 2-8 Using DB2I to Perform the BIND Function (Panel 3)

Verifying Basic System Installation (Optional)

At this point, the basic VISION:Transact Generator and Executor components are installed. Previously in this step you performed various setup functions, some of these being optional. To verify that the basic VISION:Transact Generator is installed and operational, run a simple job stream that exercises several different functions of the system.

Use the WORKLIB member TRIVP to perform the installation verification process. You supply the JOBLIB DD data set names for the VISION:Transact load library. There are no permanent data sets created by this job stream.

VISION:Transact uses the IBM Language Environment (LE). The LE run-time library must be available when running the VISION:Transact Generator.

The job stream contains the following steps. Each step completes with a condition code zero (0000).

1. **INIT**
A library utility run that allocates and initializes a common library (M5LIB). If you set Item Tracking in MARKLIBP to 1 or 3, then you must add an IT statement to the M5INPUT for this step.
2. **DEF**
A definition run that catalogs a file definition and a screen definition in the M5LIB.
3. **GEN**
A generator run that compiles a simple application with one input screen and one output screen. The cataloged file definition and screen definitions are used from the M5LIB. The Advanced Syntax Language (ASL) translator is invoked. The generated application is not executed.
4. **DUMP**
A library utility run that backs up (dumps), reinitializes, and restores the contents of the M5LIB. This is a process that condenses the M5LIB.
5. **LIST**
A definition run that produces a glossary listing of the file definition and screen definitions.

The basic installation is complete and verified. Review [Step 4 – Setting Up VISION:Transact on page 2-20](#), to determine which optional setup steps are needed for your installation

Step 5 – Setting Up the ISPF Workbench (Optional)

The following sections provide the setup requirements for the ISPF Workbench component of VISION:Transact.

Allocating ISPF Workbench Run-Time Libraries

The ISPF Workbench runs as an application under IBM's ISPF/PDF Facility, which is an extension of TSO, and takes advantage of the many standard services available under ISPF.

To integrate ISPF Workbench into the ISPF environment, the ISPF Workbench libraries and the appropriate VISION:Transact, component load libraries must be made available to the ISPF facility.

There are two methods for making ISPF Workbench libraries known to ISPF:

- Add the libraries to the TSO logon procedure or the ISPF startup CLIST.
- Use the ISPF LIBDEF service to dynamically modify your ISPF library concatenations.

Check with the systems group at your facility to confirm which method to use to get the ISPF Workbench libraries allocated for ISPF sessions.

Once you determine how your existing ISPF libraries are being allocated, you can allocate the necessary ISPF Workbench libraries by concatenating them to the existing ISPF library allocations and allocating the ISPF Workbench libraries in the front of each concatenation sequence.

Allocation Requirements

The following list shows the required ISPF ddnames and the ISPF Workbench data sets that are associated with them. ISPF Workbench data set names displayed are the suggested names shown earlier in this document. Change the names to reflect those names actually used during your installation process.

ddname: SYSPROC

Data set name: TRANSACT.WB061.WBTCLIST

Note: The ISPF Workbench CLIST library was delivered on the installation tape as a fixed blocked data set with a record length of 80 and a block size of 27920. If your installation prefers a variable blocked format, copy the contents of this library over to a different CLIST library of the proper format for your installation.

ISPLLIB functions as a task library. It is searched before the STEPLIB allocations, system link libraries, or the system link pack area.

ddname: ISPLLIB
Data set names: TRANSACT.WB061.WBTLOAD (ISPF Workbench)

TRANSACT.TR078.TRTGEN (VISION:Transact Generator)

BUILDER.BL140.SMPE.T.BLSYSL
(Advantage™ VISION:Builder®)

INFORM.R40.TARGET.INTLOAD
(Advantage™ VISION:Inform®)

ISPF Workbench Release 6.1 assumes the following releases of the associated software products, as shown below.

Software Product	Release Number
VISION:Transact	7.8
VISION:Builder	14.0
VISION:Inform	4.0

ddname: ISPLLIB
Data set name: TRANSACT.WB061.WBTPANEL

If you chose to preprocess your ISPF Workbench panel library, concatenate the preprocessed panel library, rather than the panel source library, to this ddname. Preprocessing the panel library is an optional installation step discussed in [Optional Setup for ISPF Workbench on page 2-35](#).

ddname: ISPLMLIB
Data set name: TRANSACT.WB061.WBTMSGs

ISPF uses this ddname to locate the text of all messages issued by ISPF applications like ISPF Workbench.

ddname: ISPSLIB
Data set name: TRANSACT.WB061.WBTSKELS

The ISPSLIB ddname specifies the location of ISPF file-tailoring skeletons used by ISPF Workbench.

ddname: ISPFIL
Data set name: This data set name should reference a file-tailoring output library.

Installations that use the ISPF Workbench Generate facility must preallocate this file.

The IMPORT option of ISPF Workbench requires a data set to save information entered during the IMPORT function dialogs. The entered information is then available from session to session. Preallocate and catalog the data set. The data set characteristics are as follows:

- DSORG PO
- RECFM FB
- LRECL 80
- BLKSIZE multiple of 80
- SPACE (TRK,(5,2,2))

After you define the data set, you need to add the data set to the ISPF startup CLIST allocations. Code the free and allocation entries for the ddname DEFTLIB as follows:

- FREE F(DEFTLIB)
- ALLOC F(DEFTLIB) DA(*user.defined.name*) SHR

To allow for automatic dynamic allocation and cataloging of the data set for each unique user, you can insert the following sample CLIST statements into the startup ISPF CLIST:

```
FREE F(DEFTLIB)
IF &SYSDSN('TRANSACTION.WB078.&SYSUID..TLIB') = OK THEN +
ALLOC F(DEFTLIB) DA('TRANSACTION.WB055.&SYSUID..TLIB') SHR
ELSE +
ALLOC F(DEFTLIB) DA('TRANSACTION.WB078.&SYSUID..TLIB') +
NEW CATALOG UNIT(SYSDA) SPACE(5,2) DIR(2) +
DSORG(PO) RECFM(F B) LRECL(80) BLKSIZE(27920)
```

The current User Id replaces the &SYSUID element entry when the startup CLIST is activated.

[Appendix D, “Sample ISPF Startup CLIST”](#), contains a sample ISPF startup CLIST that shows how to perform the ISPF Workbench library allocations.

Additional Information Concerning Load Libraries

Rather than being allocated to ISPLLIB, your ISPF Workbench load library and related load libraries can be made available to ISPF through a STEPLIB allocation or through system link library allocations. Alternatively, because ISPF Workbench is reentrant, its load modules can be placed in the system link pack area.

Using the LIBDEF Feature

Some installations like to take advantage of the ISPF LIBDEF service. This feature allows you to dynamically modify your ISPF library concatenations based on the ISPF application that you are running.

You can use the LIBDEF service to allocate your ISPF Workbench CLIST, panel, message, and skeleton libraries.

Do not use this feature to allocate load libraries or for file-tailoring of output data sets for VISION:Builder application generation. ISPF Workbench relies on MVS services to find load modules and to obtain the data set name for file tailoring output. MVS services do not recognize allocations performed through the ISPF LIBDEF service. ISPF Workbench will not function properly if LIBDEF is used for ISPLLIB or ISPFILIB allocations.

For more information on the LIBDEF service, refer to the IBM *ISPF Dialogue Management Guide and Reference* manual.

Additional Information on ISPFILIB Allocations

The ISPFILIB allocation is only applicable to VISION:Transact customers using ISPF Workbench. If used, the ISPFILIB allocation must not specify a concatenated sequence of data sets.

During the ISPF Workbench VISION:Transact job submission process, if you specify the keep or keep/submit processing option, the generated JCL or CLIST is written to the data set allocated to ISPFILIB. This data set is often referred to as the file tailoring output data set.

VISION:Transact users must preallocate this file if they are using the generate facility. This preallocation must not be done with the ISPF LIBDEF service (see Using the LIBDEF Feature).

VISION:Transact installations do not need to preallocate ISPFILIB. ISPF Workbench dynamically allocates the file-tailoring output data set as required and uses its own ddname (M9FTOUT) for this purpose. ISPF Workbench does not deallocate any existing ISPFILIB assignments.

The file-tailoring output data set must be a partitioned data set. Create this data set with the following recommended characteristics:

- DSORG: PO
- RECFM: FB
- LRECL: 80
- BLKSIZE: any multiple of 80

Each user normally wants to have a personal file-tailoring output data set. To do this, use the user ID as one of the qualifiers when allocating the data set name. For example, the data set name could be TRANSACT.WB061.&SYSUID..FTOUTPUT.

[Appendix D, “Sample ISPF Startup CLIST”](#), contains an example to show how a file-tailoring output data set can be allocated in your ISPF startup CLIST. For more information on allocating your file-tailoring output data set, refer to the IBM *ISPF Dialogue Management Guide and Reference* manual.

The List Data Set and Internal Work Files

The following sections explain using the List Data Set and internal work files.

The List Data Set

ISPF Workbench uses a list data set that works in a manner very similar to the ISPF list data set.

You can preallocate this data set prior to invoking your ISPF Workbench for ISPF session, but preallocation is not required. If this data set has not been preallocated, it will dynamically allocate with a disposition of new when the data set is needed.

If this data set is preallocated, it must have the following characteristics:

- DDNAME: M9LIST
- DSORG: PS or SYSOUT
- RECFM: FBA
- LRECL: 133
- BLKSIZE: any multiple of 133

The list data set corresponds to the ISPF list data set in purpose and function. It holds any output that you request while in ISPF Workbench. For example, if you use the utilities to perform source statement retrieval and you request a hard copy of the source, the source listing is written to this data set.

If you preallocate this data set, no termination processing is attempted at the end of the ISPF Workbench session.

If ISPF Workbench allocates this data set, a Process List Data Set panel displays during termination processing. This panel functions in the same manner as the ISPF Process List Data Set panel.

You can set up default processing parameters for this data set using the ISPF Workbench Parameters selection.

If the list data set is dynamically allocated by ISPF Workbench, the naming convention used is &SYSPREF.(&SYSUID.).M9TEMPn.LIST. The &SYSUID qualifier is used only if it differs from the &SYSPREF system prefix.

Internal Work Files

An ISPF Workbench session can use up to five internal work files. These data sets are allocated as needed to the following ddnames: M9LST1, M9LST2, M9LST3, M9LST4, and M5LIST. M9LST3 and M9LST4 are required only when a 3290 terminal is in use. M5LIST is used only by the VISION:Transact development facility.

These data sets are allocated and deleted as necessary and cannot be preallocated.

You can control some of the dynamic allocation parameters used by ISPF Workbench when allocating these data sets by modifying the following two panels in your panel library:

- M9DATPMI — this panel allows you to customize some of the allocation parameters used to allocate the M9LSTn data sets.
- M9DATPMV — this panel allows you to customize some of the allocation parameters used to allocate the M5LIST data set.

These panels allow you to specify unit and space allocations for the internal work files. Just prior to dynamically allocating any of these data sets, for ISPF Workbench retrieves and uses the allocation information from the appropriate panel variables.

Remember that the units specified on these panels must be known to TSO and must be eligible to contain permanent data sets.

The naming convention used by ISPF Workbench for the M9LSTn data sets are &SYSPREF.(&SYSUID.).M9TEMPn.LSTn. For the M5LIST data set, the convention is &SYSPREF.(&SYSUID.).M9TEMPn.M5LIST. In both cases, the &SYSUID qualifier is used only if it differs from the &SYSPREF system prefix.

Invoking ISPF Workbench

ISPF Workbench is designed to support VISION:Builder, VISION:Transact and, VISION:Inform. These three products and the shared component have a commonality of elements and specifications.

There are also separate elements that are unique to each product. ISPF Workbench has a common entry point that selects a subsection that is appropriate for each specific product. Full functionality of each subsection of ISPF Workbench is dependent upon the presence of the separate product software. At least one product must be available for ISPF Workbench to function.

ISPF Workbench runs under IBM's ISPF/PDF facility, which is normally started by executing an ISPF startup CLIST, once you are logged on to TSO. ISPF can also be started from the TSO logon procedure. In either case, the appropriate ISPF data sets are allocated and a menu panel displays for the user to make a selection and activate the services or dialogs you want.

There are two methods for invoking ISPF Workbench (or any ISPF dialog application):

- Adding a selection option to the standard ISPF primary menu panel used at your facility, and then, selecting the designated option to invoke ISPF Workbench. When invoked, an ISPF Workbench entry menu displays. Select the appropriate option for entry to the subsection application (VISION:Builder, VISION:Transact, and so on) you want.
- Creating a CLIST that allocates the appropriate ISPF data sets and uses the ISPSTART command to directly invoke ISPF Workbench. This allows users to invoke ISPF Workbench without going through the standard ISPF primary menu. The standard IBM ISPF options are not available for selection when this method is used.

Both methods are described in the following sections. There are variations of these basic methods to accomplish the same results. Your systems personnel know which method works best for your facility. Remember, whichever method you use, the proper ISPF environment and appropriate ISPF data sets, along with the ISPF Workbench and associated product data sets, must be established and allocated for everything to function properly.

Adding an Option to a Standard ISPF Primary Menu

In [Appendix E, “ISPF Workbench Invocation Panels”](#), the XSR@PRIM panel shows a sample of an ISPF primary menu panel specification. The arrows in the figure point to the specifications that can be added that cause ISPF Workbench to be invoked. You only need to add the specifications that are appropriate for your facility.

You need to add an assignment to the INIT section of the panel, as shown in the following statement:

```
&M9PRODUCT = Workbench
```

This assignment sets an internal variable in ISPF Workbench that is used in some of the panel displays.

To invoke the ISPF Workbench Primary Selection Menu, add the WB specifications to your primary menu. The specifications add an option code (WB) to the panel display (top portion) and an action (PROC) to be taken when the option is selected.

The system displays the following line:

```
% WB +WORKBENCH -%VISION:Workbench Facility - Release 6.1
```

The percent sign (%) means that the following text is highlighted. The plus sign (+) specifies that the following text be in normal intensity. These are standard ISPF attribute indicators. The text and characters can be almost anything you choose; they are simply displayed.

The following line causes ISPF to display the panel M9PRIM when the option WB is entered on the ISPF primary menu option line:

```
WB,'PANEL(M9PRIM)' /* Invokes VISION:Workbench Selection Menu */
```

The line must be keyed in uppercase. The /**/ is just a comment and can be ignored or used as a reference. The ISPF WB action causes a display of the M9PRIM panel and a transfer control to the actions specified on that panel from subsequent user interaction. The M9PRIM panel is the primary selection menu for ISPF Workbench. M9PRIM shows the M9PRIM panel specifications.

From the ISPF Workbench primary selection menu, enter the appropriate option to invoke the subsection you want.

You can choose to bypass the ISPF Workbench primary selection menu and go directly to the subsection you want by adding some or all of the remaining specifications shown by the arrows in XSR@PRIM.

The system displays the following lines:

```
1 - BL (M4) Workbench for VISION:Builder 14.0
2 - TR (M5) Workbench for VISION:Transact 7.8
3 - IN (DA) Workbench for VISION:Inform 4.0 (Definition Processor)
```

The purpose is to instruct you to use the BL, TR, and IN characters as an option on the option line and cause the action specified in the PROC section of the panel coding.

The following PROC section lines in TRANSACT.WB061.PANELS(XSR@PRIM) cause ISPF to transfer control to the program M9BOOT when the option BL, TR, or IN is entered on the ISPF primary menu option line:

```
BL, 'PGM(M9BOOT)  PARM(BDM4)  NOCHECK'  
TR, 'PGM(M9BOOT)  PARM(ODM5)  NOCHECK'  
IN, 'PGM(M9BOOT)  PARM(PMM4)  NOCHECK'
```

The line must be keyed in uppercase. The PARM values are passed to the M9BOOT program and cause the appropriate subsection selection menu to be displayed.

You need to add one final specification to the PROC section of the panel, as shown in the following statement:

```
&GVNXTSEL = .TRAIL
```

This assignment sets an internal variable in ISPF Workbench such that any trailing command options are available for processing.

Only use the options and actions for the products that are appropriate for your facility.

If you do not have all the companion product software, see [Using Other ISPF Workbench Subsections on page 2-34](#) for information on how to use the other portions of ISPF Workbench without having the companion product software.

The above specifications for panels are standard coding as provided by IBM for their ISPF/PDF environment. Check with your systems department if you are not sure of the best method for your facility.

Using the ISPSTART Command

You can invoke ISPF Workbench directly from TSO using the ISPSTART command. You can write a CLIST that allocates all ISPF data sets and the ISPF Workbench data sets, and then executes the ISPSTART command to invoke ISPF Workbench directly. The CLIST is almost identical to the standard ISPF startup CLIST. By using this method, you bypass the standard ISPF primary menu and the options normally available to ISPF from the primary menu cannot be accessed.

[Appendix D, “Sample ISPF Startup CLIST”](#), shows a sample ISPF startup CLIST. At the end of the CLIST is the ISPSTART command. You can change this command to invoke the ISPF Workbench selection menu (M9PRIM) or the entry program (M9BOOT) directly.

To invoke the Workbench selection menu, use the following statement:

```
ISPSTART PANEL(M9PRIM)
```

To invoke an ISPF Workbench subsection menu, use the following statement:

```
ISPSTART PGM(M9B00T) PARM(XXXX)
```

where: XXXX identifies the product subsection, as follows:

- BDM4 — VISION:Builder 14.0 Workbench
- ODM5 — VISION:Transact 7.8 Workbench
- PMM4 — VISION:Inform 4.0 Workbench

Note: Only one subsection can be started per TSO session.

These specifications for CLISTs are standard coding as provided by IBM for their TSO environment. Check with your systems department if you are not sure of the best method for your facility.

Using Other ISPF Workbench Subsections

ISPF Workbench is designed to be fully-functional when the companion product software is also installed and available to ISPF. However, ISPF Workbench can also be used when the companion product software is not installed. One of the software products must be available so that, as a minimum, the COMLIB component is present and available to ISPF Workbench.

If your facility already has all the companion product software, VISION:Builder, VISION:Transact, VISION:Inform, or if you are not interested in exploring the other portions of ISPF Workbench, then you can skip this section.

When the associated software product is not available, application validation does not function, but the data entry edits are still active. This gives users a chance to explore the other portions of ISPF Workbench.

Change the PARM(XXXX) in the following locations when the associated product software is not installed and available:

- Panel M9PRIM (described in [Appendix E, “ISPF Workbench Invocation Panels”](#))
- The tailored ISPF primary menu panel
- The ISPSTART command specifications

The following table shows the before and after changes for the PARM entry.

Before	After	Comment
PARM (BDM4)	PARM (BD)	replace the M4 with two blank spaces to indicate that VISION:Builder is not available
PARM (ODM5)	PARM (OD)	replace the M5 with two blank spaces to indicate that VISION:Transact is not available
PARM (PMM4)	PARM (PM)	replace the M4 with two blank spaces to indicate that VISION:Inform is not available

After you have made these changes, perform the following steps to ensure that a product parameters module is available to make the other portions of ISPF Workbench function:

- Use the two default parameter modules in the ISPF Workbench load library (TRANSACT.WB061.WBTLOAD).
- Rename these modules, depending on which other portion you want to be functional.
 - For VISION:Builder and VISION:Inform, rename the M4PRMMOD to M4PARAMS.
 - For VISION:Transact, rename the M5PRMMOD to FIVEPARM.

Once the appropriate specifications have been adjusted, ISPF Workbench will function in the subsections even if the companion software product is not installed. Remember, the application validation will not function, but the data entry edits are still active.

Optional Setup for ISPF Workbench

ISPF offers a panel preprocessing utility called ISPPREP to convert your ISPF Workbench panels into an encoded format that significantly improves panel display performance while in ISPF Workbench. A preprocessed panel library also takes up to 20 percent less space than an unprocessed panel library.

Preprocessing Your Panel Library

After a panel is preprocessed and is in an encoded display format, it cannot be modified. To change a preprocessed panel, you must modify the original panel source member and rerun ISPPREP for that panel.

Not all panels can be preprocessed. There are restrictions that prevent ISPPREP from successfully converting certain panels.

The following ISPF Workbench panels are bypassed automatically when ISPPREP is run:

\$\$COPYRT	M9HCAPBR	M9HCAPDA	M9HCAPED
M9HCAPPF	M9HCAPSF	M9HCAPSS	M9SVAPBR
M9SVAPPM	M9TBAPT	M9TBAPT	

If you want to preprocess your ISPF Workbench panels, allocate a second panel library to hold the preprocessed panels. Leave your original panel source library unchanged.

To preprocess your panel library, follow the directions in the next six sections. If you do not want to preprocess your panel library, proceed to [Customizing Job Submission Skeletons on page 2-38](#).

Allocating the ISPF Log Data Set

Ensure that you have an ISPF log data set allocated. The preprocess utility writes information messages to this data set.

Allocating a New Panel Library

Allocate this panel library with the same characteristics as your ISPF Workbench panel source library. The space allocation can be reduced to 80 primary tracks. The directory blocks can be reduced to 80 blocks unless you set the Save Statistics option to Yes on the ISPPREP panel, in which case you must increase the directory blocks to 240.

Running the Preprocess Utility

To run the preprocess utility, go into ISPF and select the Command option (option 6) from the primary menu. Enter the command ISPPREP on the TSO command line.

A preprocess utility panel will display. This panel allows you to specify the input data set of non-processed panels and the output data set where the pre-processed panels will be stored. Fill in the appropriate information to convert all panels and press the Enter key.

The time required to complete the panel conversion process varies from installation to installation. At the Computer Associates installation, ISPPREP takes about eight minutes to complete. Informational messages appear during this time to tell you how many panels have been processed.

You need not worry about the panels that cannot be preprocessed at this time. The ISPPREP utility will recognize that these panels cannot be encoded and will automatically skip them.

When this process is finished, view or print your ISPF log. You can view the log using ISPF option 7.5 Dialog Test, Log option. The log contains informational messages from the conversion process that pertain to the panels that could not be converted.

Copying the Unprocessed Panels

After the conversion process is complete, use the ISPF Copy utility (option 3.3) to copy the non-preprocessed panels listed below from your ISPF Workbench input data set to your new preprocessed panel library.

M9HCAPBR	M9HCAPDA	M9HCAPEB	M9HCAPPF	M9HCAPSF
M9HCAPSS	M9SVAPBR	M9SVAPPM	M9TBAPTB	M9TBAPTP

Note: Member \$\$COPYRT does not need to be copied.

Allocating the New Library to ISPLIB

The preprocessed panel library now contains all of your ISPF Workbench panels. Allocate this library to the ISPF ddname ISPLIB, so that ISPF will use the encoded ISPF Workbench panels rather than the source versions. Your ISPF Workbench panel library containing the non-preprocessed panels can be removed from the ISPLIB concatenation.

Modifying Panels After Preprocessing

If you change an ISPF Workbench panel, it must be edited in its source format. After the modification is complete, replace it in the preprocessed panel library by running the panel through the ISPPREP utility.

Customizing Job Submission Skeletons

There are four default file-tailoring skeletons, accompanied by four default user panels (see [Appendix E, “Skeleton and User Panel Listings”](#) for more information). These are used by the ISPF Workbench generate subsystems for VISION:Builder and VISION:Transact. These skeletons and panels are as follows:

- Panel M9BGUPNL and Skeleton M9BGTS
This panel/skeleton combination submits VISION:Builder background jobs.
- Panel M9FGUPNL and Skeleton M9FGTS
This panel/skeleton combination executes VISION:Builder foreground jobs.
- Panel M9GCTPU2 and Skeleton M9GCTSBG
This panel/skeleton combination submits VISION:Transact background jobs.
- Panel M9GCTPU1 and Skeleton M9GCTSFG
This panel/skeleton combination executes VISION:Transact foreground jobs.

As a safety measure, create a backup before you modify any of the panels or skeletons.

If you are currently using ISPF Workbench job submission skeletons from a previous release, you can continue to use them with this release. You should change data set names to reflect the new release library names, but no other modifications are necessary.

If you do not have working skeletons from a previous ISPF Workbench release, use the skeletons in this release as a starting point and modify them so that they work properly for your installation.

The sample file-tailoring skeletons distributed with the system refer to variables from the corresponding sample user panels. To use a sample skeleton, you must specify during job generation that the appropriate sample user panel be displayed. After it displays, complete all the entries.

The use of user panels is not mandatory. While this feature makes the system more flexible, you can bypass the user panel, if you want. Instead, just hard code the user panel information in your job submission skeletons.

Before you actually start customizing your ISPF Workbench file-tailoring skeletons, run a few job submission tests using the default versions of the skeletons and user panels.

When making these test runs, specify a processing option of "Keep" on the file-tailoring option panel. This causes your submission JCL or CLIST to be written to the file tailoring output data set, but the job does not execute. By reviewing the generated JCL and CLISTs, you will become better acquainted with how the file tailoring skeletons work. You will also be able to see exactly where you need to make changes to conform to your installation standards.

For more information on file tailoring and file-tailoring skeletons, refer to the IBM *ISPF Dialogue Management Guide and Reference* manual.

Step 6 – Setting up the File Definition Conversion Utilities (Optional)

Note: Refer to the *Advantage VISION:Transact Getting Started* guide for information about File Definition Conversion (quick start) utilities.

The VISION:Transact System contains four utilities to quick start the user in developing file definitions. These utilities convert existing COBOL, DB2, VISION:Inquiry, and VISION:Results table and file definitions into a VISION:Transact file definition that is then tailored for use. The WORKLIB data set contains JCL to help in the setup and execution of these utilities.

If you plan to use the DB2 quick start utility, you must BIND the utility first. Use the JCL member QSDB2BND in the WORKLIB data set to perform the BIND as a batch job. The DB2 quick start utility has already been preprocessed and prepared. The DBRM is delivered in the WORKLIB data set. Once the BIND is completed, the utility is ready for use.

The COBOL, VISION:Inquiry, and VISION:Results quick start utilities do not require any setup, but you can link edit interfaces with the utility for access to source code management libraries such as CA-Panvalet and CA-Librarian. Sample JCL (QSCOBLL, QSCOBPL, QSRESLL, and QSRESPL) is provided in the WORKLIB data set for the optional link edits.

The JCL members in the WORKLIB dataset for running the utilities are: QSCOBEX, QSDB2EX, QSINQEX, and QSRESEX.

Maintenance and Support

During the life of a VISION:Transact release, PTFs, formerly known as System Modifications (SMs), are developed to enhance, maintain, and customize the product and components. Any problems that arise are fixed by PTFs (SMs), which are numbered in sequence as they are developed for each release, beginning with 200. There are other patches called APARs, formerly known as Restricted System Modifications (RSMs), that are special customizations to the product and do not apply to all sites. The user should always review the APAR description carefully before applying them to a system.

The PTFs and APARs are identified by component and number using the following format:

CCNNNNN

where:

CC is the Component Identifier:

TR VISION:Transact Generator and Executor components

CL COMLIB sub-component of the Generator component

WB ISPF Workbench component

NNNNN is the Modification Number Identifier:

Examples: TR00200, TR00125, CL00215, WB00201.

Maintenance – Installing the PTFs and APARs

All PTFs and APARs are installed to VISION:Transact and its components under the control of SMP/E.

The SMP/E process for handling PTFs and APARs has the following basic steps:

1. Record and save the PTF or APAR into the global zone using the RECEIVE command.
2. Use the APPLY command to install the PTF or APAR to the target libraries.
3. Use the ACCEPT command to install the PTF or APAR into the distribution libraries.

The PTFs are general modifications that are designed for all users and all systems. These should always be installed into VISION:Builder and its components in order to keep the system up to date. PTFs should be installed in both the Target and Distribution Libraries.

The APARs are special modifications that are designed for unique situations. The APARs do not apply to all users and systems. The control statements in the SMPCNTL data set contain comments for each item that describes the situation addressed by the PTF or APAR. Review the description of any APAR you are considering for your system. Contact Computer Associates Technical Support if you have any questions, concerns, or if you just need more information regarding an APAR. See [Contacting Computer Associates on page 1-8](#).

When installing APARs, there may be some time between the APPLY to the target libraries and the ACCEPT to the distribution libraries. You should take this time to evaluate whether the APAR satisfies the special need for your system. If you decide that the APAR is not appropriate, you can use an SMP/E RESTORE command to remove the APAR from the target libraries. Additionally, you can use an SMP/E REJECT command to remove the APAR for the global zone.

Note: Once you ACCEPT an element, such as APAR or PTF, into the distribution libraries, there is no direct method for restoring the previous version of an element in your target libraries.

The SMPCNTL data set contains some model jobs for performing the various maintenance activities described above. Here are the member names and their functions:

Member	Names Description
ZRECEIVE	RECEIVE a PTF or APAR into the Global Zone and Libraries
ZAPPLY	APPLY a PTF or APAR into the Target Libraries
ZACCEPT	ACCEPT a PTF or APAR into the Distribution Libraries

Member	Names Description
ZRESTORE	RESTORE (remove) a PTF or APAR from the Target Libraries
ZREJECT	REJECT (remove) a PTF or APAR from the Global Zone and Libraries

There are other Tools and Facilities that are available for invoking SMP/E commands and functions. Any of these will work because VISION:Transact only uses the standard SMP/E processes. The Systems Group at each site has their favorite tools and procedures, and any of those should work just fine.

Note: The APAR runs get a return code of 4 from APPLY and ACCEPT runs because they do not contain prerequisites for other PTFs and APARs. When PTF runs are performed after an APAR has been processed, they get a return code of 4 because the PTF will not contain prerequisites for any APARs. Remember, the APARs can be identified by their number, which is in the range of cc00001 to cc00199, with the cc being a component identifier.

Support – Problem Reporting

When a problem is encountered, contact Computer Associates Technical Support to have a representative review your situation. You must provide details to the representative concerning what actions were being performed at the time the problem occurred. Any information on recreating the problem is very useful. Capture any messages or information displayed and communicate these messages to the support representative.

See [Contacting Computer Associates on page 1-8](#).

VISION:Transact Generator Problem Reporting

For VISION:Transact Generator problems, diagnostic information may display as part of the termination handling. Message MK5CG01 or MK5CG02 starts the display. Save the information on this display so that it can be sent to the Technical Support representative. In some cases, you may need to take a complete SYSUDUMP at the time the problem occurs to help determine the reason for the error.

VISION:Transact Executor Problem Reporting

For VISION:Transact Executor problems, a diagnostic error screen may be displayed. Save the information on this screen so that it can be sent to the Technical Support representative. In some cases, you may need to capture a CICS Transaction Dump to help determine the reason for the error. If execution tracing was active at the time the problem occurred, a copy of the execution trace messages where possible should also be forwarded to the Technical Support representative to help diagnose the problem.

ISPF Workbench Problem Reporting

For ISPF Workbench problems, gather as much of the following information as possible:

- The objective of your session.
- The name, identification, or description of the last valid panel you saw before the problem.
- Any error messages that were displayed.
- Information from the unexpected error display, if applicable.
- Any other information you feel would be useful in recreating the situation.

Panel Identification

The upper-left corner of every ISPF Workbench data entry panel contains a panel identification name. This name is used to reference the panels in ISPF Workbench documentation. This identification name is not the name of the panel member in your panel library.

To view the actual member name of a panel, type `PANELID` on the ISPF command line. To turn this feature off and return to the panel identification name, type `PANELID` again. The `PANELID` command toggles between on and off.

Unexpected Error Panel

If ISPF Workbench abnormally terminates or detects a serious internal error, it displays an Unexpected Error panel. This panel contains information that is always useful when trying to track down the cause of the problem.

If you encounter this screen, obtain a screen print of the display before continuing. If you cannot obtain a screen print, record the following information:

- Error messages on the display.
- PSW value.
- The contents of registers 12, 14, and 15.

Use the Help Primary Command to display any message that might be pending.

Problems and User Code

The extensive error checking and data validation techniques built into VISION:Transact and its components ensure processing integrity. However, these systems have no capacity for determining the integrity of user code incorporated into the application through external function calls. Consequently, an error in such user-provided code could cause system failure.

Determining the true cause of errors within user code is not easy, can be time consuming, and could significantly increase the cost of maintaining these systems. Computer Associates works with you, the customer, to help discover where the problem might exist in the user code. After determining the nature of the problem in the user code, it is up to you to make the corrections.

In some cases, if it becomes apparent that the amount of effort needed from Computer Associates to resolve the user code problem places an unreasonable burden on resources, written authorization may be needed to charge you time and materials rates for the expense involved in analyzing the problem. This is handled on a case-by-case basis.

SMP/E Install JCL Samples

This appendix lists the JCL member in File 1 of the Installation Tape, TRANSACT.SMPE.SMPCNTL. These jobs are used to define the product to SMP/E and apply maintenance to it. A brief description of the JCL members appears below, followed by the listing of the jobs.

Note: There are other non-JCL members in the SMPCNTL data set. These are input data files to the SMP/E applications and should only be modified with extreme caution. Additionally, the data set contains APARs and PTFs that may be applied to the product. The APARs should be reviewed to see if they apply to your installation. The PTFs should always be applied to your installation of the product.

Member	Description
COPYTAPE	Copy the installation tape files to disk
SMPJOB01	Allocate the VISION:Transact SMP/E data sets and initialize the CSI
SMPJOB02	Define VISION:Transact to the Global, Target, and Distribution zones
SMPJOB03	RECEIVE the VISION:Transact program element SYSMODS into the Global zone
SMPJOB04	APPLY the VISION:Transact program element SYSMODS into the Target zone
SMPJOB05	ACCEPT the VISION:Transact program element SYSMODS into the Distribution zone
ZACCEPT	ACCEPT any APAR and PTF SYSMODS into the Distribution zone
ZAPPLY	APPLY any APAR and PTF SYSMODS into the Target zone
ZRECEIVE	RECEIVE any APAR and PTF SYSMODS into the Global zone

Member	Description
ZREJECT	REJECT any APAR and PTF SYSMODS that were received into the Global zone
ZRESTORE	RESTORE (remove) any APAR and PTF SYSMODS that were applied to the Target zone

COPYTAPE

```

//          JOB ...
//*
//* MEMBER COPYTAPE
//*****
//*
//* COPY INDIRECT LIBRARIES FOR SMP/E FROM TAPE TO DISK
//*
//*****
//*
//* REPLACE THE JOB CARD ABOVE.
//*
//* CHANGE THE SET PARAMETERS BELOW TO REFLECT YOUR DATASET
//* NAMING AND ALLOCATION CONVENTIONS.  THE SETTINGS USED HERE
//* WILL ALSO NEED TO BE USED IN THE SUBSEQUENT SMP/E INSTALL JOBS.
//*
//*****
//*
// SET MARKV=TRANSACTION.TR078  HIGH-LEVEL QUALIFIER FOR TRANSACTION DSN'S
// SET BENCH=TRANSACTION.WB061  HIGH-LEVEL QUALIFIER FOR WORKBENCH DSN'S
//*
//*****
//* CHANGE THE DUNIT, DVOLSER, TUNIT, AND TVOLSER PARAMETERS IN THE
//* INSTREAM PROC BELOW AS APPROPRIATE FOR YOUR INSTALLATION.
//*
//*****
//*
//COPY      PROC DISKDSN=,
//          DUNIT=SYSDA,
//          DVOLSER=?????,          <=====
//          DTRKS=,
//          DDCB=,
//          TUNIT=3480,
//          TVOLSER=?????,          <=====
//          TFILENO=
//*
//DELETE EXEC PGM=IEFBRI4
//DD1      DD DSN=&DISKDSN,DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),UNIT=&DUNIT
//*
//COPY EXEC PGM=IEBCOPY,REGION=2M
//SYSPRINT DD SYSOUT=*
//IN      DD DSN=VISION.TRANSACTION.FILE&TFILENO,DISP=OLD,
//          UNIT=&TUNIT,LABEL=(&TFILENO,SL,EXPDT=98000),
//          VOL=(PRIVATE,RETAIN,SER=(&TVOLSER))
//OUT     DD DSN=&DISKDSN,DISP=(NEW,CATLG),
//          UNIT=&DUNIT,VOL=SER=&DVOLSER,
//          SPACE=(TRK,&DTRKS),
//          DCB=&DDCB
//SYSUT3   DD UNIT=SYSDA,SPACE=(CYL,1)
//SYSUT4   DD UNIT=SYSDA,SPACE=(CYL,1)
//*
//          PEND
//*
//*****

```

Figure A-1 COPYTAPE — Copy the installation tape files to disk (Page 1 of 4)

Figure A-1 COPYTAPE — Copy the installation tape files to disk (Page 2 of 4)

```

//          DDCB='(RECFM=U,LRECL=0,BLKSIZE=27998)',
//          TFILENO=6
//COPY.SYSIN DD *
//          COPYMOD INDD=IN,OUTDD=OUT
//*
//*****
//* FILE7 - COPY THE IMS TOOLKIT LOAD LIBRARY TO DISK
//*****
//FILE7 EXEC COPY,
//          DISKDSN=&MARKV..IMSTKL,
//          DTRKS='(15,5,10)',
//          DDCB='(RECFM=U,LRECL=0,BLKSIZE=27998)',
//          TFILENO=7
//COPY.SYSIN DD *
//          COPYMOD INDD=IN,OUTDD=OUT
//*
//*****
//* FILE8 - COPY THE CICS TOOLKIT SOURCE LIBRARY TO DISK
//*****
//FILE8 EXEC COPY,
//          DISKDSN=&MARKV..CICSTKS,
//          DTRKS='(15,5,15)',
//          DDCB='(RECFM=FB,LRECL=80,BLKSIZE=27920)',
//          TFILENO=8
//COPY.SYSIN DD *
//          COPY INDD=IN,OUTDD=OUT
//*
//*****
//* FILE9 - COPY THE IMS TOOLKIT SOURCE LIBRARY TO DISK
//*****
//FILE9 EXEC COPY,
//          DISKDSN=&MARKV..IMSTKS,
//          DTRKS='(15,5,15)',
//          DDCB='(RECFM=FB,LRECL=80,BLKSIZE=27920)',
//          TFILENO=9
//COPY.SYSIN DD *
//          COPY INDD=IN,OUTDD=OUT
//*
//*****
//* FILE10 - COPY THE ISPF WORKBENCH LOAD LIBRARY TO DISK
//*****
//FILE10 EXEC COPY,
//          DISKDSN=&BENCH..LOADLIB,
//          DTRKS='(60,30,40)',
//          DDCB='(RECFM=U,LRECL=0,BLKSIZE=27998)',
//          TFILENO=10
//COPY.SYSIN DD *
//          COPYMOD INDD=IN,OUTDD=OUT
//*
//*****
//* FILE11 - COPY THE ISPF WORKBENCH PANEL LIBRARY TO DISK
//*****
//FILE11 EXEC COPY,
//          DISKDSN=&BENCH..PANELS,
//          DTRKS='(120,30,250)',
//          DDCB='(RECFM=FB,LRECL=80,BLKSIZE=27920)',
//          TFILENO=11
//COPY.SYSIN DD *
//          COPY INDD=IN,OUTDD=OUT
//*
//*****
//* FILE12 - COPY THE ISPF WORKBENCH MESSAGE LIBRARY TO DISK
//*****
//FILE12 EXEC COPY,
//          DISKDSN=&BENCH..MSGs,
//          DTRKS='(15,5,40)',
//          DDCB='(RECFM=FB,LRECL=80,BLKSIZE=27920)',
//          TFILENO=12
//COPY.SYSIN DD *
//          COPY INDD=IN,OUTDD=OUT

```

Figure A-1 COPYTAPE — Copy the installation tape files to disk (Page 3 of 4)

```
//*
//*****
//*
//* FILE13 - COPY THE ISPF WORKBENCH SKELETON LIBRARY TO DISK
//*****
//FILE13 EXEC COPY,
//      DISKDSN=&BENCH..SKELS,
//      DTRKS=' (10,5,5) ',
//      DDCB=' (RECFM=FB,LRECL=80,BLKSIZE=27920) ',
//      TFILENO=13
//COPY.SYSIN DD *
//      COPY INDD=IN,OUTDD=OUT
//*
//*****
//*
//* FILE14 - COPY THE ISPF WORKBENCH CLIST LIBRARY TO DISK
//*****
//FILE14 EXEC COPY,
//      DISKDSN=&BENCH..CLIST,
//      DTRKS=' (20,5,10) ',
//      DDCB=' (RECFM=FB,LRECL=80,BLKSIZE=27920) ',
//      TFILENO=14
//COPY.SYSIN DD *
//      COPY INDD=IN,OUTDD=OUT
//*
//
```

Figure A-1 COPYTAPE — Copy the installation tape files to disk (Page 4 of 4)

SMPJOB01

```

//          JOB ...
//*
//* MEMBER SMPJOB01
//*****
//*
//* ALLOCATE ALL VISION:TRANSACTION SMP/E DATASETS AND INITIALIZE THE CSI
//*
//*****
//*
//* REPLACE THE JOB CARD ABOVE.
//*
//* CHANGE THE SET COMMAND PARAMETERS BELOW TO REFLECT YOUR DATASET
//* NAMING AND ALLOCATION CONVENTIONS.
//*
//* BE SURE TO CHANGE THE CSI CLUSTER NAME IN THE IDCAMS DEFINE STEP
//* NEAR THE END OF THIS JCL TO MATCH THE VALUE ASSIGNED TO THE SMPE
//* SET VARIABLE. THE SET COMMAND WILL NOT SUBSTITUTE THE NAME INTO
//* THE IDCAMS CONTROL STATEMENTS.
//*
//* IF AN EXISTING SMP/E CSI IS USED, THE LAST 2 STEPS CAN BE REMOVED.
//*
//*****
// SET SMPE=TRANSACTION.SMPE    HIGH-LEVEL QUALIFIERS FOR SMP/E CSI DSN
// SET MARKV=TRANSACTION.TR078  HIGH-LEVEL QUALIFIERS FOR TRANSACTION'S
// SET BENCH=TRANSACTION.WB061  HIGH-LEVEL QUALIFIERS FOR WORKBENCH DSN'S
// SET TUNIT=ISPDA              UNIT FOR TARGET LIBRARIES
// SET DUNIT=ISPDA              UNIT FOR DISTRIBUTION LIBRARIES
// SET SUNIT=ISPDA              UNIT FOR SMP/E LIBRARIES
// SET TVOL=ISP804              VOLUME FOR TARGET LIBRARIES
// SET DVOL=ISP805              VOLUME FOR DISTRIBUTION LIBRARIES
// SET SVOL=ISP806              VOLUME FOR SMP/E LIBRARIES
//*
//*****
//*
//* ALLOCATE VISION:TRANSACTION DISTRIBUTION LIBRARIES
//*
//*****
//DELD    EXEC PGM=IEFBR14
//DD1G    DD DSN=&MARKV..TRDGEN,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD1C    DD DSN=&MARKV..TRDCICS,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD1I    DD DSN=&MARKV..TRDIMS,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD2     DD DSN=&MARKV..TRDWORK,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD3C    DD DSN=&MARKV..TRDTKCS,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD3I    DD DSN=&MARKV..TRDTKIS,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD4C    DD DSN=&MARKV..TRDTKCL,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD4I    DD DSN=&MARKV..TRDTKIL,
//          DISP=(MOD,DELETE),

```

Figure A-2 SMPJOB01 — Allocate the VISION:Transaction SMP/E data sets and initialize the CSI (Page 1 of 5)

```

//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD5      DD DSN=&BENCH..WBDLOAD,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD6      DD DSN=&BENCH..WBDPANEL,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD7      DD DSN=&BENCH..WBDMGS,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD8      DD DSN=&BENCH..WBDSKELS,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//DD9      DD DSN=&BENCH..WBDCLIST,
//          DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),
//          UNIT=&DUNIT
//*
//ALLOCD   EXEC PGM=IEFBR14
//TRDGEN   DD DSN=&MARKV..TRDGEN,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(12,3,40))
//TRDCICS  DD DSN=&MARKV..TRDCICS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,5))
//TRDIMS   DD DSN=&MARKV..TRDIMS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,5))
//TRDWORK  DD DSN=&MARKV..TRDWORK,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(6,2,10)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=0)
//TRDTKCS  DD DSN=&MARKV..TRDTKCS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,15))
//TRDTKCL  DD DSN=&MARKV..TRDTKCL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,10))
//TRDTKIS  DD DSN=&MARKV..TRDTKIS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,15))
//TRDTKIL  DD DSN=&MARKV..TRDTKIL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,10))
//WBDLOAD  DD DSN=&BENCH..WBDLOAD,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(4,1,40)),DCB=(RECFM=U,BLKSIZE=27998)
//WBDPANEL DD DSN=&BENCH..WBDPANEL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(6,2,250))
//WBDMGS   DD DSN=&BENCH..WBDMGS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,40))
//WBDSKELS DD DSN=&BENCH..WBDSKELS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(1,1,5))
//WBDCLIST DD DSN=&BENCH..WBDCLIST,
//          DISP=(NEW,CATLG,DELETE),

```

Figure A-2 SMPJOB01 — Allocate the VISION:Transact SMP/E data sets and initialize the CSI (Page 2 of 5)

```

//          UNIT=&DUNIT,VOL=SER=&DVOL,
//          SPACE=(CYL,(2,1,10))
//*
//*****
//*
//* ALLOCATE VISION:TRANSACT TARGET LIBRARIES
//*
//*****
//*
//DELT      EXEC PGM=IEFBR14
//DDL1G     DD DSN=&MARKV..TRTGEN,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DDL1C     DD DSN=&MARKV..TRTCICS,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DDL1I     DD DSN=&MARKV..TRTIMS,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD2       DD DSN=&MARKV..TRTWORK,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD3C      DD DSN=&MARKV..TRTTKCS,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD3I      DD DSN=&MARKV..TRTTKIS,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD4C      DD DSN=&MARKV..TRTTKCL,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD4I      DD DSN=&MARKV..TRTTKIL,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD5       DD DSN=&BENCH..WBTLOAD,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD6       DD DSN=&BENCH..WBTPANEL,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD7       DD DSN=&BENCH..WBMSGSGS,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD8       DD DSN=&BENCH..WBTSKELS,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//DD9       DD DSN=&BENCH..WBTCLIST,
//           DISP=(MOD,DELETE),
//           SPACE=(TRK,(0,0)),
//           UNIT=&TUNIT
//*
//ALLOCT    EXEC PGM=IEFBR14
//TRTGEN     DD DSN=&MARKV..TRTGEN,
//           DISP=(NEW,CATLG,DELETE),
//           UNIT=&TUNIT,VOL=SER=&TVOL,
//           SPACE=(CYL,(12,3,40))
//TRTCICS    DD DSN=&MARKV..TRTCICS,
//           DISP=(NEW,CATLG,DELETE),
//           UNIT=&TUNIT,VOL=SER=&TVOL,
//           SPACE=(CYL,(1,1,5))
//TRTIMS     DD DSN=&MARKV..TRTIMS,
//           DISP=(NEW,CATLG,DELETE),
//           UNIT=&TUNIT,VOL=SER=&TVOL,

```

Figure A-2 SMPJOB01 — Allocate the VISION:Transact SMP/E data sets and initialize the CSI (Page 3 of 5)

```

//      SPACE=(CYL,(1,1,5))
//TRTWORK DD DSN=&MARKV..TRTWORK,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(6,2,10)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=0)
//TRTTKCS DD DSN=&MARKV..TRTTKCS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(1,1,15))
//TRTTKCL DD DSN=&MARKV..TRTTKCL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(1,1,10))
//TRTTKIS DD DSN=&MARKV..TRTTKIS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(1,1,15))
//TRTTKIL DD DSN=&MARKV..TRTTKIL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(1,1,10))
//WBTLOAD DD DSN=&BENCH..WBTLOAD,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(4,1,40)),DCB=(RECFM=U,BLKSIZE=27998)
//WBTPANEL DD DSN=&BENCH..WBTPANEL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(8,2,250))
//WBTMSGSGS DD DSN=&BENCH..WBTMSGSGS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(1,1,40))
//WBTSKELS DD DSN=&BENCH..WBTSKELS,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(1,1,5))
//WBTCLIST DD DSN=&BENCH..WBTCLIST,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&TUNIT,VOL=SER=&TVOL,
//          SPACE=(CYL,(2,1,10))
//*
//*****
//*
//* ALLOCATE SMP/E LIBRARIES
//*
//*****
//*
//DELS EXEC PGM=IEFBR14
//DD1 DD DSN=&SMPE..SMPMTS,
//      DISP=(MOD,DELETE),
//      SPACE=(TRK,(0,0)),
//      UNIT=&SUNIT
//DD2 DD DSN=&SMPE..SMPSCDS,
//      DISP=(MOD,DELETE),
//      SPACE=(TRK,(0,0)),
//      UNIT=&SUNIT
//DD3 DD DSN=&SMPE..SMPSTS,
//      DISP=(MOD,DELETE),
//      SPACE=(TRK,(0,0)),
//      UNIT=&SUNIT
//DD4 DD DSN=&SMPE..SMPLOG,
//      DISP=(MOD,DELETE),
//      SPACE=(TRK,(0,0)),
//      UNIT=&SUNIT
//DD5 DD DSN=&SMPE..SMPLOGA,
//      DISP=(MOD,DELETE),
//      SPACE=(TRK,(0,0)),
//      UNIT=&SUNIT
//DD6 DD DSN=&SMPE..SMPPTS,
//      DISP=(MOD,DELETE),
//      SPACE=(TRK,(0,0)),
//      UNIT=&SUNIT
//*

```

Figure A-2 SMPJOB01 — Allocate the VISION:Transact SMP/E data sets and initialize the CSI (Page 4 of 5)

```

//ALLOCS EXEC PGM=IEFBRI4
//SMPMTS DD DSN=&SMPE..SMPMTS,
//        DISP=(NEW,CATLG,DELETE),
//        UNIT=&SUNIT,VOL=SER=&SVOL,
//        SPACE=(CYL,(2,1,50)),
//        DCB=(RECFM=FB,LRECL=80)
//SMPSCDS DD DSN=&SMPE..SMPSCDS,
//        DISP=(NEW,CATLG,DELETE),
//        UNIT=&SUNIT,VOL=SER=&SVOL,
//        SPACE=(CYL,(2,1,50)),
//        DCB=(RECFM=FB,LRECL=80)
//SMPSTS DD DSN=&SMPE..SMPSTS,
//        DISP=(NEW,CATLG,DELETE),
//        UNIT=&SUNIT,VOL=SER=&SVOL,
//        SPACE=(CYL,(2,1,50)),
//        DCB=(RECFM=FB,LRECL=80)
//SMPLOG DD DSN=&SMPE..SMPLOG,
//        DISP=(NEW,CATLG,DELETE),
//        UNIT=&SUNIT,VOL=SER=&SVOL,
//        SPACE=(CYL,(5,2)),
//        DCB=(RECFM=VB,LRECL=510,BLKSIZE=27900)
//SMPLOGA DD DSN=&SMPE..SMPLOGA,
//        DISP=(NEW,CATLG,DELETE),
//        UNIT=&SUNIT,VOL=SER=&SVOL,
//        SPACE=(CYL,(5,2)),
//        DCB=(RECFM=VB,LRECL=510,BLKSIZE=27900)
//SMPPTS DD DSN=&SMPE..SMPPTS,
//        DISP=(NEW,CATLG,DELETE),
//        DSNTYPE=LIBRARY,
//        UNIT=&SUNIT,VOL=SER=&SVOL,
//        SPACE=(CYL,(5,5,50)),
//        DCB=(RECFM=FB,LRECL=80)
//*
//*****
//*
//* ALLOCATE SMP/E CSI FOR VISION:TRANSACT
//*
//*****
//DEFINE EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE (TRANSACT.SMPE.CSI) CLUSTER
SET MAXCC=0
DEFINE CLUSTER (
    NAME (TRANSACT.SMPE.CSI)
    FREESPACE (10,5)
    KEYS (24 0)
    RECORDSIZE (24 143)
    SHAREOPTIONS (2 3)
)
DATA (
    NAME (TRANSACT.SMPE.CSI.DATA)
    CONTROLINTERVALSIZE (4096)
    CYLINDERS (10 5)
)
INDEX (
    NAME (TRANSACT.SMPE.CSI.INDEX)
    CYLINDERS (2 2)
    IMBED
)
//*****
//*
//* INITIALIZE SMP/E CSI FOR VISION:TRANSACT
//*
//*****
//INIT EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD SYSOUT=*
//SMPECSI DD DSN=&SMPE..CSI,DISP=OLD
//ZPOOL DD DSN=SYS1.MACLIB(GIMZPOOL),DISP=SHR
//SYSIN DD *
REPRO OUTFILE(SMPECSI) INFILE(ZPOOL)
//

```

Figure A-2 SMPJOB01 — Allocate the VISION:Transact SMP/E data sets and initialize the CSI (Page 5 of 5)

SMPJOB02

```

//          JOB
//*
//* MEMBER SMPJOB02
//*****
//*
//* DEFINE THE VISION:Transact CSI GLOBAL, TARGET & DISTRIBUTION ZONES
//*
//*****
//*
//* REPLACE THE JOB CARD ABOVE.
//*
//* CHANGE THE SET PARAMETER BELOW TO MATCH THE VALUE ASSIGNED IN
//* JOB SMPJOB01.
//*
//*****
//*
//* SET SMPE=TRANSACTION.SMPE    HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
//*
//*****
//*
//* CHANGE THE CSI DATASET QUALIFIERS IN THE ZONEINDEX ENTRY BELOW
//* TO MATCH THE VALUE IN THE SET STATEMENT ABOVE.
//*
//* CHANGE THE DA NAMES FOR THE DDDEF ENTRIES BELOW BEGINNING
//* WITH 'SMP' AND WITH 'TR' AND WITH 'WB' TO THE DATASET NAMES
//* THAT WERE ALLOCATED IN SMPJOB01.
//*
//* FIND THE TWO OCCURRENCES OF THE DDDEF ENTRIES FOR RESLIB BELOW
//* AND CHANGE THE DA NAMES TO YOUR IMS SYSTEM RESIDENCE LIBRARY.
//*
//* FIND THE TWO OCCURRENCES OF THE DDDEF ENTRIES FOR DSNLOAD BELOW
//* AND CHANGE THE DA NAMES TO YOUR DB2 SYSTEM LOAD LIBRARY.
//*
//* MAKE OTHER CHANGES AS APPROPRIATE FOR YOUR INSTALLATION.
//*
//*****
//*
//SMPE    EXEC PGM=GIMSMP,REGION=4M
//SMPCSI  DD DSN=&SMPE..CSI,DISP=SHR
//SMPLOG  DD DSN=&SMPE..SMPLOG,DISP=SHR
//SMPLOGA DD DSN=&SMPE..SMPLOGA,DISP=SHR
//SMPPTS  DD DSN=&SMPE..SMPPTS,DISP=SHR
//SMPOUT  DD SYSOUT=*
//SMPPUNCH DD SYSOUT=*
//SMRPRT  DD SYSOUT=*
//SMPLIST DD SYSOUT=*
//SMPSNAP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD DUMMY
//SMPTLIB DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SMPWRK1 DD UNIT=SYSDA,SPACE=(CYL,(1,1,5))
//SMPWRK2 DD UNIT=SYSDA,SPACE=(CYL,(1,1,5))
//SMPWRK3 DD UNIT=SYSDA,SPACE=(CYL,(1,1,5))
//SMPWRK4 DD UNIT=SYSDA,SPACE=(CYL,(1,1,5))
//SMPWRK5 DD UNIT=SYSDA,SPACE=(CYL,(1,1,5))
//SMPWRK6 DD UNIT=SYSDA,SPACE=(CYL,(1,1,5))
//SYSUT1  DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSUT2  DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSUT3  DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSUT4  DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SMPCNTL DD *
SET BDY(GLOBAL) .
UCLIN.
ADD GLOBALZONE
SREL (Z038)
FMID (CDDX078)
OPTIONS(TR78OPT)
ZONEDESCRIPTION(Advantage VISION:Transact for CICS, Version 7.8)
ZONEINDEX(
    (TR78CTZ,TRANSACTION.SMPE.CSI,TARGET)

```

Figure A-3 SMPJOB02 — Define VISION:Transact to the Global, Target, and Distribution zones (Page 1 of 5)

```

                                (TR78CDZ, TRANSACT.SMPE.CSI,DLIB)
                                ).
ADD OPTIONS (TR78OPT)
  AMS (AMS)
  ASM (ASSEM)
  COMP (COMPRESS)
  COPY (COPY)
  LKED (LINKEDIT)
  NOPURGE
  NOREJECT
  RETRY (RETRY)
  UPDATE (UPDATE)
  ZAP (IMASPZAP) .
ADD UTILITY (AMS)          NAME (IDCAMS) .
ADD UTILITY (ASSEM)        NAME (ASMA90)
                                PARM (DECK,NOOBJECT,XREF (SHORT)) RC (04) .
ADD UTILITY (COMPRESS)    NAME (IEBCOPY) .
ADD UTILITY (COPY)        NAME (IEBCOPY) .
ADD UTILITY (LINKEDIT)    NAME (IEWL) PARM (LET,LIST,MAP) RC (08) .
ADD UTILITY (RETRY)        NAME (IEBCOPY) .
ADD UTILITY (UPDATE)      NAME (IEBUPDTE) .
ADD UTILITY (IMASPZAP)    NAME (IMASPZAP) PARM (IGNIDRFULL) RC (04) .
ADD DDDEF (SMPLOG)
  DA (TRANSACT.SMPE.SMPLOG)
  MOD.
ADD DDDEF (SMPLOGA)
  DA (TRANSACT.SMPE.SMPLOGA)
  MOD.
ADD DDDEF (SMPPTS)
  DA (TRANSACT.SMPE.SMPPTS)
  OLD.
ADD DDDEF (SMPOUT)        SYSOUT (*) .
ADD DDDEF (SMPRPT)        SYSOUT (*) .
ADD DDDEF (SYSPRINT)      SYSOUT (*) .
ADD DDDEF (SMPTLIB)        UNIT (SYSDA) CYL SPACE (1,1) .
ADD DDDEF (SYSUT1)         UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ADD DDDEF (SYSUT2)         UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ADD DDDEF (SYSUT3)         UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ADD DDDEF (SYSUT4)         UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ENDUCL.
SET BDY (TR78CTZ) .
UCLIN.
  ADD TARGETZONE (TR78CTZ)
    SREL (Z038)
    RELATED (TR78CDZ)
    OPTIONS (TR78OPT) .
  ADD DDDEF (SMPSCDS)
    DA (TRANSACT.SMPE.SMPSCDS)
    OLD.
  ADD DDDEF (SMPMTS)
    DA (TRANSACT.SMPE.SMPMTS)
    OLD.
  ADD DDDEF (SMPPTS)
    DA (TRANSACT.SMPE.SMPPTS)
    SHR.
  ADD DDDEF (SMPSTS)
    DA (TRANSACT.SMPE.SMPSTS)
    OLD.
  ADD DDDEF (SMPLOG)
    DA (TRANSACT.SMPE.SMPLOG)
    MOD.
  ADD DDDEF (SMPLOGA)
    DA (TRANSACT.SMPE.SMPLOGA)
    MOD.
  ADD DDDEF (TRTGEN)
    DA (TRANSACT.TR078.TRTGEN)
    SHR.
  ADD DDDEF (TRTCICS)
    DA (TRANSACT.TR078.TRTCICS)
    SHR.
  ADD DDDEF (TRTWORK)
    DA (TRANSACT.TR078.TRTWORK)
    SHR.
  ADD DDDEF (TRTTKCS)
    DA (TRANSACT.TR078.TRTTKCS)

```

Figure A-3 SMPJOB02 — Define VISION:Transact to the Global, Target, and Distribution zones (Page 2 of 5)

```

        SHR.
ADD DDDEF (TRTTKCL)
DA      (TRANSACTION.TR078.TR078.TR078.TR078)
        SHR.
ADD DDDEF (WBTLOAD)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBTPANEL)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBTMSG)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBTSEKLS)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBTCLIST)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (TRDGEN)
DA      (TRANSACTION.TR078.TR078.TR078.TR078)
        SHR.
ADD DDDEF (TRDCICS)
DA      (TRANSACTION.TR078.TR078.TR078.TR078)
        SHR.
ADD DDDEF (TRDWORK)
DA      (TRANSACTION.TR078.TR078.TR078.TR078)
        SHR.
ADD DDDEF (TRDTKCS)
DA      (TRANSACTION.TR078.TR078.TR078.TR078)
        SHR.
ADD DDDEF (TRDTKCL)
DA      (TRANSACTION.TR078.TR078.TR078.TR078)
        SHR.
ADD DDDEF (WBDLOAD)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBDPANEL)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBDMSG)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBDSEKLS)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (WBDCLIST)
DA      (TRANSACTION.WB061.WB061.WB061.WB061)
        SHR.
ADD DDDEF (RESLIB)
DA      (IMSSYS81.RESLIB)
        SHR.
ADD DDDEF (DSNLOAD)
DA      (DB2.DB2610.SDSNLOAD)
        SHR.
ADD DDDEF (SDFHLOAD)
DA      (CICSSYS.CTS220.CICS.SDFHLOAD)
        SHR.
ADD DDDEF (SDFHMAC)
DA      (CICSSYS.CTS220.CICS.SDFHMAC)
        SHR.
ADD DDDEF (SCEEMAC)
DA      (CEE.SCEEMAC)
        SHR.
ADD DDDEF (SYSLIB)  CONCAT (SDFHMAC
                           SCEEMAC
                           TRTWORK) .
ADD DDDEF (SMPOUT)  SYSOUT (*).
ADD DDDEF (SMPPUNCH) SYSOUT (*).
ADD DDDEF (SMPRPT)  SYSOUT (*).
ADD DDDEF (SMPLIST)  SYSOUT (*).
ADD DDDEF (SMPSNAP)  SYSOUT (*).
ADD DDDEF (SYSPRINT) SYSOUT (*).
ADD DDDEF (SYSUDUMP) SYSOUT (*).
ADD DDDEF (SMPTLIB) UNIT (SYSDA) CYL SPACE (1,1) .

```

Figure A-3 SMPJOB02 — Define VISION:Transact to the Global, Target, and Distribution zones (Page 3 of 5)

```

ADD DDDEF(SMPWRK1) UNIT(SYSDA) CYL SPACE(3,1) DIR(20) NEW DELETE.
ADD DDDEF(SMPWRK2) UNIT(SYSDA) CYL SPACE(3,1) DIR(20) NEW DELETE.
ADD DDDEF(SMPWRK3) UNIT(SYSDA) CYL SPACE(3,1) DIR(20) NEW DELETE.
ADD DDDEF(SMPWRK4) UNIT(SYSDA) CYL SPACE(3,1) DIR(20) NEW DELETE.
ADD DDDEF(SMPWRK5) UNIT(SYSDA) CYL SPACE(5,5) DIR(50) NEW DELETE.
ADD DDDEF(SMPWRK6) UNIT(SYSDA) CYL SPACE(1,1) DIR(10) NEW DELETE.
ADD DDDEF(SYSUT1) UNIT(SYSDA) CYL SPACE(5,2) NEW DELETE.
ADD DDDEF(SYSUT2) UNIT(SYSDA) CYL SPACE(5,2) NEW DELETE.
ADD DDDEF(SYSUT3) UNIT(SYSDA) CYL SPACE(5,2) NEW DELETE.
ADD DDDEF(SYSUT4) UNIT(SYSDA) CYL SPACE(5,2) NEW DELETE.
ENDUCL.
SET BDY(TR78CDZ) .
UCLIN.
  ADD DLIBZONE(TR78CDZ)
  SREL (Z038)
  RELATED(TR78CTZ)
  OPTIONS(TR78OPT) .
ADD DDDEF(SMPSCDS)
  DA (TRANSACTION.SMPE.SMPSCDS)
  OLD.
ADD DDDEF(SMPMTS)
  DA (TRANSACTION.SMPE.SMPMTS)
  OLD.
ADD DDDEF(SMPPTS)
  DA (TRANSACTION.SMPE.SMPPTS)
  SHR.
ADD DDDEF(SMPSTS)
  DA (TRANSACTION.SMPE.SMPSTS)
  OLD.
ADD DDDEF(SMPLOG)
  DA (TRANSACTION.SMPE.SMPLOG)
  MOD.
ADD DDDEF(SMPLOGA)
  DA (TRANSACTION.SMPE.SMPLOGA)
  MOD.
ADD DDDEF(TRDGEN)
  DA (TRANSACTION.TR078.TRDGEN)
  SHR.
ADD DDDEF(TRDCICS)
  DA (TRANSACTION.TR078.TRDCICS)
  SHR.
ADD DDDEF(TRDWORK)
  DA (TRANSACTION.TR078.TRDWORK)
  SHR.
ADD DDDEF(TRDTKCS)
  DA (TRANSACTION.TR078.TRDTKCS)
  SHR.
ADD DDDEF(TRDTKCL)
  DA (TRANSACTION.TR078.TRDTKCL)
  SHR.
ADD DDDEF(WBDLOAD)
  DA (TRANSACTION.WB061.WBDLOAD)
  SHR.
ADD DDDEF(WBDPANEL)
  DA (TRANSACTION.WB061.WBDPANEL)
  SHR.
ADD DDDEF(WBDMGS)
  DA (TRANSACTION.WB061.WBDMGS)
  SHR.
ADD DDDEF(WBDSKELS)
  DA (TRANSACTION.WB061.WBDSKELS)
  SHR.
ADD DDDEF(WBDCLIST)
  DA (TRANSACTION.WB061.WBDCLIST)
  SHR.
ADD DDDEF(TRTGEN)
  DA (TRANSACTION.TR078.TRRTGEN)
  SHR.
ADD DDDEF(TRTCICS)
  DA (TRANSACTION.TR078.TRTCICS)
  SHR.
ADD DDDEF(TRTWORK)
  DA (TRANSACTION.TR078.TRRTWORK)
  SHR.
ADD DDDEF(TRTTKCS)

```

Figure A-3 SMPJOB02 — Define VISION:Transact to the Global, Target, and Distribution zones (Page 4 of 5)

```

        DA      (TRANSACT.TR078.TRITTKCS)
        SHR.
ADD DDDEF (TRITTKCL)
        DA      (TRANSACT.TR078.TRITTKCL)
        SHR.
ADD DDDEF (WBTLOAD)
        DA      (TRANSACT.WB061.WBTLOAD)
        SHR.
ADD DDDEF (WBTPANEL)
        DA      (TRANSACT.WB061.WBTPANEL)
        SHR.
ADD DDDEF (WBTMSGGS)
        DA      (TRANSACT.WB061.WBTMSGGS)
        SHR.
ADD DDDEF (WBTSKELS)
        DA      (TRANSACT.WB061.WBTSKELS)
        SHR.
ADD DDDEF (WBTCLIST)
        DA      (TRANSACT.WB061.WBTCLIST)
        SHR.
ADD DDDEF (RESLIB)
        DA      (IMSSYS81.RESLIB)
        SHR.
ADD DDDEF (DSNLOAD)
        DA      (DB2.DB2610.SDSNLOAD)
        SHR.
ADD DDDEF (SDFHLOAD)
        DA      (CICSSYS.CTS220.CICS.SDFHLOAD)
        SHR.
ADD DDDEF (SDFHMAC)
        DA      (CICSSYS.CTS220.CICS.SDFHMAC)
        SHR.
ADD DDDEF (SCEEMAC)
        DA      (CEE.SCEEMAC)
        SHR.
ADD DDDEF (SYSLIB)  CONCAT (SDFHMAC
                           SCEEMAC
                           TRDWORK) .
ADD DDDEF (SMPOUT)  SYSOUT (*) .
ADD DDDEF (SMPPUNCH) SYSOUT (*) .
ADD DDDEF (SMPRPT)  SYSOUT (*) .
ADD DDDEF (SMPLIST) SYSOUT (*) .
ADD DDDEF (SMPSNAP) SYSOUT (*) .
ADD DDDEF (SYSPRINT) SYSOUT (*) .
ADD DDDEF (SYSUDUMP) SYSOUT (*) .
ADD DDDEF (SMPTLIB) UNIT (SYSDA) CYL SPACE (1,1) .
ADD DDDEF (SMPWRK1) UNIT (SYSDA) CYL SPACE (3,1) DIR (20) NEW DELETE.
ADD DDDEF (SMPWRK2) UNIT (SYSDA) CYL SPACE (3,1) DIR (20) NEW DELETE.
ADD DDDEF (SMPWRK3) UNIT (SYSDA) CYL SPACE (3,1) DIR (20) NEW DELETE.
ADD DDDEF (SMPWRK4) UNIT (SYSDA) CYL SPACE (3,1) DIR (20) NEW DELETE.
ADD DDDEF (SMPWRK5) UNIT (SYSDA) CYL SPACE (5,5) DIR (50) NEW DELETE.
ADD DDDEF (SMPWRK6) UNIT (SYSDA) CYL SPACE (1,1) DIR (10) NEW DELETE.
ADD DDDEF (SYSUT1)  UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ADD DDDEF (SYSUT2)  UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ADD DDDEF (SYSUT3)  UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ADD DDDEF (SYSUT4)  UNIT (SYSDA) CYL SPACE (5,2) NEW DELETE.
ENDUCL.

```

Figure A-3 SMPJOB02 — Define VISION:Transact to the Global, Target, and Distribution zones (Page 5 of 5)

SMPJOB03

```

//          JOB ...
// *
// * MEMBER SMPJOB03
// *****
// *
// * RECEIVE THE VISION:TRANSACT PROGRAM ELEMENTS INTO THE SMP/E
// * GLOBAL ZONE
// *
// *****
// *
// * REPLACE THE JOB CARD ABOVE.
// *
// * CHANGE THE SET PARAMETER BELOW TO MATCH THE VALUE ASSIGNED IN
// * JOB SMPJOB01.
// *
// *****
// *
// * SET SMPE=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
// * SET CNTL=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CNTL DSN
// *
// *****
// *
// RECEIVE EXEC PGM=GIMSMP,REGION=4M
// SMPCSI DD DSN=&SMPE..CSI,DISP=SHR
// SMPPTFIN DD DSN=&CNTL..SMPCNTL(SMPMCS0),DISP=SHR MCS HEADER
// DD DSN=&CNTL..SMPCNTL(SMPMCS1G),DISP=SHR GENERATOR
// DD DSN=&CNTL..SMPCNTL(SMPMCS1C),DISP=SHR CICS EXECUTOR
// DD DSN=&CNTL..SMPCNTL(SMPMCS2),DISP=SHR COMLIB
// DD DSN=&CNTL..SMPCNTL(SMPMCS3),DISP=SHR ISPF WORKBENCH
// DD DSN=&CNTL..SMPCNTL(SMPMCS4),DISP=SHR SAS/C RUNTIME
// DD DSN=&CNTL..SMPCNTL(SMPMCS5C),DISP=SHR CICS TOOLKIT
// SMPCNTL DD *
// SET BDY(GLOBAL).
// RECEIVE SYSMODS LIST.
// LIST.
//

```

Figure A-4 SMPJOB03 — RECEIVE the VISION:Transact program element SYSMODS into the Global zone

SMPJOB04

```

//          JOB ...
//*
//* MEMBER SMPJOB04
//*****
//*
//* APPLY THE VISION:TRANSACT ELEMENTS TO THE SMP/E
//* TARGET ZONE
//*
//*****
//*
//* REPLACE THE JOB CARD ABOVE.
//*
//* CHANGE THE SET PARAMETERS BELOW TO MATCH THE VALUES ASSIGNED IN
//* JOB SMPJOB01.
//*
//*****
//*
// SET SMPE=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
// SET CNTL=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CNTL DSN
// SET MARKV=TRANSACT.TR078    HIGH-LEVEL QUALIFIER FOR TRANSACT DSN'S
// SET BENCH=TRANSACT.WB061    HIGH-LEVEL QUALIFIER FOR WORKBENCH DSN'S
//*
//APPLY EXEC PGM=GIMSMP,REGION=4M
//SMPCSI DD DSN=&SMPE..CSI,DISP=SHR
//IJCLIN DD DSN=&CNTL..SMPCNTL,DISP=SHR
//TRIWORK DD DSN=&MARKV..WORKLIB,DISP=SHR
//TRIGEN DD DSN=&MARKV..GENLIB,DISP=SHR
//TRICICS DD DSN=&MARKV..CICSLIB,DISP=SHR
//TRITKCL DD DSN=&MARKV..CICSTKL,DISP=SHR
//TRITKCS DD DSN=&MARKV..CICSTKS,DISP=SHR
//WBILOAD DD DSN=&BENCH..LOADLIB,DISP=SHR
//WBIPANEL DD DSN=&BENCH..PANELS,DISP=SHR
//WBIMSGS DD DSN=&BENCH..MSGS,DISP=SHR
//WBISKELS DD DSN=&BENCH..SKELS,DISP=SHR
//WBICLIST DD DSN=&BENCH..CLIST,DISP=SHR
//SMPCNTL DD *
//      SET BDY(TR78CTZ).
//      APPLY SELECT(CDDX078).
//      LIST.
//

```

Figure A-5 SMPJOB04 — APPLY the VISION:Transact program element SYSMODS into the Target zone

SMPJOB05

```

//          JOB ...
// *
// * MEMBER SMPJOB05
// *****
// *
// * ACCEPT THE VISION:TRANSACTION PROGRAM ELEMENTS INTO THE SMP/E
// * DISTRIBUTION ZONE
// *
// *****
// *
// * REPLACE THE JOB CARD ABOVE.
// *
// * CHANGE THE SET PARAMETERS BELOW TO MATCH THE VALUES ASSIGNED IN
// * JOB SMPJOB01.
// *
// *****
// *
// SET SMPE=TRANSACTION.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
// SET CNTL=TRANSACTION.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CNTL DSN
// SET MARKV=TRANSACTION.TR078    HIGH-LEVEL QUALIFIER FOR TRANSACTION DSN'S
// SET BENCH=TRANSACTION.WB061    HIGH-LEVEL QUALIFIER FOR WORKBENCH DSN'S
// *
//ACCEPT EXEC PGM=GIMSMP,REGION=4M
//SMPCSI DD DSN=&SMPE..CSI,DISP=SHR
//IJCLIN DD DSN=&CNTL..SMPCNTL,DISP=SHR
//TRIWORK DD DSN=&MARKV..WORKLIB,DISP=SHR
//TRIGEN DD DSN=&MARKV..GENLIB,DISP=SHR
//TRICICS DD DSN=&MARKV..CICSLIB,DISP=SHR
//TRITKCL DD DSN=&MARKV..CICSTKL,DISP=SHR
//TRITKCS DD DSN=&MARKV..CICSTKS,DISP=SHR
//WBILOAD DD DSN=&BENCH..LOADLIB,DISP=SHR
//WBIPANEL DD DSN=&BENCH..PANELS,DISP=SHR
//WBIMSGS DD DSN=&BENCH..MSGS,DISP=SHR
//WBISKELS DD DSN=&BENCH..SKELS,DISP=SHR
//WBICLIST DD DSN=&BENCH..CLIST,DISP=SHR
//SMPCNTL DD *
//      SET BDY(TR78CDZ).
//      ACCEPT SELECT(CDDX078).
//      LIST.
//

```

Figure A-6 SMPJOB05 — ACCEPT the VISION:Transact program element SYSMODS into the Distribution zone

ZACCEPT

```

//          JOB ...
//
// * MEMBER ZACCEPT
// *****
// *
// * ACCEPT THE VISION:TRANSACT APARS AND PTFS INTO THE SMP/E
// * DISTRIBUTION ZONE
// *
// *****
// *
// * REPLACE THE JOB CARD ABOVE.
// *
// * CHANGE THE SET PARAMETERS BELOW TO MATCH THE VALUES ASSIGNED IN
// * JOB SMPJOB01.
// *
// * CHANGE THE ACCEPT SELECT PARAMTER TO THE APAR OR PTF YOU WANT TO
// * ACCEPT.
// *
// * NOTE THAT THE APAR OR PTF WILL BE ACCEPTED INTO THE DISTRIBUTION
// * ZONE LIBRARIES DEFINED VIA THE DDDEF ENTRIES FOR THAT ZONE.
// * SEE JOB SMPJOB02 FOR THE APPLICABLE DDDEF ENTRIES.
// *
// * IF YOUR OPERATIONAL LIBRARIES ARE A COPY OF THE DISTRIBUTION ZONE
// * LIBRARIES, YOU MUST REFRESH YOUR OPERATIONAL LIBRARIES FROM THE
// * DISTRIBUTION ZONE LIBRARIES AFTER THE ACCEPT IS COMPLETE IN ORDER
// * FOR THE APAR OR PTF TO TAKE EFFECT IN YOUR OPERATIONAL LIBRARIES.
// *
// *****
// *
// * SET SMPE=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
// * SET CNTL=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CNTL DSN
// *
//ACCEPT EXEC PGM=GIMSMP,REGION=4M
//SMPCSI DD DSN=&SMPE..CSI,DISP=SHR
//IJCLIN DD DSN=&CNTL..SMPCNTL,DISP=SHR
//SMPCNTL DD *
//      SET BDY(TR78CDZ) .
//      ACCEPT SELECT(NNNNNNN) .
//      LIST.
//

```

Figure A-7 ZACCEPT — ACCEPT any APAR and PTF SYSMODS into the Distribution zone

ZAPPLY

```

//          JOB ...
//*
//* MEMBER ZAPPLY
//*****
//*
//* APPLY THE VISION:TRANSACTION APARS AND PTFS TO THE SMP/E
//* TARGET ZONE
//*
//*****
//*
//* REPLACE THE JOB CARD ABOVE.
//*
//* CHANGE THE SET PARAMETERS BELOW TO MATCH THE VALUES ASSIGNED IN
//* JOB SMPJOB01.
//*
//* CHANGE THE APPLY SELECT PARAMETER TO THE APAR OR PTF YOU WANT TO
//* APPLY.
//*
//* NOTE THAT THE APAR OR PTF WILL BE APPLIED TO THE TARGET ZONE
//* LIBRARIES DEFINED VIA THE DDDEF ENTRIES FOR THE TARGET ZONE.
//* SEE JOB SMPJOB02 FOR THE APPLICABLE DDDEF ENTRIES.
//*
//* IF YOUR OPERATIONAL LIBRARIES ARE A COPY OF THE TARGET ZONE
//* LIBRARIES, YOU MUST REFRESH YOUR OPERATIONAL LIBRARIES FROM THE
//* TARGET ZONE LIBRARIES AFTER THE APPLY IS COMPLETE IN ORDER FOR
//* THE APAR OR PTF TO TAKE EFFECT IN YOUR OPERATIONAL LIBRARIES.
//*
//*****
//*
//* SET SMPE=TRANSACTION.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
//* SET CNTL=TRANSACTION.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CNTL DSN
//*
//APPLY EXEC PGM=GIMSMP,REGION=4M
//SMPCSI DD DSN=&SMPE..CSI,DISP=SHR
//IJCLIN DD DSN=&CNTL..SMPCNTL,DISP=SHR
//SMPCNTL DD *
//      SET BDY(TR78CTZ) .
//      APPLY SELECT(NNNNNNN) .
//      LIST.
//

```

Figure A-8 ZAPPLY — APPLY any APAR and PTF SYSMODS into the Target zone

ZRECEIVE

```

//          JOB ...
//*
/* MEMBER ZRECEIVE
/******
/*
/* RECEIVE THE VISION:TRANSACT USERMODS, APARS, PTFs INTO THE SMP/E
/* GLOBAL ZONE
/*
/******
/*
/* REPLACE THE JOB CARD ABOVE.
/*
/* CHANGE THE SET PARAMETERS BELOW TO MATCH THE VALUE ASSIGNED IN
/* JOB SMPJOB01.
/*
/* CHANGE THE MEMBER NAME FOR THE SMPPTFIN DSN TO THE APAR OR PTF
/* MEMBER NAME YOU WANT TO RECEIVE.
/*
/******
/*
/* SET SMPE=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
/* SET CNTL=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CNTL DSN
/*
/******
/*
/*RECEIVE EXEC PGM=GIMSMP,REGION=4M
/*SMPCSI DD DSN=&SMPE..CSI,DISP=SHR
/*SMPPTFIN DD DSN=&CNTL..SMPCNTL(NNNNNNN),DISP=SHR
/*SMPCNTL DD *
/*      SET BDY(GLOBAL).
/*      RECEIVE SYSMODS LIST.
/*      LIST.
//

```

Figure A-9 ZRECEIVE — RECEIVE any APAR and PTF SYSMODS into the Global zone

ZREJECT

```
//          JOB ...
// *
// * MEMBER ZREJECT
// *****
// *
// * REJECT (REMOVE) VISION:TRANSACT APARS AND PTF'S FROM THE SMP/E
// * GLOBAL ZONE
// *
// *****
// *
// * REPLACE THE JOB CARD ABOVE.
// *
// * CHANGE THE SET PARAMETER BELOW TO MATCH THE VALUE ASSIGNED IN
// * JOB SMPJOB01.
// *
// * CHANGE THE REJECT SELECT PARAMETER TO THE APAR OR PTF YOU WANT TO
// * REJECT.
// *
// *****
// *
// * SET SMPE=TRANSACT.SMPE      HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
// *
// *****
// *
// REJECT EXEC PGM=GIMSMP,REGION=4M
// SMPCSI DD DSN=&SMPE..CSI,DISP=SHR
// SMPCTL DD *
// SET BDY(GLOBAL) .
// REJECT SELECT (NNNNNN) .
// LIST.
//
```

Figure A-10 ZREJECT — REJECT any APAR and PTF SYSMODS that were received into the Global zone

ZRESTORE

```

//          JOB ...
//*
//* MEMBER ZRESTORE
//*****
//*
//* RESTORE (REMOVE) VISION:TRANSACT APARS AND PTFS FROM THE SMP/E
//* TARGET ZONE
//*
//*****
//*
//* REPLACE THE JOB CARD ABOVE.
//*
//* CHANGE THE SET PARAMETERS BELOW TO MATCH THE VALUE ASSIGNED IN
//* JOB SMPJOB01.
//*
//* CHANGE THE RESTORE SELECT PARAMETER TO THE APAR OR PTF YOU WANT
//* REMOVE.
//*
//* NOTE THAT THIS JOB WILL RESTORE YOUR TARGET ZONE LIBRARIES TO THE
//* STATE IN WHICH THEY WERE BEFORE THE APAR OR PTF WAS APPLIED.
//*
//* IF YOUR OPERATIONAL LIBRARIES ARE A COPY OF THE TARGET ZONE
//* LIBRARIES, YOU MUST REFRESH YOUR OPERATIONAL LIBRARIES FROM THE
//* TARGET ZONE LIBRARIES AFTER THE RESTORE IS COMPLETE IN ORDER
//* FOR THE RESTORE TO AFFECT YOUR OPERATIONAL LIBRARIES.
//*
//*****
//*
//* SET SMPE=TRANSACT.SMPE    HIGH-LEVEL QUALIFIER FOR SMP/E CSI DSN
//*
//*****
//*
//RESTORE EXEC PGM=GIMSMP,REGION=4M
//SMPCSI   DD DSN=&SMPE..CSI,DISP=SHR
//SMPCNTL  DD *
//          SET BDY(TR78CTZ).
//          RESTORE SELECT(NNNNNNN).
//          LIST.
//

```

Figure A-11 ZRESTORE — RESTORE (remove) any APAR and PTF SYSMODS that were applied to the Target zone

Customization and Setup JCL Samples

This appendix lists the JCL members used in the product customization and setup process. The members are contained in the TRANSACT.TR078.WORKLIB data set. A brief description of each of the JCL members appears below, followed by a listing of the members.

Note: There are other non-JCL members in the WORKLB data set. These members are used as input data files and may be modified as discussed in [Chapter 2, “Installation Instructions”](#).

Member	Description
CLASM1	Assemble and linkedit the COMLIB component parameter module MARKLIBP
QSCOBEX	Execute the COBOL Quick Start utility
QSCOBLL	Set up the COBOL Quick Start utility for CA-Librarian library access
QSCOBPL	Set up the COBOL Quick Start utility for CA-Panvalet library access
QSDB2BND	Bind the DB2 Plan for the DB2 Quick Start utility
QSDB2EX	Execute the DB2 Quick Start utility
QSINQEX	Execute the VISION:Inquiry Quick Start utility
QSRESEX	Execute the VISION:Results Quick Start utility
QSRESLL	Set up the VISION:Results Quick Start utility for CA-Librarian library access
QSRESPL	Set up the VISION:Results Quick Start utility for CA-Panvalet library access
TRACICEM	Assemble and linkedit the CICS runtime error message module FIVCM078

Member	Description
TRACICS1	Pre-process, assemble and linkedit the CICS Terminal program interface modules CEEUOPT, MZCOB, and MZCICS
TRACICS2	Pre-process, assemble and linkedit the CICS COMMAREA program interface modules CEEUOPT, MZCOB, and MZCOMM
TRACICS3	Pre-process, assemble and linkedit the CICS application entry module AECICS
TRACICTM	Assemble and linkedit the CICS runtime trace message module FIVCT078
TRASM1	Assemble and linkedit the Generator component parameter modules FIVEPARM and M5LEPARM
TRBMSE	Assemble and linkedit the BMS error display map FIVCE07
TRCOPY1	Copy the Generator and Executor operational libraries to new data sets
TRCOPY2	Copy the Generator and Executor operational libraries to existing data sets
TRCSDUP	Define the Executor program elements to CICS
TRDB2A	Assemble and linkedit the Generator DB2 Interface module FIVESQL
TRDB2B	Bind the Generator DB2 plan FIVESQL
TRIVP	Verify that the Generator component install was successful
WBCOPY1	Copy the ISPF Workbench operational libraries to new data sets
WBCOPY2	Copy the ISPF Workbench operational libraries to existing data sets

CLASM1

```

//          JOB ...                                00010002
//*                                                00020002
//* MEMBER CLASM1                                00030000
//*****00040000
//* THIS PROCEDURE AND JOB STREAM ASSEMBLES AND LINKS THE 00050000
//* COMLIB COMPONENT PARAMETER MODULE                00060000
//* MARKLIBP.                                         00070000
//*                                                  00080000
//ASMBPRM PROC SRCLIB=,                             00090000
//          LOADLIB=                                00100000

```

Figure B-1 CLASM1 — Assemble and linkedit the COMLIB component parameter module MARKLIBP (Page 1 of 2)

```

//ASM      EXEC PGM=ASMA90,REGION=2M,PARM='NODECK,OBJECT,LIST'          00110001
//SYSPRINT DD SYSOUT=*                                                  00120000
//SYSLIB   DD DSN=SYS1.MACLIB,DISP=SHR                                00130000
//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(1,1))                            00140000
//SYSLIN   DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(1,1))            00150000
//SYSIN    DD DSN=&SRCLIB(MARKLIBP),DISP=SHR                          00160000
//*                                                00170000
//LINK     EXEC PGM=IEWL,REGION=2M,PARM='LET,LIST,MAP,NCAL'            00180000
//SYSPRINT DD SYSOUT=*                                                  00190000
//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(1,1))                            00200000
//SYSIMOD  DD DSN=&LOADLIB(MARKLIBP),DISP=SHR                        00210000
//SYSLIN   DD DSN=*.ASM.SYSLIN,DISP=(OLD,DELETE)                      00220000
//          PEND                                                        00230000
//*****00240000
//* THE FOLLOWING IS A SAMPLE EXECUTION OF THE JOBS TO ASSEMBLE        00250000
//* AND LINK THE COMLIB COMPONENT PARAMETER MODULE.                   00260000
//*                                                                    00270000
//* BEFORE YOU RUN THIS JOB, SPECIFY:                                  00280000
//*                                                                    00290000
//*      SRCLIB - THE SOURCE PDS THAT CONTAINS THE PARAMETER MODULE.  00300000
//*      LOADLIB - THE LOAD LIBRARY TO CONTAIN THE LINK EDITED MODULE. 00310000
//*                                                                    00320000
//*****00330000
//* MARKLIBP ASSEMBLY AND LINK                                         00340000
//*****00350000
//*                                                                    00360000
//CLPARM   EXEC ASMBPRM,                                                00370000
//          SRCLIB='TRANSACT.TR078.WORKLIB',                          00380002
//          LOADLIB='TRANSACT.TR078.GENLIB'                            00390002
//                                                                    00400000

```

Figure B-1 CLASM1 — Assemble and linkedit the COMLIB component parameter module MARKLIBP (Page 2 of 2)

QSCOBEX

```

//          JOB ...                                                    00010000
//*                                                00020000
//* MEMBER QSCOBEX                                                    00030000
//*****00040000
//* EXECUTE THE COBOL QUICK START UTILITY. * 00050000
//* ***** NOTE ***** * 00060000
//* THE SYSCOPY DD STATEMENT IS USED FOR MVS COPYBOOK LIBRARIES. * 00070000
//* THE PANDD1 DD STATEMENT IS USED FOR PANVALET COPYBOOK LIBRARIES. * 00080000
//* THE MASTER DD STATEMENT IS USED FOR LIBRARIAN COPYBOOK LIBRARIES * 00090000
//*****00100000
//COBOLQS PROC TRLOAD=,                                                00110000
//          COPYLIB=,                                                  00120000
//          DEFLIB=,                                                    00130000
//          MEMBER=,                                                    00140000
//COBOLQS EXEC PGM=COBOLQS,REGION=2M                                    00150000
//STEPLIB  DD DSN=&TRLOAD,DISP=SHR                                      00160000
//SYSPRINT DD SYSOUT=*                                                  00170000
//SYSCOPY  DD DSN=&COPYLIB,DISP=SHR                                      00180000
//PANDD1   DD DSN=&COPYLIB,DISP=SHR                                      00190000
//MASTER  DD DSN=&COPYLIB,DISP=SHR                                      00200000
//SYS004   DD DSN=&DEFLIB(&MEMBER),DISP=OLD                             00210000
//SYSIN    DD DUMMY                                                    00220000
//          PEND                                                        00230000
//*****00240000
//* BEFORE SUBMITTING THIS JCL, YOU MUST SPECIFY THE FOLLOWING * 00250000
//* INFORMATION: * 00260000
//* * 00270000
//* TRLOAD - NAME OF YOUR TRANSACT GENERATOR LOAD LIBRARY. * 00280000
//* COPYLIB - NAME OF YOUR COBOL COPY LIBRARY. THIS IS AN * 00290000
//* MVS, PANVALET, OR LIBRARIAN COPYBOOK LIBRARY. * 00300000
//* DEFLIB - NAME OF YOUR TRANSACT DEFINITION LIBRARY (PDS). * 00310000
//* THE GENERATED FILE DEFINITION IS WRITTEN TO * 00320000

```

Figure B-2 QSCOBEX — Execute the COBOL Quick Start utility (Page 1 of 2)

```

/*          THIS LIBRARY.                                * 00330000
/* MEMBER - MEMBER NAME FOR THE DEFINITION YOU ARE GENERATING. * 00340000
/*          * 00350000
/* YOU MUST ALSO PROVIDE THE APPROPRIATE SYSIN DATA IN THE * 00360000
/* COBOLQS.SYSIN DD OVERRIDE STMT. * 00370000
/*          * 00380000
//QS EXEC COBOLQS, * 00390000
// TRLOAD='TRANSACTION.TR078.TRTGEN', * 00400000
// COPYLIB='COBOL.COPYBOOK.LIBRARY', * 00410000
// DEFLIB='YOUR.OWN.DEFLIB', * 00420000
// MEMBER='SAMPLEFD' * 00430000
//COBOLQS.SYSIN DD * 00440000
FILEGEN NAME=SAMPLEFD,TYPE=FIXED,RECSIZE=80 00450000
SEGMENT NAME=OFFICE,NUMBER=10,LEVEL=1 00460000
$COBOL 00470000
      01 OFFICE-DATA. 00480000
        02 OFFICE-CODE PIC S9(3). 00490000
        02 OFFICE-ADDRESS. 00500000
          03 OFFICE-STREET PIC X(20). 00510000
          03 OFFICE-CITY PIC X(15). 00520000
          03 OFFICE-STATE PIC X(2). 00530000
          03 OFFICE-ZIP. 00540000
            04 OFFICE-ZIP-FIRST-FIVE PIC X(5). 00550000
            04 OFFICE-ZIP-LAST-FOUR PIC X(4). 00560000
        02 OFFICE-PHONE PIC 9(7). 00570000
        02 OFFICE-AREA-CODE PIC X(3). 00580000
        02 SPEED-DIAL PIC X(3). 00590000
        02 FILLER PIC X(18). 00600000
$ECOBOL 00610000
/* 00620000

```

Figure B-2 QSCOBEX — Execute the COBOL Quick Start utility (Page 2 of 2)

QSCOBLL

```

//          JOB ... 00010000
/* 00020000
/* MEMBER QSCOBLL 00030000
/*          * 00040000
/* LINK EDIT LIBRARIAN INTERFACE MODULES WITH COBOL QUICK START. * 00050000
/*          * 00060000
//TRLIBLK PROC TRLOAD=, 00070000
// LIBLOAD= 00080000
//LINK EXEC PGM=IEWL,REGION=2M,PARM='LET,LIST,MAP,NCAL' 00090000
//SYSLIB DD DUMMY 00100000
//SYSPRINT DD SYSOUT=* 00110000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00120000
//LIBSYS DD DSN=&LIBLOAD,DISP=SHR 00130000
//LLIB DD DSN=&TRLOAD,DISP=SHR 00140000
//SYSLMOD DD DSN=&TRLOAD,DISP=SHR 00150000
// PEND 00160000
/*          * 00170000
/* BEFORE SUBMITTING THIS JCL, YOU MUST SPECIFY THE FOLLOWING * 00180000
/* INFORMATION: * 00190000
/*          * 00200000
/* TRLOAD - NAME OF YOUR TRANSACTION GENERATOR LOAD LIBRARY. * 00210000
/* LIBLOAD - NAME OF YOUR LIBRARIAN SYSTEM LOAD LIBRARY. * 00220000
/*          * 00230000
/*          * 00240000
//LIBLINK EXEC TRLIBLK, 00250000
// TRLOAD='TRANSACTION.TR078.TRTGEN', 00260000
// LIBLOAD='LIBRARN.SYSTEM.LOADLIB' 00270000
//LINK.SYSLIN DD * 00280000
INCLUDE LIBSYS(FAIRCLS) 00290000
INCLUDE LIBSYS(FAIRERR) 00300000
INCLUDE LIBSYS(FAIRLOC) 00310000
INCLUDE LIBSYS(FAIRMOD) 00320000

```

Figure B-3 QSCOBLL — Set up the COBOL Quick Start utility for Librarian library access (Page 1 of 2)

```

INCLUDE LIBSYS (FAIRNTE)                                00330000
INCLUDE LIBSYS (FAIROPN)                                00340000
INCLUDE LIBSYS (FAIRPNT)                                00350000
INCLUDE LIBSYS (FAIRREC)                                00360000
INCLUDE LIBSYS (FAIRSCAN)                               00370000
INCLUDE LIBSYS (FAIRSEC)                                00380000
INCLUDE LLIB (COMLIBL)                                  00390000
ENTRY COMLIBL                                           00400000
NAME COMLIBL (R)                                        00410000
//                                                       00420000

```

Figure B-3 QSCOBLL — Set up the COBOL Quick Start utility for Librarian library access (Page 2 of 2)

QSCOBPL

```

//              JOB ...                                00001000
// *                                                     00002000
// * MEMBER CLCOBPL                                     00010000
// *                                                     00020000
// * LINK EDIT PANVALET INTERFACE MODULES WITH COBOL QUICK START. * 00030000
// *                                                     00040000
// * TRPANLK PROC TRLOAD=,                               00050000
// *               PANLOAD=                               00060000
// * LINK EXEC PGM=IEWL,REGION=2M,PARM='LET,LIST,MAP,NCAL' 00070000
// * SYSLIB DD DUMMY                                     00080000
// * SYSPRINT DD SYSOUT=*                                00090000
// * SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))              00100000
// * LIBSYS DD DSN=&PANLOAD,DISP=SHR                      00110000
// * LLIB DD DSN=&TRLOAD,DISP=SHR                         00120000
// * SYSLMOD DD DSN=&TRLOAD,DISP=SHR                      00130000
// * PEND                                                 00140000
// *                                                     00150000
// * BEFORE SUBMITTING THIS JCL, YOU MUST SPECIFY THE FOLLOWING * 00160000
// * INFORMATION:                                         * 00170000
// *                                                     * 00180000
// * TRLOAD - NAME OF YOUR TRANSACT GENERATOR LOAD LIBRARY. * 00190000
// * PANLOAD - NAME OF YOUR PANVALET SYSTEM LOAD LIBRARY. * 00200000
// *                                                     * 00210000
// *                                                     00220000
// * PANLINK EXEC TRPANLK,                               00230000
// *               TRLOAD='TRANSACT.TR078.TRGGEN',        00240000
// *               PANLOAD='PANVALET.SYSTEM.LOADLIB'      00250000
// * LINK.SYSLIN DD *                                     00260000
// * INCLUDE LIBSYS (PAM)                                00270000
// * INCLUDE LLIB (COMLIBP)                              00280000
// * ENTRY COMLIBP                                        00290000
// * NAME COMLIBP (R)                                    00300000
//                                                       00310000

```

Figure B-4 QSCOBPL — Set up the COBOL Quick Start utility for Panvalet library access

QSDB2BND

```

//          JOB ...                                00001000
//*                                                00002000
//* MEMBER QSDB2BND                                00010000
//*****                                          00020000
//* BIND THE DB2 PLAN FOR THE DB2 QUICK START UTILITY. * 00030000
//*                                                00040000
//* BEFORE SUBMITTING THIS JCL, YOU MUST SPECIFY THE FOLLOWING * 00050000
//* INFORMATION:                                     * 00060000
//*                                                * 00070000
//* - NAME OF YOUR DB2 SYSTEM DSNLOAD LIBRARY AS THE * 00080000
//* STEPLIB DATA SET NAME.                         * 00090000
//*                                                * 00100000
//* - APPROPRIATE VALUES FOR THE SYSTSIN DSN PARAMETERS: * 00110000
//*                                                * 00120000
//* PARAMETER                                       * 00130000
//* -----                                       * 00140000
//* SYSTEM      YOUR DB2 SUBSYSTEM ID.             * 00150000
//* PLAN        NAME OF YOUR DB2 PLAN FOR THE DB2 QUICK * 00160000
//*              START UTILITY.                     * 00170000
//* LIBRARY     NAME OF THE LIBRARY CONTAINING THE DB2 * 00180000
//*              QUICK START DATA BASE REQUEST MODULE (DBRM). * 00190000
//*              THE DB2 QUICK START DBRM IS DELIVERED IN THE * 00200000
//*              TRANSACT WORKLIB PDS MEMBER DB2QDBRM. * 00210000
//* MEMBER     NAME OF THE DB2 QUICK START DBRM - DB2QDBRM * 00220000
//* QUALIFIER   NAME OF THE QUALIFIER FOR YOUR DB2 SYSTEM * 00230000
//*              CATALOG TABLE. THIS MUST BE THE QUALIFIER * 00240000
//*              FOR YOUR SYSCOLUMNS TABLE (E.G., SYSIBM). * 00250000
//*****                                          00260000
//BINDPGM EXEC PGM=IKJEFT01,DYNAMNBR=20,REGION=1024K 00270000
//STEPLIB DD DISP=SHR,DSN=DB2.SYSTEM.DSNLOAD        00280000
//SYSTSPRT DD SYSOUT=*                               00290000
//SYSTSIN DD *                                       00300000
//          DSN SYSTEM(???)                          00310000
//          BIND PLAN (DB2QS) -                      00320000
//          LIBRARY ('TRANSACT.TR078.WORKLIB') -      00330000
//          MEMBER (DB2QDBRM) -                      00340000
//          QUALIFIER (SYSIBM) -                     00350000
//          ACTION (REPLACE ) RETAIN                  00360000
//          END                                        00370000
//                                                    00380000

```

Figure B-5 QSDB2BND — Bind the DB2 Plan for the DB2 Quick Start utility

QSDB2EX

```

//          JOB ...                                00010000
//*                                                00020000
//* MEMBER QSDB2EX                                00030000
//*****                                          00040000
//* EXECUTE THE DB2 QUICK START UTILITY. * 00050000
//*****                                          00060000
//DB2QS PROC TRLOAD=,                               00070000
//          DB2LOAD=,                                00080000
//          DEFLIB=,                                 00090000
//DB2QS EXEC PGM=DB2QS,REGION=1024K                 00100000
//STEPLIB DD DSN=&TRLOAD,DISP=SHR                   00110000
//          DD DSN=&DB2LOAD,DISP=SHR                 00120000
//SYSTEM DD DUMMY                                   00130000
//SYSPRINT DD SYSOUT=*,                             00140000
//          DCB=(DSORG=PS,RECFM=FBA,LRECL=133,BLKSIZE=1330) 00150000
//SYS004 DD DSN=&DEFLIB,DISP=OLD                    00160000
//SYSIN DD DUMMY                                    00170000
//          PEND                                       00180000
//*****                                          00190000

```

Figure B-6 QSDB2EX — Execute the DB2 Quick Start utility (Page 1 of 2)

```

/* BEFORE SUBMITTING THIS JCL, YOU MUST SPECIFY THE FOLLOWING      * 00200000
/* INFORMATION:                                                    * 00210000
/*                                                                    * 00220000
/* TRLOAD - NAME OF YOUR TRANSACT GENERATOR LOAD LIBRARY.         * 00230000
/* DB2LOAD - NAME OF YOUR DB2 DSN.DSNLOAD LIBRARY.                 * 00240000
/* DEFLIB - NAME OF YOUR SOURCE CODE DEFINITION LIBRARY.          * 00250000
/* THE GENERATED FILE DEFINITION IS WRITTEN TO                     * 00260000
/* THIS LIBRARY.                                                    * 00270000
/*                                                                    * 00280000
/* YOU MUST ALSO PROVIDE THE APPROPRIATE SYSIN DATA IN THE        * 00290000
/* DB2QS.SYSIN DD OVERRIDE STMT.                                    * 00300000
/* *****                                                         * 00310000
/*                                                                    * 00320000
//QS EXEC DB2QS,                                                    * 00330000
// TRLOAD='TRANSACT.TR0478.TRTGEN',                                * 00340000
// DB2LOAD='DB2.SYSTEM.DSNLOAD',                                    * 00350000
// DEFLIB='YOUR.OWN.DEFLIB'                                         * 00360000
/*                                                                    * 00370000
//DB2QS.SYSIN DD *                                                 * 00380000
DB2CNTL DB2PLAN=DB2QS,DB2SYS=DB2T                                  * 00390000
FILEGEN NAME=DB2FD,BUFFSIZE=1024K                                  * 00400000
SEGMENT NAME=DEPT,NUMBER=10,LEVEL=1, TABLE=DEPT,CREATOR=DSN8230, * 00410000
PRINT=ALL                                                            * 00420000
NEWPAGE                                                              * 00430000
SEGMENT NAME=EMPLOYEE,NUMBER=20,LEVEL=2, TABLE=EMP,CREATOR=DSN8230, * 00440000
PRINT=ALL                                                            * 00450000
NEWPAGE                                                              * 00460000
SEGMENT NAME=PROJECT,NUMBER=30,LEVEL=2, TABLE=PROJ,CREATOR=DSN8230, * 00470000
PRINT=ALL                                                            * 00480000
//                                                                    * 00490000

```

Figure B-6 QSDB2EX — Execute the DB2 Quick Start utility (Page 2 of 2)

QSINQEX

```

// JOB ...                                                         00010000
/*                                                                    00020000
/* MEMBER QSINQEX                                                  00030000
/* *****                                                         00040000
/* UTILITY TO CONVERT VISION:INQUIRY FILE DEFINITIONS INTO         * 00050000
/* VISION:TRANSACT FORMAT FILE DEFINITIONS.                        * 00060000
/* THE VISION:INQUIRY FILE DEFINITIONS MUST COME FROM AN           * 00070000
/* VISION:INQUIRY UNLOADED SYSTEM DATABASE FILE. SEE YOUR          * 00080000
/* VISION:INQUIRY TECHNICAL REFERENCE MANUAL FOR INFORMATION ON    * 00090000
/* HOW TO CREATE AN UNLOADED COPY OF THE SYSTEM DATABASE.         * 00100000
/*                                                                    * 00110000
/* THIS UTILITY MAY ALSO BE INVOKED UNDER TSO/ISPF USING THE      * 00120000
/* ISPF WORKBENCH IMPORT FUNCTION.                                  * 00130000
/* *****                                                         00140000
//INQRYQS PROC RGN=2M,                                             00150000
// TRLOAD=,                                                         00160000
// ULSYSDB=,                                                         00170000
// DEFLIB=                                                           00180000
//INQRYQS EXEC PGM=INQRYQS,REGION=&RGN                             00190000
//STEPLIB DD DISP=SHR,DSN=&TRLOAD                                   00200000
//SYSPRINT DD SYSOUT=*                                              00210000
//SYSUT1 DD DISP=SHR,DSN=&ULSYSDB                                   00220000
//SYS004 DD DISP=OLD,DSN=&DEFLIB                                    00230000
// PEND                                                            00240000
/* *****                                                         00250000
/* FOLLOWING IS A SAMPLE EXECUTION OF THIS PROCEDURE. BEFORE YOU   * 00260000
/* RUN THIS PROCEDURE, SPECIFY:                                     * 00270000
/*                                                                    * 00280000
/* RGN - THE REGION SIZE; DEFAULT IS 2M.                           * 00290000
/* TRLOAD - THE TRANSACT GENERATOR LOAD LIBRARY.                   * 00300000
/* ULSYSDB - THE UNLOADED VISION:INQUIRY SYSTEM DATABASE FILE.     * 00310000
/* DEFLIB - THE VISION:TRANSACT DEFINITION LIBRARY.                * 00320000

```

Figure B-7 QSINQEX — Execute the VISION:Inquiry Quick Start utility (Page 1 of 2)

```

//***** 00330000
//STEP01 EXEC INQRYQS,RGN=2M, 00340000
//          TRLOAD='TRANSACTION.TR078.TRTGEN', 00350000
//          ULSYSDB='VISION.INQUIRY.UNLOADED.SYSDBASE', 00360000
//          DEFLIB='VISION.TRANSACTION.DEFLIB' 00370000
//SYSIN DD * 00380000
//          FILEGEN NAME=VSHPLANT,FLDPREFX=PLT 00390000
//          FILEGEN NAME=SALARIES,FLDPREFX=SAL 00400000
// 00410000

```

Figure B-7 QSINQEX — Execute the VISION:Inquiry Quick Start utility (Page 2 of 2)

QSRESEX

```

//          JOB ... 00010000
//* 00020000
//* MEMBER QSRESEX 00030000
//***** 00040000
//* EXECUTE THE RESULTS QUICK START UTILITY * 00050000
//* ***** NOTE ***** * 00060000
//* THE SYSCOPY DD STATEMENT IS USED FOR MVS COPYBOOK LIBRARIES. * 00070000
//* THE PANDD1 DD STATEMENT IS USED FOR PANVALET COPYBOOK LIBRARIES. * 00080000
//* THE MASTER DD STATEMENT IS USED FOR LIBRARIAN COPYBOOK LIBRARIES * 00090000
//***** 00100000
//RESULTQS PROC RGN=2M, 00110000
//          TRLOAD=, 00130000
//          DEFLIB=, 00140000
//          MEMBER=, 00150000
//          RSLTLIB=, 00160000
//          RSLTDEF= 00170000
//CONVRT EXEC PGM=RESULTQS,REGION=&RGN 00180000
//STEPLIB DD DISP=SHR,DSN=&TRLOAD 00190000
//SYSPRINT DD SYSOUT=* 00210000
//*SYSCOPY DD DISP=SHR,DSN=USER.RESULTS.COPYLIB 00220000
//*PANDD1 DD DISP=SHR,DSN=USER.PANVALET.LIBRARY 00230000
//*MASTER DD DISP=SHR,DSN=USER.LIBR.MASTER 00240000
//SYS004 DD DISP=OLD,DSN=&DEFLIB(&MEMBER) 00250000
//SYSIN DD DISP=SHR,DSN=&RSLTLIB(&RSLTDEF) 00260000
//          PEND 00270000
//***** 00280000
//* FOLLOWING IS A SAMPLE EXECUTION OF THIS PROCEDURE. BEFORE YOU * 00290000
//* RUN THIS PROCEDURE, SPECIFY: * 00300000
//* * 00310000
//* RGN - THE REGION SIZE. DEFAULT IS 2M. * 00320000
//* TRLOAD - THE NAME OF YOUR TRANSACTION GENERATOR LOAD LIBRARY. * 00340000
//* DEFLIB - THE LIBRARY TO CONTAIN THE GENERATED TRANSACTION * 00350000
//* FILE DEFINITION SOURCE CODE. * 00351000
//* MEMBER - THE MEMBER NAME FOR THE CONVERTED VISION:TRANSACTION * 00360000
//* FILE DEFINITION IN THE DEFINITION LIBRARY. * 00370000
//* RSLTLIB - THE PDS CONTAINING THE VISION:RESULTS FILE * 00380000
//* DEFINITION SOURCE STATEMENTS. * 00390000
//* RSLTDEF - THE PDS MEMBER NAME OF THE INPUT VISION:RESULTS * 00400000
//* FILE DEFINITION TO BE CONVERTED. * 00410000
//* * 00420000
//* *** NOTE *** * 00430000
//* * 00440000
//* THIS PROCEDURE ASSUMES INPUT FROM A PDS MEMBER. OPTIONALLY, IT * 00450000
//* MAY ALSO COME FROM A RESULTS COPY (MVS PDS), COPYP (PANVALET), * 00460000
//* OR COPYL (LIBRARIAN) STATEMENT. IF SO, YOU MUST UN-COMMENT THE * 00470000
//* APPROPRIATE SYSCOPY (MVS PDS), PANDD1 (PANVALET), OR MASTER * 00480000
//* (LIBRARIAN) DD STATEMENT IN THE PROCEDURE, SPECIFYING THE * 00490000
//* PROPER DATA SET NAME FOR THE LIBRARY USED. PLEASE REFER TO THE * 00500000

```

Figure B-8 QSRESEX — Execute the VISION:Results Quick Start utility (Page 1 of 2)

```

/* * MANUAL FOR DETAILS IN SETTING UP COPY SUPPORT. * 00510000
/* ***** 00520000
//STEP01 EXEC RESLTQS,RGN=2M, 00530000
//      TRLOAD='TRANSACT.TR078.TRTGEN', 00550000
//      DEFLIB='VISION.TRANSACT.DEFLIB', 00560000
//      MEMBER=FILENAME, 00570000
//      RSLTLIB='VISION.RESULTS.FILEDEFS', 00580000
//      RSLTDEF=FILENAME 00590000
// 00600000

```

Figure B-8 QSRESEX — Execute the VISION:Results Quick Start utility (Page 2 of 2)

QSRESLL

```

//      JOB ... 00010000
/* * 00020000
/* * MEMBER QSRESLL 00030000
/* ***** 00040000
/* * LINK LIBRARIAN INTERFACE MODULES WITH RESULTS QUICK START. * 00050000
/* ***** 00060000
//LBLNK PROC LOADLIB=, 00070000
//      LIBLOAD= 00080000
//LINK EXEC PGM=IEWL,REGION=2M,PARM='LIST,MAP,LET,NCAL' 00090000
//SYSLIB DD DUMMY 00100000
//SYSPRINT DD SYSOUT=* 00110000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00120000
//LIBSYS DD DISP=SHR,DSN=&LIBLOAD 00130000
//SYSLMOD DD DISP=SHR,DSN=&LOADLIB 00140000
//      PEND 00150000
/* ***** 00160000
/* * BEFORE SUBMITTING THIS JCL, YOU MUST SPECIFY THE FOLLOWING * 00170000
/* * INFORMATION: * 00180000
/* * 00190000
/* * LOADLIB - NAME OF YOUR TRANSACT GENERATOR LOAD LIBRARY. * 00200000
/* * LIBLOAD - NAME OF YOUR LIBRARIAN SYSTEM LOAD LIBRARY. * 00210000
/* ***** 00220000
//LIBLINK EXEC LBLNK, 00230000
//      LOADLIB='TRANSACT.TR078.TRTGEN', 00240000
//      LIBLOAD='LIBRARN.SYSTEM.LOADLIB' 00250000
//LINK.SYSLIN DD * 00260000
INCLUDE LIBSYS (FAIRCLS) 00270000
INCLUDE LIBSYS (FAIROPN) 00280000
INCLUDE LIBSYS (FAIRREC) 00290000
INCLUDE LIBSYS (FAIRMOD) 00300000
INCLUDE LIBSYS (FAIRERR) 00310000
INCLUDE LIBSYS (FAIRLOC) 00320000
INCLUDE LIBSYS (FAIRNTE) 00330000
INCLUDE LIBSYS (FAIRPNT) 00340000
INCLUDE LIBSYS (FAIRSCAN) 00350000
INCLUDE LIBSYS (FAIRSEC) 00360000
INCLUDE SYSLMOD (DYL280LX) 00370000
ENTRY DYL280L 00380000
NAME DYL280L (R) 00390000
// 00400000

```

Figure B-9 QSRESLL — Set up the VISION:Results Quick Start utility for Librarian library access

QSRESPL

```

//          JOB ...                                00010000
//*                                                00020000
//* MEMBER QSRESPL                                00030000
//*****                                          00040000
//* LINK PANVALET INTERFACE MODULES WITH RESULTS QUICK START. * 00050000
//*****                                          00060000
//PNLNK  PROC LOADLIB=,                            00070000
//          PANLOAD=,                              00080000
//LINK   EXEC PGM=IEWL,REGION=2M,PARM='LIST,MAP,LET,NCAL' 00090000
//SYSLIB DD DUMMY                                00100000
//SYSPRINT DD SYSOUT=*                          00110000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))         00120000
//LIBSYS DD DISP=SHR,DSN=&PANLOAD                00130000
//SYSLMOD DD DISP=SHR,DSN=&LOADLIB               00140000
//          PEND                                  00150000
//*****                                          00160000
//* BEFORE SUBMITTING THIS JCL, YOU MUST SPECIFY THE FOLLOWING * 00170000
//* INFORMATION:                                  * 00180000
//*                                                * 00190000
//*   LOADLIB - NAME OF YOUR TRANSACT GENERATOR LOAD LIBRARY * 00200000
//*   PANLOAD - NAME OF YOUR PANVALET SYSTEM LOAD LIBRARY.  * 00210000
//*****                                          00220000
//PANLINK EXEC PNLNK,                             00230000
//          LOADLIB='TRANSACT.TR078.TRTGEN',        00240000
//          PANLOAD='PANVALET.SYSTEM.LOADLIB'        00250000
//LINK.SYSLIN DD *                                00260000
//          INCLUDE LIBSYS (PAM)                    00270000
//          INCLUDE SYSLMOD(DYL280PX)                00280000
//          ENTRY DYL280P                            00290000
//          NAME DYL280P(R)                          00300000
//                                                   00310000

```

Figure B-10 QSRESPL — Set up the VISION:Results Quick Start utility for Panvalet library access

TRACICEM

```

//          JOB ...                                00010002
//*                                                00020003
//* MEMBER TRACICEM                                00030000
//*****                                          00040000
//*                                                00050000
//* THIS JOB ASSEMBLES AND LINK EDITS THE CICS EXECUTION-TIME ERROR * 00060000
//* MESSAGES FOR VISION:TRANSACT.                 00070000
//*                                                00080000
//*****                                          00090000
//* ASSEMBLE THE CICS EXECUTION-TIME ERROR MESSAGE MODULE 00100000
//*                                                00110000
//*                                                00120000
//ASM   EXEC PGM=ASMA90,REGION=2M,                00130002
//          PARM='LIST,NODECK,OBJECT,RENT'          00140000
//SYSIN  DD DISP=SHR,DSN=TRANSACT.TR078.WORKLIB(FIVCM078) 00150001
//SYSPRINT DD SYSOUT=*                          00160000
//SYSLIN DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(5,3)), 00170000
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)      00180000
//SYSLIB DD DISP=SHR,DSN=TRANSACT.TR078.WORKLIB        00190001
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))             00200000
//*                                                00210000

```

Figure B-11 TRACICEM — Assemble and linkedit the CICS runtime error message module FIVCM078 (Page 1 of 2)

```

//* LINKEDIT THE CICS EXECUTION-TIME ERROR MESSAGE MODULE 00220000
//* 00230000
//LKED EXEC PGM=IEWL,PARM='LET,LIST,MAP,NCAL,RENT',REGION=2M, 00240000
// COND=(8,LT,ASM) 00250000
//SYSLIN DD DSN=*.ASM.SYSLIN,DISP=(OLD,DELETE) 00260000
//SYSPRINT DD SYSOUT=* 00270000
//SYSLMOD DD DISP=SHR,DSN=TRANSACT.TR078.CICSLIB(FIVCM078) 00280001
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00290000
// 00300000

```

Figure B-11 TRACICEM — Assemble and linkedit the CICS runtime error message module FIVCM078 (Page 2 of 2)

TRACICS1

```

// JOB ... 00010002
//* 00020002
//* MEMBER TRACICS1 00030002
//***** 00040000
//* 00050000
//* PROCEDURE AND JOBSTREAM TO TRANSLATE, COMPILE, AND LINKEDIT 00060000
//* THE COMPONENTS OF THE CICS INTERFACE MODULE FIVCI078. 00070001
//* 00080000
//***** 00090000
//* 00100000
//TRACICS1 PROC CICSREF=, CICS SYSTEM DATASET PREFIX 00110000
// COBCOMP=, COBOL COMPILER DATASET PREFIX 00120000
// LEPREF=, LANGUAGE ENVIRONMENT DATASET PREFIX 00130000
// WORKLIB=, VISION:TRANSACT WORK LIBRARY 00140000
// CICSLIB=, VISION:TRANSACT CICS LIBRARY 00150000
//* 00160000
//* TRANSLATE THE MZCCOB COMPONENT 00170000
//* 00180000
//TRAN1 EXEC PGM=DFHECP1$,REGION=2M,PARM='COBOL3' 00190000
//STEPLIB DD DISP=SHR,DSN=&CICSREF..SDFHLOAD 00200000
//SYSIN DD DISP=SHR,DSN=&WORKLIB(MZCCOB) 00210000
//SYSPRINT DD SYSOUT=* 00220000
//SYSPUNCH DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(20,10)) 00230000
//* 00240000
//* COMPILE THE MZCCOB COMPONENT 00250000
//* 00260000
//COB1 EXEC PGM=IGYCRCTL,REGION=2M,PARM='LIB,NODYNAM' 00270000
//STEPLIB DD DSN=&COBCOMP..SIGYCOMP,DISP=SHR 00280002
//SYSIN DD DSN=*.TRAN1.SYSPUNCH,DISP=(OLD,DELETE) 00290000
//SYSPRINT DD SYSOUT=* 00300000
//SYSLIB DD DISP=SHR,DSN=&CICSREF..SDFHCOB 00310000
// DD DISP=SHR,DSN=&WORKLIB 00320000
//SYSLIN DD DISP=OLD,DSN=&WORKLIB(MZOBJC11) 00330002
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00340000
//SYSUT2 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00350000
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00360000
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00370000
//SYSUT5 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00380000
//SYSUT6 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00390000
//SYSUT7 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00400000
//* 00410000
//* TRANSLATE THE MZCICS COMPONENT 00420000
//* 00430000
//TRAN2 EXEC PGM=DFHEAP1$,REGION=2M 00440000
//STEPLIB DD DISP=SHR,DSN=&CICSREF..SDFHLOAD 00450000
//SYSIN DD DISP=SHR,DSN=&WORKLIB(MZCICS) 00460000
//SYSPRINT DD SYSOUT=* 00470000
//SYSPUNCH DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(20,10)) 00480000
//* 00490000
//* ASSEMBLE THE MZCICS COMPONENT 00500000

```

Figure B-12 TRACICS1 — Pre-process, assemble and linkedit the CICS Terminal program interface modules CEEUOPT, MZCOB, and MZCICS (Page 1 of 2)

```

//* 00510000
//ASM2 EXEC PGM=ASMA90,REGION=2M, 00520002
// PARM='RENT,NODECK,OBJECT,LIST' 00530000
//SYSIN DD DSN=*.TRAN2.SYSPUNCH,DISP=(OLD,DELETE) 00540000
//SYSPRINT DD SYSOUT=* 00550000
//SYSLIB DD DISP=SHR,DSN=&WORKLIB 00560000
// DD DISP=SHR,DSN=&CICSPREF..SDFHMAC 00570000
// DD DISP=SHR,DSN=&LEPREF..SCEEMAC 00580000
// DD DISP=SHR,DSN=SYS1.MACLIB 00590000
//SYSLIN DD DISP=OLD,DSN=&WORKLIB (MZOBJCI2) 00600002
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00610000
//* 00620000
//* ASSEMBLE THE CEEUOPT COMPONENT 00630000
//* 00640000
//ASM3 EXEC PGM=ASMA90,REGION=2M, 00650002
// PARM='RENT,NODECK,OBJECT,LIST' 00660000
//SYSIN DD DISP=OLD,DSN=&WORKLIB (CEEUOPT) 00670000
//SYSPRINT DD SYSOUT=* 00680000
//SYSLIB DD DISP=SHR,DSN=&LEPREF..SCEEMAC 00690000
//SYSLIN DD DISP=OLD,DSN=&WORKLIB (MZOBJCI3) 00700002
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00710000
//* 00720000
//* LINKEDIT THE FIVCI078 MODULE 00730001
//* 00740000
//LINK EXEC PGM=IEWL,REGION=2M,COND=(4,LT), 00750000
// PARM='LET,LIST,MAP,RENT' 00760000
//SYSLIN DD DISP=SHR,DSN=&WORKLIB (TRLINKCI) 00770000
//SYSPRINT DD SYSOUT=* 00780000
//SYSLIB DD DISP=SHR,DSN=&CICSPREF..SDFHLOAD 00790000
// DD DISP=SHR,DSN=&LEPREF..SCEELKED 00800000
//OBJLIB DD DISP=SHR,DSN=&WORKLIB 00810000
//SYSLMOD DD DISP=OLD,DSN=&CICSLIB (FIVCI078) 00820001
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00830000
// PEND 00840000
//* 00850000
//***** 00860000
//* 00870000
//MZCICS EXEC TRACICS1, 00880002
// CICSPREF='CICSSYS.CTS220.CICS', 00890002
// COBCOMP='IGY', 00900002
// LEPREF='CEE', 00910002
// WORKLIB='TRANSACTION.TR078.WORKLIB', 00920001
// CICSLIB='TRANSACTION.TR078.CICSLIB' 00930001
// 00940000

```

Figure B-12 TRACICS1 — Pre-process, assemble and linkedit the CICS Terminal program interface modules CEEUOPT, MZCOB, and MZCICS (Page 2 of 2)

TRACICS2

```

// JOB ... 00010000
//* 00020000
//* MEMBER TRACICS2 00030000
//***** 00040000
//* 00050000
//* PROCEDURE AND JOBSTREAM TO TRANSLATE, COMPILE, AND LINKEDIT 00060000
//* THE COMPONENTS OF THE CICS INTERFACE MODULE FIVCJ078. 00070000
//* 00080000
//***** 00090000
//* 00100000
//TRACICS2 PROC CICSPREF=, CICS SYSTEM DATASET PREFIX 00110000
// COBCOMP=, COBOL COMPILER DATASET PREFIX 00120000
// LEPREF=, LANGUAGE ENVIRONMENT DATASET PREFIX 00130000
// WORKLIB=, VISION:TRANSACTION WORK LIBRARY 00140000

```

Figure B-13 TRACICS2 — Pre-process, assemble and linkedit the CICS COMMAREA program interface modules CEEUOPT, MZCOB, and MZCOMM (Page 1 of 3)

```

//          CICSLIB=          VISION:TRANSACT CICS LIBRARY          00150000
//          *                  00160000
//          *          TRANSLATE THE MZCCOB COMPONENT          00170000
//          *                  00180000
//TRAN1 EXEC PGM=DFHECP1$,REGION=2M,PARM='COBOL3'          00190000
//STEPLIB DD DISP=SHR,DSN=&CICSPREF..SDFHLOAD          00200000
//SYSIN DD DISP=SHR,DSN=&WORKLIB(MZCCOB)          00210000
//SYSPRINT DD SYSOUT=*          00220000
//SYSPUNCH DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(20,10))          00230000
//          *                  00240000
//          *          COMPILE THE MZCCOB COMPONENT          00250000
//          *                  00260000
//COB1 EXEC PGM=IGYCRCTL,REGION=2M,PARM='LIB,NODYNAM'          00270000
//STEPLIB DD DSN=&COBCOMP..SIGYCOMP,DISP=SHR          00280000
//SYSIN DD DSN=*.TRAN1.SYSPUNCH,DISP=(OLD,DELETE)          00290000
//SYSPRINT DD SYSOUT=*          00300000
//SYSLIB DD DISP=SHR,DSN=&CICSPREF..SDFHCOB          00310000
//          DD DISP=SHR,DSN=&WORKLIB          00320000
//SYSLIN DD DISP=OLD,DSN=&WORKLIB(MZOBJCJ1)          00330000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00340000
//SYSUT2 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00350000
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00360000
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00370000
//SYSUT5 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00380000
//SYSUT6 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00390000
//SYSUT7 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00400000
//          *                  00410000
//          *          TRANSLATE THE MZCOMM COMPONENT          00420000
//          *                  00430000
//TRAN2 EXEC PGM=DFHEAP1$,REGION=2M          00440000
//STEPLIB DD DISP=SHR,DSN=&CICSPREF..SDFHLOAD          00450000
//SYSIN DD DISP=SHR,DSN=&WORKLIB(MZCOMM)          00460000
//SYSPRINT DD SYSOUT=*          00470000
//SYSPUNCH DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(20,10))          00480000
//          *                  00490000
//          *          ASSEMBLE THE MZCOMM COMPONENT          00500000
//          *                  00510000
//ASM2 EXEC PGM=ASMA90,REGION=2M,          00520000
//          PARM='RENT,NODECK,OBJECT,LIST'          00530000
//SYSIN DD DSN=*.TRAN2.SYSPUNCH,DISP=(OLD,DELETE)          00540000
//SYSPRINT DD SYSOUT=*          00550000
//SYSLIB DD DISP=SHR,DSN=&WORKLIB          00560000
//          DD DISP=SHR,DSN=&CICSPREF..SDFHMAC          00570000
//          DD DISP=SHR,DSN=&LEPREF..SCEEMAC          00580000
//          DD DISP=SHR,DSN=SYS1.MACLIB          00590000
//SYSLIN DD DISP=OLD,DSN=&WORKLIB(MZOBJCJ2)          00600000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00610000
//          *                  00620000
//          *          ASSEMBLE THE CEEUOPT COMPONENT          00630000
//          *                  00640000
//ASM3 EXEC PGM=ASMA90,REGION=2M,          00650000
//          PARM='RENT,NODECK,OBJECT,LIST'          00660000
//SYSIN DD DISP=OLD,DSN=&WORKLIB(CEEUOPT)          00670000
//SYSPRINT DD SYSOUT=*          00680000
//SYSLIB DD DISP=SHR,DSN=&LEPREF..SCEEMAC          00690000
//SYSLIN DD DISP=OLD,DSN=&WORKLIB(MZOBJCJ3)          00700000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00710000
//          *                  00720000
//          *          LINKEDIT THE FIVCJ078 MODULE          00730000
//          *                  00740000
//LINK EXEC PGM=IEWL,REGION=2M,COND=(4,LT),          00750000
//          PARM='LET,LIST,MAP,RENT'          00760000
//SYSLIN DD DISP=SHR,DSN=&WORKLIB(TRLINKCJ)          00770000
//SYSPRINT DD SYSOUT=*          00780000
//SYSLIB DD DISP=SHR,DSN=&CICSPREF..SDFHLOAD          00790000
//          DD DISP=SHR,DSN=&LEPREF..SCEELKED          00800000
//OBJLIB DD DISP=SHR,DSN=&WORKLIB          00810000
//SYSLMOD DD DISP=OLD,DSN=&CICSLIB(FIVCJ078)          00820000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00830000
//          PEND          00840000
//          *                  00850000

```

Figure B-13 TRACICS2 — Pre-process, assemble and linkedit the CICS COMMAREA program interface modules CEEUOPT, MZCOB, and MZCOMM (Page 2 of 3)

```

//***** 00860000
//* 00870000
//MZCOMM EXEC TRACICS2, 00880000
// CICSREF='CICSSYS.CTS220.CICS', 00890000
// COBCOMP='IGY', 00900000
// LEPREF='CEE', 00910000
// WORKLIB='TRANSACT.TR078.WORKLIB', 00920000
// CICSCLIB='TRANSACT.TR078.CICSCLIB' 00930000
// 00940000

```

Figure B-13 TRACICS2 — Pre-process, assemble and linkedit the CICS COMMAREA program interface modules CEEUOPT, MZCOB, and MZCOMM (Page 3 of 3)

TRACICS3

```

// JOB ... 00010002
//* 00020002
//* MEMBER TRACICS3 00030002
//***** 00040000
//* 00050000
//* PROCEDURE AND JOBSTREAM TO TRANSLATE, ASSEMBLE, AND LINKEDIT 00060000
//* THE CICS APPLICATION ENTRY ROUTINE FIVCA078. 00070001
//* 00080000
//***** 00090000
//* 00100000
//TRACICS3 PROC WORKLIB=, 00110000
// CICSREF=, 00120000
// CICSCLIB= 00130000
//* 00140000
//* EXECUTE THE CICS COMMAND LEVEL TRANSLATOR 00150000
//* 00160000
//TRAN EXEC PGM=DFHEAP1$,REGION=2M 00170000
//STEPLIB DD DISP=SHR,DSN=&CICSREF..SDFHLOAD 00180000
//SYSIN DD DISP=SHR,DSN=&WORKLIB(AECICS) 00190002
//SYSPRINT DD SYSOUT=* 00200000
//SYSPUNCH DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(20,10)) 00210000
//* 00220000
//* ASSEMBLE THE PROGRAM 00230000
//* 00240000
//ASM EXEC PGM=ASMA90,REGION=2M, 00250002
// PARM='RENT,NODECK,OBJECT,LIST' 00260000
//SYSIN DD DSN=*.TRAN.SYSPUNCH,DISP=(OLD,DELETE) 00270000
//SYSPRINT DD SYSOUT=* 00280000
//SYSLIB DD DISP=SHR,DSN=&WORKLIB 00290000
// DD DISP=SHR,DSN=&CICSREF..SDFHMAC 00300000
// DD DISP=SHR,DSN=SYS1.MACLIB 00310000
//SYSLIN DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(10,10)), 00320000
// DCB=BLKSIZE=3200 00330000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00340000
//* 00350000
//* LINKEDIT THE PROGRAM 00360000
//* 00370000
//LINK EXEC PGM=IEWL,REGION=2M,COND=(4,LT,ASM), 00380000
// PARM='LET,LIST,MAP,NCAL,RENT' 00390000
//SYSLIN DD DSN=*.ASM.SYSLIN,DISP=(OLD,DELETE) 00400000
//SYSPRINT DD SYSOUT=* 00410000
//SYSLMOD DD DISP=SHR,DSN=&CICSCLIB(FIVCA078) 00420001
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00430000

```

Figure B-14 TRACICS3 — Pre-process, assemble and linkedit the CICS application entry module AECICS (Page 1 of 2)

```

//          PEND                                00440000
//*                                                00450000
//*****                                          00460000
//*                                                00470000
//AECICS EXEC TRACICS3,                          00480002
//          WORKLIB='TRANSACT.TR078.WORKLIB',    00490001
//          CICSlib='TRANSACT.TR078.CICSlib',      00500001
//          CICSpref='CICSSYS.CTS220.CICS'         00510002
//                                                00520000

```

Figure B-14 TRACICS3 — Pre-process, assemble and linkedit the CICS application entry module AECICS (Page 2 of 2)

TRACICTM

```

//          JOB ...                            00010002
//*                                                00020002
//* MEMBER TRACICTM                            00030000
//*****                                          00040000
//*                                                00050000
//* THIS JOB ASSEMBLES AND LINK EDITS THE CICS EXECUTION-TIME TRACE 00060000
//* MESSAGES FOR VISION:TRANSACT.              00070000
//*                                                00080000
//*****                                          00090000
//*                                                00100000
//* ASSEMBLE THE CICS EXECUTION-TIME TRACE MESSAGE MODULE          00110000
//*                                                00120000
//ASM EXEC PGM=ASMA90,REGION=2M,                00130002
//          PARM='LIST,NODECK,OBJECT,RENT'        00140000
//SYSIN DD DISP=SHR,DSN=TRANSACT.TR078.WORKLIB(FIVCT078) 00150001
//SYSPRINT DD SYSOUT=*                          00160000
//SYSLIN DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(5,3)),      00170000
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)              00180000
//SYSLIB DD DISP=SHR,DSN=TRANSACT.TR078.WORKLIB 00190001
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00200000
//*                                                00210000
//* LINKEDIT THE CICS EXECUTION-TIME TRACE MESSAGE MODULE          00220000
//*                                                00230000
//LKED EXEC PGM=IEWL,PARM='LET,LIST,MAP,NCAL,RENT',REGION=2M, 00240000
//          COND=(8,LT,ASM)                          00250000
//SYSLIN DD DSN=*.ASM.SYSLIN,DISP=(OLD,DELETE)      00260000
//SYSPRINT DD SYSOUT=*                              00270000
//SYSMOD DD DISP=SHR,DSN=TRANSACT.TR078.CICSlib(FIVCT078) 00280001
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))          00290000
//                                                00300000

```

Figure B-15 TRACICTM — Assemble and linkedit the CICS runtime trace message module FIVCT078

TRASM1

```

//          JOB ...                            00010002
//*                                                00020002
//* MEMBER TRASM1                            00030000
//*****                                          00040000
//*                                                00050000
//* THIS PROCEDURE AND JOB STREAM ASSEMBLES AND LINKS THE          00060000
//* VISION:TRANSACT SYSTEM AND COMPONENT PARAMETER MODULES        00070000
//* FIVEPARM AND M5LEPARM.                                         00080000

```

Figure B-16 TRASM1 — Assemble and linkedit the Generator component parameter modules FIVEPARM and M5LEPARM (Page 1 of 2)

```

//* 00090000
//***** 00100000
//* 00110000
//ASMBPRM PROC WORKLIB=, 00120000
// MEMBER=, 00130000
// GENLIB= 00140000
//* 00150000
//ASM EXEC PGM=ASMA90,REGION=2M, 00160002
// PARM='LIST,NODECK,OBJECT' 00170000
//SYSIN DD DISP=SHR,DSN=&WORKLIB (&MEMBER) 00180000
//SYSPRINT DD SYSOUT=* 00190000
//SYSLIB DD DISP=SHR,DSN=&WORKLIB 00200000
// DD DISP=SHR,DSN=SYS1.MACLIB 00210000
//SYSLIN DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(3,1)), 00220000
// DCB=BLKSIZE=3200 00230000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00240000
//* 00250000
//LINK EXEC PGM=IEWL,REGION=2M, 00260000
// PARM='LET,LIST,MAP,NCAL' 00270000
//SYSLIN DD DSN=*.ASM.SYSLIN,DISP=(OLD,DELETE) 00280000
//SYSPRINT DD SYSOUT=* 00290000
//SYSLMOD DD DISP=SHR,DSN=&GENLIB (&MEMBER) 00300000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1)) 00310000
// PEND 00320000
//* 00330000
//***** 00340000
//* THE FOLLOWING IS A SAMPLE EXECUTION OF THE JOBS TO ASSEMBLE 00350000
//* AND LINK THE TRANSACT SYSTEM AND COMPONENT PARAMETER MODULES. 00360000
//* 00370000
//* BEFORE YOU RUN THIS JOB, SPECIFY: 00380000
//* 00390000
//* WORKLIB - THE SOURCE PDS THAT CONTAINS THE PARAMETER MODULE. 00400000
//* MEMBER - THE MEMBER NAME OF THE PARAMETER MODULE. 00410000
//* GENLIB - THE LOAD LIBRARY TO CONTAIN THE LINK EDITED MODULE. 00420000
//* 00430000
//***** 00440000
//* FIVEPARM ASSEMBLY AND LINK 00450000
//***** 00460000
//* 00470000
//FIVEPRM EXEC ASMBPRM, 00480000
// WORKLIB='TRANSACT.TR078.WORKLIB', 00490001
// MEMBER='FIVEPARM', 00500000
// GENLIB='TRANSACT.TR078.GENLIB' 00510001
//* 00520000
//***** 00530000
//* M5LEPARM ASSEMBLY AND LINK 00540000
//***** 00550000
//* 00560000
//M5LEPRM EXEC ASMBPRM, 00570000
// WORKLIB='TRANSACT.TR078.WORKLIB', 00580001
// MEMBER='M5LEPARM', 00590000
// GENLIB='TRANSACT.TR078.GENLIB' 00600001
// 00610002

```

Figure B-16 TRASM1 — Assemble and linkedit the Generator component parameter modules FIVEPARM and M5LEPARM (Page 2 of 2)

TRBMSE

```

// JOB ... 00010002
//* 00020002
//* MEMBER TRBMSE 00030000
//***** 00040000
//* THIS PROCEDURE AND JOBSTREAM REASSEMBLES THE BMS MAP FOR THE 00050000
//* CICS ERROR SCREEN 00060000
//***** 00070000
//* 00080000

```

Figure B-17 TRBMSE — Assemble and linkedit the BMS error display map FIVCE07 (Page 1 of 2)

```

//TRBMSE PROC GENLIB=,                                00090000
//                                                    00100000
//          WORKLIB=,                                00110000
//          CICSREF=,                                00120000
//          CICSCLIB=,                                00130000
//*                                                    00140000
//* ALLOCATE AND INITIALIZE A BDAM M5LIB              00150000
//*                                                    00160000
//INIT EXEC PGM=M5INIT,REGION=1M                      00170000
//STEPLIB DD DSN=&GENLIB,DISP=SHR                     00180000
//          DD DSN=&COMLIB,DISP=SHR                   00190000
//M5LIST DD SYSOUT=*                                  00200000
//M5LIB DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,5,,CONTIG) 00210000
//*                                                    00220000
//* CATALOG ERROR SCREEN DEFINITION AND GEN BMS      00230000
//*                                                    00240000
//DEF EXEC PGM=MARKV,REGION=1M,COND=(0,NE,INIT)       00250000
//STEPLIB DD DSN=&GENLIB,DISP=SHR                     00260000
//M5INPUT DD DSN=&WORKLIB(FIVCE07),DISP=SHR           00270000
//M5LIST DD SYSOUT=*                                  00280000
//M5LIB DD DSN=*.INIT.M5LIB,DISP=(OLD,DELETE)         00290000
//M5PUNCH DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(5,5)), 00300000
//          DCB=BLKSIZE=3200                          00310000
//*                                                    00320000
//* ASSEMBLE BMS MAP                                00330000
//*                                                    00340002
//ASM EXEC PGM=ASMA90,REGION=2M,COND=(0,NE,DEF),      00350000
//          PARM='OBJECT,NODECK,NORENT,NORLD,SYSPARM(MAP) '
//SYSIN DD DSN=*.DEF.M5PUNCH,DISP=(OLD,DELETE)       00360000
//SYSPRINT DD SYSOUT=*                                00370000
//SYSLIB DD DISP=SHR,DSN=&CICSREF..SDFHMAC            00380000
//          DD DISP=SHR,DSN=SYS1.MACLIB                00390000
//SYSLIN DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(TRK,(5,5)), 00400000
//          DCB=BLKSIZE=3200                          00410000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))              00420000
//*                                                    00430000
//* LINKEDIT BMS MAP                                00440000
//*                                                    00450000
//LINK EXEC PGM=IEWL,REGION=1M,COND=(16,LT,ASM),      00460000
//          PARM='LET,LIST,MAP'                        00470000
//SYSLIN DD DSN=*.ASM.SYSLIN,DISP=(OLD,DELETE)       00480000
//SYSPRINT DD SYSOUT=*                                00490000
//SYSLMOD DD DSN=&CICSCLIB(FIVCE07),DISP=SHR         00500000
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))              00510000
// PEND                                                00520000
//*                                                    00530000
//***** 00540000
//* 00550000
//* THE FOLLOWING IS A SAMPLE EXECUTION OF THE JOB TO ASSEMBLE AND 00560000
//* LINKEDIT THE BMS MAP FOR THE ERROR SCREEN. 00570000
//* BEFORE YOU RUN THIS JOB, SPECIFY: 00580000
//* GENLIB - THE LIBRARY CONTAINING THE TRANSACT GENERATOR 00590000
//* WORKLIB - THE PDS CONTAINING THE TRANSACT ERROR SCREEN SOURCE 00600000
//* CICSREF - THE CICS SYSTEM HIGH LEVEL INDEX 00610000
//* CICSCLIB - THE CICS EXECUTOR LOAD LIBRARY 00620000
//* 00630000
//***** 00640000
//* 00650000
//BMSE EXEC TRBMSE, 00660000
//          GENLIB='TRANSACT.TR078.GENLIB', 00670001
//          WORKLIB='TRANSACT.TR078.WORKLIB', 00680001
//          CICSREF='CICSSYS.CTS220.CICS', 00690002
//          CICSCLIB='TRANSACT.TR078.CICSCLIB' 00700001
// 00710000

```

Figure B-17 TRBMSE — Assemble and linkedit the BMS error display map FIVCE07
(Page 2 of 2)

TRCOPY1

```

//          JOB ...
//*
//* MEMBER TRCOPY1
//*****
//*
//* COPY THE GENERATOR AND EXECUTOR LIBRARIES TO OPERATIONAL DATASETS.
//*
//*****
//*
//* CHANGE THE MARKV PARAMETER BELOW TO MATCH THE VALUE ASSIGNED IN
//* JOB SMPJOB01.
//*
//* CHANGE THE NEWOP PARAMETER BELOW TO THE DESIRED HIGH LEVEL
//* QUALIFIER FOR YOUR GENERATOR AND EXECUTOR OPERATIONAL DATASETS.
//*
//*****
//*
//* SET MARKV=TRANSACTION.TR078  HIGH-LEVEL FOR YOUR SMP/E TARGET DSN'S
//* SET NEWOP=TRANSACTION.REL78  HIGH-LEVEL FOR YOUR OPERATIONAL DSN'S
//*
//*****
//*
//* CHANGE THE DUNIT PARAMETER IN THE INSTREAM PROC BELOW AS
//* APPROPRIATE FOR YOUR INSTALLATION.
//*
//*****
//*
//COPY      PROC MARKVDSN=,
//           NEWOPDSN=,
//           DUNIT=ISPDA,
//           DTRKS=,
//           DDCB=
//*
//DELETE EXEC PGM=IEFBR14
//DD1      DD DSN=&NEWOPDSN,DISP=(MOD,DELETE),
//          SPACE=(TRK,(0,0)),UNIT=&DUNIT
//*
//COPY      EXEC PGM=IEBCOPY,REGION=2M
//SYSPRINT DD SYSOUT=*
//IN        DD DSN=&MARKVDSN,DISP=SHR
//OUT        DD DSN=&NEWOPDSN,DISP=(NEW,CATLG),
//           UNIT=&DUNIT,SPACE=(TRK,&DTRKS),
//           DCB=&DDCB
//SYSUT3     DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4     DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//*
//          PEND
//*
//*****
//*
//* THE FOLLOWING JCL EXECUTES THE STEPS NEEDED TO TRANSFER THE
//* VISION:TRANSACTION SYSTEM TAPE FILE TO DISK DATASETS.
//*
//* BEFORE YOU RUN THIS JOB, SPECIFY:
//*
//* MARKVDSN - DSN OF THE SMP/E TARGET LIBRARY.
//* NEWOPDSN - DSN OF THE NEW OPERATIONAL LIBRARY.
//* DUNIT     - UNIT TYPE FOR THE DASD.  THE DEFAULT IS SYSDA.
//* DTRKS     - SPACE ALLOCATION FOR THE DASD DATASET.  THE SAMPLES
//*             SHOW MINIMUM REQUIREMENTS FOR 3390 DEVICES.
//* DDCB      - DASD DATASET DCB VALUES.  THE BLKSIZE VALUES FOR
//*             THE LOAD LIBRARIES CAN BE CHANGED IF NEEDED.
//* NOTE      - THE DISK DATASETS ARE ALLOCATED HERE
//*             WITH "DISP=(NEW,CATLG)".
//*
//*****
//*
//* STEP1 - COPY THE GENERATOR LOAD LIBRARY
//*****
//STEP1     EXEC COPY,

```

Figure B-18 TRCOPY1 — Copy the Generator and Executor operational libraries to new data sets (Page 1 of 2)

```

//          NEWOPDSN=&NEWOP..GENLIB,
//          DTRKS='(180,30,50)',
//          DDCB='(RECFM=U,LRECL=0,BLKSIZE=27998)',
//          MARKVDSN=&MARKV..TRTGEN
//COPY.SYSIN DD *
//          COPYMOD INDD=IN,OUTDD=OUT
//*
//*****
//*
//* STEP2 - COPY THE EXECUTOR LOAD LIBRARY
//*****
//STEP2 EXEC COPY,
//          NEWOPDSN=&NEWOP..CICSLIB,
//          DTRKS='(15,5,5)',
//          DDCB='(RECFM=U,LRECL=0,BLKSIZE=27998)',
//          MARKVDSN=&MARKV..TRTCICS
//COPY.SYSIN DD *
//          COPYMOD INDD=IN,OUTDD=OUT
//*
//*****
//*
//* STEP3 - COPY THE TOOLKIT LOAD LIBRARY
//*****
//STEP3 EXEC COPY,
//          NEWOPDSN=&NEWOP..CICSTKL,
//          DTRKS='(15,5,10)',
//          DDCB='(RECFM=U,LRECL=0,BLKSIZE=27998)',
//          MARKVDSN=&MARKV..TRTTKCL
//COPY.SYSIN DD *
//          COPYMOD INDD=IN,OUTDD=OUT
//

```

Figure B-18 TRCOPY1 — Copy the Generator and Executor operational libraries to new data sets (Page 2 of 2)

TRCOPY2

```

//          JOB ...
//*
//* MEMBER TRCOPY2
//*****
//* COPY THE GENERATOR AND EXECUTOR LIBRARIES TO OPERATIONAL DATASETS.
//*
//*****
//* CHANGE THE MARKV PARAMETER BELOW TO MATCH THE VALUE ASSIGNED IN
//* JOB SMPJOB01.
//*
//* CHANGE THE NEWOP PARAMETER BELOW TO THE DESIRED HIGH LEVEL
//* QUALIFIER FOR YOUR GENERATOR AND EXECUTOR OPERATIONAL DATASETS.
//*
//*****
//*
// SET MARKV=TRANSACT.TR078 HIGH-LEVEL FOR YOUR SMP/E TARGET DSN'S
// SET NEWOP=TRANSACT.REL78 HIGH-LEVEL FOR YOUR OPERATIONAL DSN'S
//*****
//*
//COPY PROC MARKVDSN=,
//          NEWOPDSN=
//*
//COPY EXEC PGM=IEBCOPY,REGION=2M
//SYSPRINT DD SYSOUT=*
//IN DD DSN=&MARKVDSN,DISP=SHR
//OUT DD DSN=&NEWOPDSN,DISP=OLD
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))

```

Figure B-19 TRCOPY2 — Copy the Generator and Executor operational libraries to existing data sets (Page 1 of 2)

```

//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//*
// PEND
//*
//*****
//* THE FOLLOWING JCL EXECUTES THE STEPS NEEDED TO TRANSFER THE
//* VISION:TRANSACTION GENERATOR AND EXECUTOR OPERATIONAL DATASETS.
//*
//*****
//* STEP1 - COPY THE GENERATOR LOAD LIBRARY
//*****
//STEP1 EXEC COPY,
// NEWOPDSN=&NEWOP.GENLIB,
// MARKVDSN=&MARKV..TRTGEN
//COPY.SYSIN DD *
// COPYMOD INDD=( (IN,R) ),OUTDD=OUT
// COPY INDD=OUT,OUTDD=OUT
//*
//*****
//* STEP2 - COPY THE EXECUTOR LOAD LIBRARY
//*****
//STEP2 EXEC COPY,
// NEWOPDSN=&NEWOP..CICSLIB,
// MARKVDSN=&MARKV..TRTCICS
//COPY.SYSIN DD *
// COPYMOD INDD=( (IN,R) ),OUTDD=OUT
// COPY INDD=OUT,OUTDD=OUT
//*
//*****
//* STEP3 - COPY THE TOOLKIT LOAD LIBRARY
//*****
//STEP3 EXEC COPY,
// NEWOPDSN=&NEWOP..CICSTKL,
// MARKVDSN=&MARKV..TRTTKCL
//COPY.SYSIN DD *
// COPYMOD INDD=( (IN,R) ),OUTDD=OUT
// COPY INDD=OUT,OUTDD=OUT
//

```

Figure B-19 TRCOPY2 — Copy the Generator and Executor operational libraries to existing data sets (Page 2 of 2)

TRCSDUP

```

// JOB ... 00010002
//* 00020002
//* MEMBER TRCSDUP 00030000
//***** 00040000
//* * 00050002
//* THIS JOB INSTALLS THE VISION:TRANSACTION SYSTEM DEFINITIONS FOR * 00060002
//* CICS TRANSACTION SERVER VERSION 1.3 OR HIGHER. * 00070002
//* * 00080002
//***** 00090000
//* 00100000
//INST PROC CICSPREF=, 00110000
// CSDFILE= 00120000
//* 00130000
//CSDUP EXEC PGM=DFHCSDUP,REGION=1M 00140000
//STEPLIB DD DISP=SHR,DSN=&CICSPREF..SDFHLOAD 00150000
//DFHCSD DD DISP=SHR,DSN=&CSDFILE 00160000
//SYSPRINT DD SYSOUT=* 00170000
// PEND 00180000
//* 00190000

```

Figure B-20 TRCSDUP — Define the Executor program elements to CICS (Page 1 of 2)

```

//***** 00200000
/* THE FOLLOWING IS A SAMPLE EXECUTION OF THIS PROCEDURE. * 00210000
/* BEFORE YOU RUN THIS PROCEDURE, SPECIFY: * 00220000
/* * 00230000
/* CICSREF - PREFIX FOR THE CICS 'SDFHLOAD' LOAD LIBRARY * 00240005
/* CONTAINING THE DFHCSDUP PROGRAM. * 00250005
/* CSDFILE - THE CICS SYSTEM DEFINITION FILE DATASET NAME. * 00260005
/* * 00270000
/* NOTE: THE CSD ENTRIES MAY BE ALTERED AS FOLLOWS: * 00280000
/* * 00290000
/* THE GROUP AND LIST NAMES MAY BE CHANGED AS NEEDED. * 00300003
/* THE GROUP NAME MUST BE THE SAME IN ALL DEFINITIONS. * 00310000
/* THE DEFAULT GROUPNAME IS TRANSACT. * 00320000
/* * 00330000
//***** 00340000
/* 00350000
//INST EXEC INST, 00360000
// CICSREF='CICSSYS.CTS220.CICS', 00370002
// CSDFILE='DYLCICS.CTS220.CSD' 00380001
//SYSIN DD * 00390000
***** 00400000
* DEFINE THE GROUP AND LIST ENTRIES. * 00410000
***** 00420000
ADD GROUP(TRAN078) LIST(TRANLIST) 00430001
***** 00440000
* DEFINE THE MAPSET ENTRIES. * 00450000
***** 00460000
DEFINE MAPSET(FIVCE07) GROUP(TRAN078) 00470001
***** 00480000
* DEFINE THE PROGRAM ENTRIES. * 00490000
***** 00500000
DEFINE PROGRAM(FIVCS078) GROUP(TRAN078) LANGUAGE(ASSEMBLER) 00510001
DEFINE PROGRAM(FIVCI078) GROUP(TRAN078) LANGUAGE(LE370) 00520001
DEFINE PROGRAM(FIVCJ078) GROUP(TRAN078) LANGUAGE(LE370) 00530002
DEFINE PROGRAM(FIVCM078) GROUP(TRAN078) LANGUAGE(ASSEMBLER) 00540001
DEFINE PROGRAM(FIVCT078) GROUP(TRAN078) LANGUAGE(ASSEMBLER) 00550001
***** 00560002
* DEFINE THE TRANSIENT DATA TRACE OUTPUT DESTINATION ENTRY * 00570002
***** 00580002
DEFINE TDQUEUE(M5ET) TYPE(EXTRA) DDNAME(M5ETOUT) TYPEFILE(OUTPUT) *00590002
RECORDSIZE(136) RECORDFORMAT(VARIABLE) BLOCKFORMAT(BLOCKED) *00600004
GROUP(TRAN078) 00610004
// 00620000

```

Figure B-20 TRCSDUP — Define the Executor program elements to CICS (Page 2 of 2)

TRDB2A

```

// JOB ... 00010002
/* 00020002
/* MEMBER TRDB2A 00030000
//***** 00040000
/* THIS FOLLOWING SAMPLE JOB STREAM PREPARES, ASSEMBLES AND LINKS 00050000
/* THE "FIVESQL" MODULE. 00060000
/* 00070000
/* NOTE: THE STANDARD IBM PROCEDURE "DSNHASM" IS USED TO ACCOMPLISH 00080002
/* THE DB2 PREPARATION PROCESS. YOU MAY NEED TO CONFER WITH 00090000
/* YOUR DB2 DATA BASE ADMINISTRATOR BEFORE RUNNING THIS JOB. 00100000
/* 00110000
/* BEFORE YOU RUN THESE JOBS, 00120000
/* REVIEW THE JCL STATEMENTS FLAGGED WITH "<<<<" AND SPECIFY: 00130000
/* 00140000
/* SQLPREP - THE "DB2" ENTRY IS HIGH LEVEL QUALIFIER FOR DB2 00150002
/* DATASETS. 00160002
/* - THE "MEM" ENTRY IS THE PLAN NAME FOR THE 00170002
/* DB2 RUN MODULE. 00180000

```

Figure B-21 TRDB2A — Assemble and linkedit the Generator DB2 Interface module FIVESQL (Page 1 of 2)

```

/*          - THE "USER" ENTRY IS THE AUTHORIZED USERID.          00190000
/*          THIS NAME IS USED BY THE IBM PROCEDURE "DSNHASHM"      00200000
/*          TO CONSTRUCT DEFAULT DATASET NAMES.                   00210000
/*  SYSIN    - THE DATASET/MEMBER FOR THE "FIVESQL" SOURCE CODE.   00220000
/*  SYSLOAD  - THE LOAD LIBRARY FOR THE LINKED DB2 RUN MODULES.    00230000
/*          00240000
/*          *****00250002
/*          00260002
/*          INSTREAM PROCEDURE TO PRE-COMPILE, ASSEMBLE, AND LINKEDIT 00270002
/*          A HIGH LEVEL ASSEMBLER PROGRAM                          00280002
/*          00290002
/*          *****00300002
/*          00310002
/*          PRE-COMPILE THE ASSEMBLER PROGRAM                      00320002
/*          00330002
/*          DSNHASHM PROC WSPC=500, MEM=TEMPNAME, USER=USER        00340002
/*          PC EXEC PGM=DSNHPC, PARM='HOST (ASM), STDSQL (YES)', REGION=4096K 00350002
/*          DBRMLIB DD DSN=&USER..DBRMLIB.DATA (&MEM),             00360002
/*          DISP=SHR                                               00370002
/*          STEPLIB DD DISP=SHR, DSN=&DB2..SDSNEXIT                 00380002
/*          DD DISP=SHR, DSN=&DB2..SDSNLOAD                         00390002
/*          SYSCIN DD DSN=&DSNHOUT, DISP=(MOD,PASS), UNIT=SYSDA,    00400002
/*          SPACE=(800, (&WSPC, &WSPC))                          00410002
/*          SYSLIB DD DSN=&USER..SRCLIB.DATA,                       00420002
/*          DISP=SHR                                              00430002
/*          SYSPRINT DD SYSOUT=*                                   00440002
/*          SYSTEM DD SYSOUT=*                                    00450002
/*          SYSUT1 DD SPACE=(800, (&WSPC, &WSPC),,ROUND), UNIT=SYSDA 00460002
/*          00470002
/*          ASSEMBLE IF THE PRECOMPILE RETURN CODE                00480002
/*          IS 4 OR LESS                                          00490002
/*          00500002
/*          ASM EXEC PGM=ASMA90, PARM='OBJECT,NODECK', COND=(4,LT,PC) 00510002
/*          SYSIN DD DSN=&DSNHOUT, DISP=(OLD,DELETE)               00520002
/*          SYSLIB DD DSN=SYS1.MACLIB, DISP=SHR                   00530002
/*          SYSLIN DD DSN=&LOADSET, DISP=(MOD,PASS), UNIT=SYSDA,   00540002
/*          SPACE=(800, (&WSPC, &WSPC)), DCB=(BLKSIZE=800)       00550002
/*          SYSPRINT DD SYSOUT=*                                  00560002
/*          SYSUT1 DD SPACE=(800, (&WSPC, &WSPC),,ROUND), UNIT=SYSDA 00570002
/*          00580002
/*          LINKEDIT IF THE PRECOMPILER AND ASSEMBLER             00590002
/*          RETURN CODES ARE 4 OR LESS                            00600002
/*          00610002
/*          LKED EXEC PGM=HEWL, PARM='XREF',                       00620002
/*          COND=( (4,LT,ASM), (4,LT,PC) )                       00630002
/*          SYSLIB DD DISP=SHR, DSN=DB2.DB2610.SDSNLOAD            00640002
/*          DD DISP=SHR, DSN=IMSSYS61.RESLIB                      00650002
/*          DD DISP=SHR, DSN=CICSSYS.CTS220.CICS.SDFHLOAD         00660002
/*          DD DISP=SHR, DSN=ISP.SISPLOAD                         00670002
/*          SYSLIN DD DSN=&LOADSET, DISP=(OLD,DELETE)              00680002
/*          DD DDNAME=SYSIN                                       00690002
/*          SYSLOAD DD DSN=&USER..RUNLIB.LOAD (&MEM),              00700002
/*          DISP=SHR                                              00710002
/*          SYSPRINT DD SYSOUT=*                                   00720002
/*          SYSUT1 DD SPACE=(1024, (50,50)), UNIT=SYSDA           00730002
/*          DSNHASHM PEND                                         00740002
/*          00750002
/*          *****00760000
/*          00770002
/*          PREPARE THE DB2 TSO ATTACH RUN MODULE.                 00780000
/*          00790002
/*          *****00800000
/*          00810000
/*          SQLPREP EXEC DSNHASHM, DB2=DB2.DB2710,                <<<< 00820002
/*          MEM=FIVESQL, USER=PUBLIC,                             <<<< 00830002
/*          PARM.PC='HOST (ASM), STDSQL (86)'                     00840000
/*          PC.SYSIN DD DSN=TRANSACTION.TR078.WORKLIB (FIVESQL), DISP=SHR <<<< 00850001
/*          LKED.SYSLOAD DD DSN=TRANSACTION.TR078.GENLIB, DISP=OLD <<<< 00860001
/*          LKED.SYSIN DD *                                         00870000
/*          INCLUDE SYSLIB (DSNELI)                                00880000
/*          NAME FIVESQL (R)                                       00890000
/*          00900000

```

Figure B-21 TRDB2A — Assemble and linkedit the Generator DB2 Interface module FIVESQL (Page 2 of 2)

TRDB2B

```

//          JOB ...                                00010002
//*                                                00020002
//* MEMBER TRDB2B                                00030000
//*****00040000
//* THE FOLLOWING SAMPLE JOB STREAM DOES THE DB2 BIND FOR THE 00050000
//* "FIVESQL" MODULE THAT IS USED FOR ATTACHING TO DB2      00060000
//* DURING GENERATOR RUNS.                                00070000
//*                                                        00080000
//*                                                        00090000
//* BEFORE RUNNING THIS JOB,                            00100000
//* CONFER WITH YOUR DB2 DATA BASE ADMINISTRATOR AND      00110000
//* REVIEW THE VALUES FOR THE FOLLOWING ENTRIES:          00120000
//*                                                        00130000
//* SYSTEM - THE DB2 SUBSYSTEM NAME.                      00140000
//* PLAN - THE PLAN NAME. THE DEFAULT IS SHOWN.           00150000
//* LIBRARY - THE DBRM LIBRARY NAME.                      00160000
//* MEMBER - THE PLAN MEMBER NAME. THE SAME AS THE PLAN NAME. 00170000
//*                                                        00180000
//*****00190000
//* BIND THE DB2 TSO ATTACH PLAN                        00200000
//*****00210000
//*                                                        00220000
//BINDC EXEC PGM=IKJEFT01,DYNAMNBR=20,REGION=1M          00230000
//SYSTSPT DD SYSOUT=*                                    00240000
//SYSTSIN DD *                                           00250000
DSN SYSTEM(SYSTEM)                                       00260000
  BIND PLAN (FIVESQL) -                                  00270000
    LIBRARY ('PUBLIC.DBRMLIB.DATA') -                    00280000
    MEMBER (FIVESQL) -                                   00290000
    ACTION (REPLACE) RETAIN -                             00300000
    ISOLATION (CS)                                       00310000
END                                                       00320000
//                                                       00330000

```

Figure B-22 TRDB2B — Bind the Generator DB2 plan FIVESQL

TRIVP

```

//          JOB ...                                00010002
//*                                                00020002
//* MEMBER TRIVP                                00030000
//*****00040000
//* THIS JOB STREAM IS USED TO DEMONSTRATE TO THE INSTALLER THAT 00050000
//* THE VISION:TRANSACT GENERATOR INSTALLATION WAS SUCCESSFUL. 00060000
//* SEVERAL DIFFERENT JOB STEPS ARE RUN TO PERFORM A VARIETY OF 00070000
//* FUNCTIONS THAT MAKE SURE THAT THE PRODUCT IS OPERATIONAL.  00080000
//*                                                        00090000
//* BEFORE YOU RUN THIS JOB, SPECIFY ON THE "JOBLIB" DD      00100000
//*                                                        00110000
//* THE VISION:TRANSACT AND COMLIB COMPONENT LOAD LIBRARIES. 00120000
//*                                                        00130000
//*****00140000
//*                                                        00150000
//JOBLIB DD DSN=TRANSACT.TR078.GENLIB,DISP=SHR           00160002
//*                                                        00170000
//* INITIALIZE AN M5LIB                                    00180000
//*                                                        00190000
//INIT EXEC PGM=M5INIT,REGION=1M                         00200000
//M5LIST DD SYSOUT=*                                     00210000
//M5LIB DD DSN=&&TEMPLIB,DISP=(,PASS),                    00220000
// SPACE=(TRK,2,,CONTIG),UNIT=SYSDA                     00230000
//*                                                        00240000

```

Figure B-23 TRIVP — Verify that the Generator component install was successful
(Page 1 of 3)

```

/* CATALOG SCREEN AND FILE DEFINITIONS FOR CICS                                00250000
/* FOR IMS, CHANGE 'C' IN COLUMN 39 OF RC STATEMENT TO 'I'                    00260001
/*                                                                              00270000
//DEF      EXEC PGM=MARKV,REGION=2M                                           00280000
//M5LIB     DD DSN=&&TEMPLIB,DISP=(OLD,PASS)                                   00290000
//M5LIST     DD SYSOUT=*                                                       00300000
//M5PUNCH    DD DUMMY                                                         00310000
//M5INPUT    DD *                                                             00320000
IVP1       RC          #          DC                                         00330000
IVPSCN1    SDR @ IVP1                                                         00340000
IVPSCN1    SL 30 F.TODAY                                                         00350000
IVPSCN1    SL 40 F.TODAYX _____ Input Date: _____ 00360000
IVPSCN1    SL 41 _____ Output Date: _____ 00370000
IVPSCN1    SL 50 F.ISDATE _____ 00380000
IVPSCN1    SL 60 F.DATE _____ 00390000
IVPSCN1    SL 70 F.JULIAN _____ 00400000
IVPSCN1    SL 80 F.JULANX _____ 00410000
IVPSCN1    SL 90 F.LILIAN _____ 00420000
IVPSCN1    SS 1 1 _____ 00430000
IVPSCN1    S0 1 FTODAY O NM TODAY 00440000
IVPSCN1    S0 1 TINDATE I Y P NM INDATE 00450000
IVPSCN1    S0 1 FTODAYX O NM TODAYX 00460000
IVPSCN1    S0 1 TINDATE O NM OUTDATE 00470000
IVPSCN1    S1 _____ ED Wwwwwww, Mmm DD, YYYY 00480000
IVPSCN1    S0 1 FISDATE O NM ISDATE 00490000
IVPSCN1    S0 1 FDATE O NM DATE 00500000
IVPSCN1    S0 1 FJULIAN O NM JULIAN 00510000
IVPSCN1    S0 1 FJULANX O NM JULANX 00520000
IVPSCN1    S0 1 FLILIAN O NM LILIAN 00530000
MASTER    FDDBDNAME R D 80 00540000
MASTER    LSSEGS10 10 00550000
MASTER    L0CUSTNUM 101 1 6C1 00560000
MASTER    L1CUSTNUM _____ CUSTOMER 00570000
MASTER    L2CUSTNUM _____ NUMBER 00580000
MASTER    L0CUSTNAME 101 7 20C 00590000
MASTER    L1CUSTNAME _____ CUSTOMER 00600000
MASTER    L2CUSTNAME _____ NAME 00610000
MASTER    L0CINDSTRE 101 27 15C 00620000
MASTER    L1CINDSTRE _____ CUSTOMER 00630000
MASTER    L2CINDSTRE _____ INDUSTRY 00640000
MASTER    L0CSTCODE 101 42 2C 00650000
MASTER    L1CSTCODE _____ CUSTOMER 00660000
MASTER    L2CSTCODE _____ STATE CODE 00670000
/*                                                                              00680000
/* GENERATE APPLICATION FOR CICS                                              00690000
/* FOR IMS, CHANGE 'C' IN COLUMN 39 OF RC STATEMENT TO 'I'                    00700001
/*                                                                              00710000
//GEN      EXEC PGM=MARKV,REGION=2M                                           00720000
//M5LIB     DD DSN=&&TEMPLIB,DISP=(OLD,PASS)                                   00730000
//M5LIST     DD SYSOUT=*                                                       00740000
//M5PUNCH    DD DUMMY                                                         00750000
//M5INPUT    DD *                                                             00760000
IVP1       RC          #          GC                                         00770000
IVPSCN1    SC          OS          IVPSCN1 N 00780000
TEMPS      ER          N 00790000
##PROC                                           00800000
INDATE:    FIELD D 4 00810000
##PEND                                           00820000
/*                                                                              00830000
/* DUMP, INIT, RESTORE AN M5LIB 00840000
/*                                                                              00850000
//DUMP     EXEC PGM=M5UTIL,REGION=2M                                           00860000
//M5LIB     DD DSN=&&TEMPLIB,DISP=(OLD,PASS)                                   00870000
//M5LIST     DD SYSOUT=*                                                       00880000
//M5WORK     DD SPACE=(CYL,(1,1)),UNIT=SYSDA 00890000
//M5INPUT    DD * 00900000
UCDUMP                                           00910000
UCINIT                                           00920000
UCREST                                           00930000
/*                                                                              00940000
/* LIST FILE & SCREEN DEFINITION GLOSSARIES 00950000
/* FOR IMS, CHANGE 'C' IN COLUMN 39 OF RC STATEMENT TO 'I' 00960001
/*                                                                              00970000

```

Figure B-23 TRIVP — Verify that the Generator component install was successful
(Page 2 of 3)

```

//LIST      EXEC PGM=MARKV,REGION=2M                                00980000
//M5LIB      DD DSN=&TEMPLIB,DISP=(OLD,DELETE)                      00990000
//M5LIST      DD SYSOUT=*                                           01000000
//M5PUNCH     DD DUMMY                                              01010000
//M5INPUT     DD *                                                  01020000
LISTGLOSRC      #          DC                                       01030000
                CTLFG                                              01040000
                CTLSG                                              01050000
//                                                    01060000

```

Figure B-23 TRIVP — Verify that the Generator component install was successful
(Page 3 of 3)

WBCOPY1

```

//          JOB ...
//*
//* MEMBER WBCOPY1
//*****
//*
//* COPY THE SMP/E TARGET LIBRARIES FOR THE ISPF WORKBENCH TO
//* OPERATIONAL DATASETS.
//*
//*****
//*
//* CHANGE THE BENCH PARAMETER BELOW TO MATCH THE VALUE ASSIGNED IN
//* JOB SMPJOB01.
//*
//* CHANGE THE NEWOP PARAMETER BELOW TO THE DESIRED HIGH LEVEL
//* QUALIFIER FOR YOUR ISPF WORKBENCH OPERATIONAL DATASETS.
//*
//*****
//*
//* SET BENCH=TRANSACTION.WB061 HIGH-LEVEL FOR SMP/E TARGET DSN'S
//* SET NEWOP=TRANSACTION.REL78 HIGH-LEVEL FOR OPERATIONAL DSN'S
//*
//*****
//*
//* CHANGE THE DUNIT PARAMETER IN THE INSTREAM PROC BELOW AS
//* APPROPRIATE FOR YOUR INSTALLATION.
//*
//*****
//*
//COPY      PROC BENCHDSN=,
//            NEWOPDSN=,
//            DUNIT=ISPDA,
//            DTRKS=,
//            DDCB=
//*
//DELETE EXEC PGM=IEFBR14
//DD1      DD DSN=&NEWOPDSN,DISP=(MOD,DELETE),
//            SPACE=(TRK,(0,0)),UNIT=&DUNIT
//*
//COPY EXEC PGM=IEBCOPY,REGION=2M
//SYSPRINT DD SYSOUT=*
//IN      DD DSN=&BENCHDSN,DISP=SHR
//OUT     DD DSN=&NEWOPDSN,DISP=(NEW,CATLG),
//            UNIT=&DUNIT,SPACE=(TRK,&DTRKS),
//            DCB=&DDCB
//SYSUT3  DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4  DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//*
//          PEND
//*
//*****
//* THE FOLLOWING JCL EXECUTES THE STEPS NEEDED TO TRANSFER THE

```

Figure B-24 WBCOPY1 — Copy the ISPF Workbench operational libraries to new data sets (Page 1 of 2)

```

/* VISION:TRANSACTION SYSTEM TAPE FILE TO DISK DATASETS.
/*
/* BEFORE YOU RUN THIS JOB, SPECIFY:
/*
/* BENCHDSN - DSN OF THE INPUT SMPE/E TARGET LIBRARY.
/* NEWOPDSN - DSN OF THE NEW OPERATIONAL LIBRARY.
/* DUNIT - UNIT TYPE FOR THE DASD. THE DEFAULT IS SYSDA.
/* DTRKS - SPACE ALLOCATION FOR THE DASD DATASET. THE SAMPLES
/* SHOW MINIMUM REQUIREMENTS FOR 3390 DEVICES.
/* DDCB - DASD DATASET DCB VALUES. THE BLKSIZE VALUES FOR
/* THE LOAD LIBRARIES CAN BE CHANGED IF NEEDED.
/* NOTE - THE DISK DATASETS ARE ALLOCATED HERE
/* WITH "DISP=(NEW,CATLG)".
/*
/* *****
/*
/* STEP1 - COPY THE ISPF WORKBENCH LOAD LIBRARY
/* *****
/*STEP1 EXEC COPY,
/* NEWOPDSN=&NEWOP..LOADLIB,
/* DTRKS=' (60,30,40) ',
/* DDCB=' (RECFM=U,LRECL=0,BLKSIZE=27998) ',
/* BENCHDSN=&BENCH..WBTLOAD
/*COPY.SYSIN DD *
/* COPYMOD INDD=IN,OUTDD=OUT
/*
/* *****
/*
/* STEP2 - COPY THE ISPF WORKBENCH PANEL LIBRARY
/* *****
/*STEP2 EXEC COPY,
/* NEWOPDSN=&NEWOP..PANELS,
/* DTRKS=' (120,30,250) ',
/* DDCB=' (RECFM=FB,LRECL=80,BLKSIZE=27920) ',
/* BENCHDSN=&BENCH..WBTPANEL
/*COPY.SYSIN DD *
/* COPY INDD=IN,OUTDD=OUT
/*
/* *****
/*
/* STEP3 - COPY THE ISPF WORKBENCH MESSAGE LIBRARY
/* *****
/*STEP3 EXEC COPY,
/* NEWOPDSN=&NEWOP..MSGGS,
/* DTRKS=' (15,5,40) ',
/* DDCB=' (RECFM=FB,LRECL=80,BLKSIZE=27920) ',
/* BENCHDSN=&BENCH..WBTMSGGS
/*COPY.SYSIN DD *
/* COPY INDD=IN,OUTDD=OUT
/*
/* *****
/*
/* STEP4 - COPY THE ISPF WORKBENCH SKELETON LIBRARY
/* *****
/*STEP4 EXEC COPY,
/* NEWOPDSN=&NEWOP..SKELS,
/* DTRKS=' (10,5,5) ',
/* DDCB=' (RECFM=FB,LRECL=80,BLKSIZE=27920) ',
/* BENCHDSN=&BENCH..WBTSKELS
/*COPY.SYSIN DD *
/* COPY INDD=IN,OUTDD=OUT
/*
/* *****
/*
/* STEP5 - COPY THE ISPF WORKBENCH CLIST LIBRARY
/* *****
/*STEP5 EXEC COPY,
/* NEWOPDSN=&NEWOP..CLIST,
/* DTRKS=' (20,5,10) ',
/* DDCB=' (RECFM=FB,LRECL=80,BLKSIZE=27920) ',
/* BENCHDSN=&BENCH..WBTCLIST
/*COPY.SYSIN DD *
/* COPY INDD=IN,OUTDD=OUT
/*

```

Figure B-24 WBCOPY1 — Copy the ISPF Workbench operational libraries to new data sets (Page 2 of 2)

WBCOPY2

```

//          JOB ...
//*
//* MEMBER WBCOPY2
//*****
//*
//* COPY THE SMP/E TARGET LIBRARIES FOR THE ISPF WORKBENCH TO
//* OPERATIONAL DATASETS.
//*
//*****
//*
//* CHANGE THE BENCH PARAMETER BELOW TO MATCH THE VALUE ASSIGNED IN
//* JOB SMPJOB01.
//*
//* CHANGE THE NEWOP PARAMETER BELOW TO THE DESIRED HIGH LEVEL
//* QUALIFIER FOR YOUR ISPF WORKBENCH OPERATIONAL DATASETS.
//*
//*****
//*
//* SET BENCH=TRANSACTION.WB061  HIGH-LEVEL FOR SMP/E TARGET DSN'S
//* SET NEWOP=TRANSACTION.REL78  HIGH-LEVEL FOR OPERATIONAL DSN'S
//*
//*****
//*
//* CHANGE THE DUNIT PARAMETER IN THE INSTREAM PROC BELOW AS
//* APPROPRIATE FOR YOUR INSTALLATION.
//*
//*****
//*
//COPY      PROC BENCHDSN=,
//           NEWOPDSN=
//*
//COPY      EXEC PGM=IEBCOPY,REGION=2M
//SYSPRINT  DD SYSOUT=*
//IN        DD DSN=&BENCHDSN,DISP=SHR
//OUT       DD DSN=&NEWOPDSN,DISP=OLD
//SYSUT3    DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4    DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//*
//          PEND
//*
//*****
//*
//* THE FOLLOWING JCL EXECUTES THE STEPS NEEDED TO TRANSFER THE
//* VISION:TRANSACTION SYSTEM TAPE FILE TO DISK DATASETS.
//*
//* BEFORE YOU RUN THIS JOB, SPECIFY:
//*
//*      NEWOPDSN - DSN OF THE NEW OPERATIONAL LIBRARY.
//*      BENCHDSN - DSN OF THE INPUT SMP/E TARGET LIBRARY.
//*
//*****
//*
//* STEP1 - COPY THE ISPF WORKBENCH LOAD LIBRARY
//*****
//STEP1     EXEC COPY,
//           NEWOPDSN=&NEWOP..LOADLIB,
//           BENCHDSN=&BENCH..WBTLLOAD
//COPY.SYSIN DD *
//COPYMOD INDD=( (IN,R) ),OUTDD=OUT
//COPY      INDD=OUT,OUTDD=OUT
//*
//*****
//*
//* STEP2 - COPY THE ISPF WORKBENCH PANEL LIBRARY
//*****
//STEP2     EXEC COPY,
//           NEWOPDSN=&NEWOP..PANELS,
//           BENCHDSN=&BENCH..WBTPANEL
//COPY.SYSIN DD *
//COPY      INDD=( (IN,R) ),OUTDD=OUT

```

Figure B-25 WBCOPY2 — Copy the ISPF Workbench operational libraries to existing data sets (Page 1 of 2)

```

COPY INDD=OUT,OUTDD=OUT
/*
/******
/*
/** STEP3 - COPY THE ISPF WORKBENCH MESSAGE LIBRARY
/******
//STEP3 EXEC COPY,
//          NEWOPDSN=&NEWOP..MSGGS,
//          BENCHDSN=&BENCH..WBTMSGGS
//COPY.SYSIN DD *
//COPY INDD=( (IN,R) ),OUTDD=OUT
//COPY INDD=OUT,OUTDD=OUT
/*
/******
/*
/** STEP4 - COPY THE ISPF WORKBENCH SKELETON LIBRARY
/******
//STEP4 EXEC COPY,
//          NEWOPDSN=&NEWOP..SKELS,
//          BENCHDSN=&BENCH..WBTSKELS
//COPY.SYSIN DD *
//COPY INDD=( (IN,R) ),OUTDD=OUT
//COPY INDD=OUT,OUTDD=OUT
/*
/******
/*
/** STEP5 - COPY THE ISPF WORKBENCH CLIST LIBRARY
/******
//STEP5 EXEC COPY,
//          NEWOPDSN=&NEWOP..CLIST,
//          BENCHDSN=&BENCH..WBTCLIST
//COPY.SYSIN DD *
//COPY INDD=( (IN,R) ),OUTDD=OUT
//COPY INDD=OUT,OUTDD=OUT
//

```

Figure B-25 WBCOPY2 — Copy the ISPF Workbench operational libraries to existing data sets (Page 2 of 2)

VISION:Transact Parameter Modules

This appendix lists the Parameter Module source code members in the TRANSACT.TR078.WORKLIB data set. These members may be modified to customize/tailor the product to your installation's particular preferences or requirements.

Note: Other non-parameter module members in the WORKLIB data set contain JCL and other types of code.

Member	Description
CEEUOPT	The Executor component LE user options/parameters
FIVEPARM	The Generator component installation parameters
MARKLIBP	The Generator component COMLIB installation parameters
M5LEPARM	The Generator component LE options/parameters

Members CEEUOPT and M5LEPARM are used to specify the IBM Language Environment user options and/or parameters for the Executor and Generator components, respectively. A proper understanding of IBM Language Environment options and/or parameters is a pre-requisite to making changes to these members. In particular, inappropriate changes to the CEEUOPT member may have a negative impact upon the operation and performance of VISION:Transact applications programs under CICS.

Members FIVEPARM and MARKLIBP are used to specify installation parameters for the Generator component. Additional descriptive information on each of the parameter values for the FIVEPARM and MARKLIBP members is included with the parameter module listing.

CEEUOPT

```

*/*****/ 00010000
*/          */ 00020000
*/          */ 00030000
*/          */ 00040000
*/          */ 00050000
*/          */ 00060000
*/          */ 00070000
*/          */ 00080000
*/          */ 00090000
*/          */ 00100000
*/          */ 00110000
*/          */ 00120000
*/          */ 00130000
*/          */ 00140000
CEEUOPT CSECT 00150000
CEEUOPT AMODE ANY 00160000
CEEUOPT RMODE ANY 00170000
CEEUOPT CEEUOPT ANYHEAP=(24K,8K,ANYWHERE,FREE), X00180000
              BELOWHEAP=(4K,4K,FREE), X00190000
              HEAP=(4K,4K,ANYWHERE,KEEP,4K,4K), X00200000
              LIBSTACK=(4K,4K,FREE), X00210000
              RPTOPTS=(OFF), X00220000
              RPTSTG=(OFF), X00230000
              STACK=(16K,4K,ANYWHERE,KEEP), X00240000
              STORAGE=(00,NONE,00,0K), X00241000
              TERMTHDACT=(TRACE), X00250001
              TRACE=(OFF,8K,NODUMP,LE=0), X00260001
              TRAP=(OFF) 00270000
              DC C'5688-198 (C) COPYRIGHT IBM CORP. 1991, 1995. ' 00280000
              DC C'LICENSED MATERIALS - PROPERTY OF IBM' 00290000
              END 00300000

```

Figure C-1 CEEUOPT — The Executor component LE user options/parameters

FIVEPARM

Only one M5PARM macro must be coded in the FIVEPARM module. Default values are provided for most M5PARM parameters. [Figure C-1](#) contains the keywords for specifying the M5PARM parameters; the table also contains an explanation for each of the parameters as well as their default values.

The M5TERMDV in the FIVEPARM module has the following format:

```
label M5TERMDV keyword parameters
```

An M5TERMDV macro is required to define each device type to be used with VISION:Transact. The label field (positions 1-8 of the macro) specifies a DEVICE ID name. This is the name used by VISION:Transact to identify a single device or for IMS/DC a group of devices. For IMS/DC, a set of contiguous M5TERMDV macros with the same DEVICE ID name defines a device group. VISION:Transact generates a separate DIF and/or DOF for each device within a device group. The DEVICE ID named in the M5TERMDV macro may be entered in the DEVICE ID entry on the SD statement.

DEVICE ID is limited to eight letters or digits and the first character must be a letter; no embedded blanks are allowed in the name. The last M5TERMDV macro must have the label name of END to indicate the end of the device table. The M5TERMDV macros must follow the M5PARM macro.

Note that at least two M5TERMDV macros must be coded in the FIVEPARM module. The first M5TERMDV macro must define the default DEVICE ID for all VISION:Transact screen definitions and must be followed by an END M5TERMDV macro.

[Figure C-2 on page C-4](#) contains the keywords for specifying the M5PARM keyword parameters. [Figure C-3 on page C-12](#) contains the keywords for specifying the M5TERMDV parameters. All keyword parameter values must be enclosed in single quotation marks (apostrophes). Note that the DEVICE ID name coded in columns 1-8 of the M5TERMDV statement should not be enclosed in single quotation marks.

CICS Instructions

CICS customers must supply a separate M5TERMDV macro for each different device size to be used with VISION:Transact. The only keywords that should be specified for CICS on the M5TERMDV macro are the DEVICE ID (in columns 1-8), SCRNNHT, and SCRNNWD.

For example, to support devices with sizes 24 x 80 and 43 x 80, the following M5TERMDV macro statements can be used for CICS:

```
-----1-----2-----3-----4-----5-----6-----7--
MOD2      M5TERMDV SCRNNHT='24',SCRNNWD='80'
MOD4      M5TERMDV SCRNNHT='43',SCRNNWD='80'
END       M5TERMDV
```

MOD2 defines a 24 x 80 device and MOD4 defines a 43 x 80 device. The names MOD2 and MOD4 were chosen for this example, however, any names may be used. Note that a BMS map generated for a 24 x 80 device can be used on a 43 x 80 device or any device with larger dimensions. This results in a screen layout with 24 lines and 80 columns being used on the larger dimension device; a BMS map cannot be used on a device that has smaller dimensions. When a screen layout requires larger dimensions than the default DEVICE ID, then a separate M5TERMDV macro is required to define the larger device. The DEVICE ID is then used on the screen's SD statement.

IMS Instructions

IMS requires a separate DIF and/or DOF definition for each device type to be used with a screen layout. In order to generate multiple DIF/DOF definitions for a single screen layout, the same DEVICE ID name must be used on each M5TERMDV macro; one M5TERMDV macro for each separate device type. A contiguous set of M5TERMDV macros with the same DEVICE ID is called a device group.

For example, to define a device group with three different device types each of which can be used to display screen layouts up to 24 x 80, the following M5TERMDV macro statements can be used for IMS:

```
-----1-----2-----3-----4-----5-----6-----7--
T24X80  M5TERMDV  SCRNSHT='24',SCRNSWD='80',TERMDEV='(3270,2) '
T24X80  M5TERMDV  SCRNSHT='24',SCRNSWD='80',TERMDEV='3270-A2 '
T24X80  M5TERMDV  SCRNSHT='24',SCRNSWD='80',TERMDEV='3270-A4 '
END      M5TERMDV
```

When a screen uses the DEVICE ID T24X80, three DIF/DOF definitions are generated for the screen definition each with a TYPE= parameter from the TERMDEV keyword. If a device group contains devices of different sizes, then the smallest size device must be the first one defined in the group. VISION:Transact uses the SCRNSHT and SCRNSWD sizes of the first device defined in the group to verify that the screen layout fits within the device boundaries. VISION:Transact only verifies the size of the screen layout for the first device in the group and does not verify that other devices in the group have consistent sizes.

The M5TERMDV keywords that affect the generation of the MFS MOD and MID must be identical on all M5TERMDV macro statements for a device group. These keywords are MODPAGE, MODIGN, MIDIGN, and MIDNMOD.

If all of the devices in a group are output-only devices (3270 printers, SCS1, and/or SCS2 devices), then no MFS MID is generated and the MOD does not contain a NXT= parameter. In this case, operator logical paging should be suppressed with the MODPAGE keyword.

Parameter Description

Keyword	Explanation
ASTORZ	This is the amount of contiguous storage that the generator acquires as a work area. This area must be large enough to hold the largest file definition, screen definition, or request. The amount is specified in 1024-byte increments. The default is ASTORSZ='256' (256K). If message MK5S501 is issued, increase this value.

Figure C-2 M5PARM Keyword Parameters (Page 1 of 9)

Keyword	Explanation
CICSKEY	Specifies how the CLEAR and PA keys are to be handled by the VISION:Transact system under CICS. This keyword is ignored for IMS. CICSKEY='N' indicates that the VISION:Transact Executor should intercept the CLEAR and PA keys when depressed on a screen and treat these keys as a request for application termination. Application termination results in VISION:Transact erasing the screen, terminating the pseudo conversation, and deleting all resources used such as saved working storage and pseudo file data. If you select this option, the VISION:Transact application logic (the interaction cycle) is not executed when the CLEAR or PA keys are depressed as it is for the Enter and PF keys. CICSKEY='Y' indicates that the VISION:Transact Executor should not intercept the CLEAR and PA keys when depressed on a screen. This option causes the VISION:Transact application logic (the interaction cycle) to be executed when the CLEAR or PA keys are depressed just as is done for the Enter and PF keys. CICSKEY='C' indicates that the VISION:Transact Executor should not intercept the CLEAR key when depressed on a screen. This option causes the VISION:Transact application logic to be executed when the CLEAR key is depressed. With this option, PA keys are intercepted by the VISION:Transact Executor and treated as a request for application termination. CICSKEY='P' indicates that the VISION:Transact Executor should not intercept PA keys when depressed on a screen. This option causes the VISION:Transact application logic to be executed when a PA key is depressed. With this option, the CLEAR key is intercepted by the VISION:Transact Executor and treated as a request for application termination. When you select a 'Y', 'C', or 'P' for this keyword, VISION:Transact application programs must have appropriate logic to handle the CLEAR or PA keys. Application requirements and considerations for processing CLEAR and PA keys are discussed in the <i>Advantage VISION:Transact for CICS Environment Guide</i>. CICSKEY='N' is the default. Note that the value selected for this keyword can be overridden on the RC statement during application generation.

Figure C-2 M5PARM Keyword Parameters (Page 2 of 9)

Keyword	Explanation
CONREQ	Indicates whether or not the connector entry is required by the VISION:Transact system when using logical operators (EQ, NE, LT, LE, GT, and GE) on screen control (SC) statements or request processing (PR) statements. The connector entry is used to join two or more statements within a logical expression. (The first statement of a logical expression does not contain a connector entry.) CONREQ='N' indicates that the connector entry is not required by the VISION:Transact system when joining two or more statements within a logical expression. With this option, if the connector entry is left blank on the second or subsequent statements of a logical expression, then a default "OR" connector is assumed by the VISION:Transact system. CONREQ='Y' indicates that the connector entry is required by the VISION:Transact system when joining two or more statements within a logical expression. If the connector entry is left blank on the second or subsequent statements of a logical expression, then a type 3 error is generated by the VISION:Transact system. CONREQ='N' is the default.
CUSTNAM	Specifies the character string that appears in the upper left hand corner of the first page of the source listing. The default is your company name.
DATFMT	Specifies the format used in displaying the date flag. The following entries are permissible: DATFMT='0' if desired format is MMM DD,YYYY DATFMT='1' if desired format is DD MMM YYYY DATFMT='2' if desired format is YYYY MMM DD DATFMT='0' is the default.
DECCHAR	Specifies the character assigned as a decimal point for numeric data. DECCHAR='.' is the default.
DEFTERM	Default terminal device. Specifies the device that is to be used for a screen when Device Type is not specified on the SD statement. The name used here must appear in an M5TERMDV macro. If no value is supplied for this parameter, then the device named in the first M5TERMDV macro is the default terminal device. At least one M5TERMDV macro must be coded for a terminal followed by the END M5TERMDV macro.

Figure C-2 M5PARM Keyword Parameters (Page 3 of 9)

Keyword	Explanation
DELIM	Specifies the character used as the VISION:Transact system delimiter. This character can be a hexadecimal value but must not appear in a VISION:Transact input statement except as the delimiter. DELIM='#' is the default.
ERRDEST	Specifies the LTERM name to which diagnostic messages (an error screen) are directed when an unrecoverable error occurs during application execution. CICS/VS: Error messages are sent to the originating terminal even when the VISION:Transact application program ABENDs. If a value is specified for this parameter, then this is the name of an additional terminal that is to receive error messages. BMS routing is used to send messages to this terminal. The terminal named here must be compatible with the terminal characteristics described in the BMS error message map. IMS/DC: Error messages are sent to the originating terminal except when an ABEND occurs in the VISION:Transact application. If a value is supplied for this parameter, then messages are sent to the terminal named here if the PSB for the application contains a PCB for this terminal. If no value is supplied for this parameter, then messages are sent to the destination defined by the second PCB in the PSB (for instance, the first PCB after the IOPCB). If this is a modifiable PCB, the output is sent to the terminal named in the IOPCB. The PCB for the error terminal must have EXPRESS=YES so that the errors are received if the VISION:Transact application ABENDs.
FLDDSGN	Specifies the character used to indicate screen fields in SL statements. This character may be any value except the system delimiter or the highlight designator character. Choose a character that is printable and not needed in screen literals on SL statements. This entry can be overridden at screen definition time through the field designator entry on the SD statement. FLDDSGN='_' is the default.
GRPCHAR	Specifies the grouping character inserted between numeric field digits. This character is inserted between each three digits from the decimal point left. GRPCHAR=',' is the default.
HILDSGN	Specifies the character used to indicate that portions of the screen display are highlighted. This character may be any value except the system delimiter or the field designator character, but it must not be a character used for screen literals within the SL statements. HILDSGN='@' is the default.

Figure C-2 M5PARM Keyword Parameters (Page 4 of 9)

Keyword	Explanation
INVALID	VISION:Transact uses special characters in the output screen to indicate missing data base segments, invalid source field content, and overflow conditions. This character is displayed when the contents of a source field are invalid. INVALID='*' is the default.
LEFTMRK	On the VISION:Transact source listing source entries are separated by three delimiters. When at least two print positions are available between source fields, the characters specified in LEFTMRK and RIGHTMRK are used to enclose the field. When only one space is available on the print line, the character specified by SNGLSEP is used. The LEFTMRK entry specifies the left hand delimiter when sufficient space is available on the print line. LEFTMRK='(' is the default.
LIBOPNA	Specifies whether the primary VISION:Transact library should be opened for update access or read only access during an application generation run (RUN TYPE entry of G on the RC statement). When COMLIB usage tracking is specified (item tracking value of 2 or 3), all libraries are opened for update during application generation regardless of the value supplied for this keyword. LIBOPNA='U' indicates that the primary VISION:Transact library should be opened for update during an application generation (APPGEN) run. This option allows screen definitions and individual requests to be cataloged, replaced, or deleted on the primary VISION:Transact library during an APPGEN run. LIBOPNA='R' indicates that the primary VISION:Transact library should be opened for read only access during an application generation (APPGEN) run. With this option, screen definitions cannot be input and individual requests cannot be cataloged, replaced, or deleted in an APPGEN run unless the RC statement specifies a 'U' entry in LIBRARY OPEN (position 28). LIBOPNA='U' is the default. The value specified for this keyword can be overridden by the LIBRARY OPEN entry on the RC statement during application generation.

Figure C-2 M5PARM Keyword Parameters (Page 5 of 9)

Keyword	Explanation
LKEDCIS	Specifies whether all called programs are to be link-edited with a VISION:Transact application targeted for CICS or only those programs named on RP LKEDCALL statements during application generation. LKEDCIS='N' indicates that only those called programs named on RP LKEDCALL statements are to be link-edited with a VISION:Transact CICS application. LKEDCIS='Y' indicates that all called programs are to be link-edited with a VISION:Transact CICS application. When LKEDCIS='Y' is specified, RP LKEDCALL statements supplied during application generation for CICS are ignored unless an "N" is specified in the LKEDCALL PROGRAMS entry on the RC statement. LKEDCIS='N' is the default. Note that the value selected for this keyword can be overridden on the RC statement during application generation.
LKEDIMS	Specifies whether all called programs are to be link-edited with a VISION:Transact application targeted for IMS/DC or only those programs named on RP LKEDCALL statements during application generation. LKEDIMS='N' indicates that only those called programs named on RP LKEDCALL statements are to be link-edited with a VISION:Transact IMS application. LKEDIMS='Y' indicates that all called programs are to be link-edited with a VISION:Transact IMS application. When LKEDIMS='Y' is specified, RP LKEDCALL statements supplied during application generation for IMS are ignored unless an "N" is specified in the LKEDCALL PROGRAMS entry on the RC statement. LKEDIMS='Y' is the default. Note that the value selected for this keyword can be overridden on the RC statement during application generation.
LSTHGHT	Specifies the number of print lines on the output listing. The default is 66 lines.
LSTWDTH	Specifies the number of print positions on an output listing line. The default is 132 print positions. <\$IPrint positions>
Figure C-2	M5PARM Keyword Parameters (Page 6 of 9)

Keyword	Explanation
MNHTTAB	Specifies a 36-byte string representing three-character abbreviations for each month of the year. These are used for printing dates on source listings and for formatting the DATE flag.
MSGPROC	Specifies the IMS/DC message processing option. Applications generated for IMS/DC with MSGPROC='M', process multiple messages per execution. MSGPROC='S' results in single message processing by an application generated for IMS/DC. This parameter is ignored in CICS. MSGPROC='M' is the default.
NOEXIST	VISION:Transact uses special characters on output screen displays that indicate missing data base segments, invalid source field contents, and overflow conditions (source fields that do not fit on the output screen). The NOEXIST character is displayed when a specified source field is missing. NOEXIST=' ' is the default.
OVERFLW	This character is displayed when a source field cannot be displayed in the defined screen field length. OVERFLW='+' is the default.
PRTCTL	Specifies carriage control for printing the source listing. Enter the ASA control characters to produce single spacing (PRTCTL=' '), double spacing (PRTCTL='0'), or triple spacing (PRTCTL='<196>'). PRTCTL=' ' is the default.
RIGHTMK	On the VISION:Transact source listing source entries are separated by three delimiters. When at least two print positions are available between source fields, the characters specified in LEFTMRK and RIGHTMK are used to enclose the field. When only one space is available on the print line, the SNGLSEP character is used. The RIGHTMK entry specifies the right hand delimiter to be used when sufficient space is available on the print line. RIGHTMK=')' is the default.
SNGLSEP	Specifies the character used to separate entries on the source listing when only one print position is available between entries. SNGLSEP=', ' is the default.

Figure C-2 M5PARM Keyword Parameters (Page 7 of 9)

Keyword	Explanation
TFREQ	Indicates whether or not TF statements are required by the VISION:Transact system for definition of temporary fields. TFREQ='N' indicates that TF statements are not required by the VISION:Transact system for the definition of temporary fields. If a temporary field is encountered that has not been defined on a TF statement, the VISION:Transact system creates an implicit definition of the temporary field. In this case, a type 0 message is output by the VISION:Transact system informing the programmer of the implicit attributes created for the field (length, type, scale, and edit length). TFREQ='Y' indicates that TF statements are required by the VISION:Transact system for the definition of temporary fields. If a temporary field is encountered that has not been defined on a TF statement, a type 3 message is output by the VISION:Transact system indicating the field is not defined; no implicit definition is created for the field. In this case, a TF statement must be provided for the temporary field to permit generation of the application. TFREQ='N' is the default.
TODAY	Specifies the format used to print or display the TODAY flag. Permissible entries are: TODAY='MMDDYY' TODAY='DDMMYY' TODAY='YYMMDD' TODAY='MMYYDD' TODAY='DDYYMM' TODAY='YYDDMM' TODAY='MMDDYY' is the default.
TPMON	Identifies the teleprocessing environment for which applications are to be generated. TPMON='C' indicates CICS/VS generation and TPMON='I' indicates IMS/DC generation. The default is either 'C' or 'I' depending on whether your VISION:Transact generator supports the generation of CICS/VS or IMS/DC applications. If your VISION:Transact generator supports generation of both CICS/VS and IMS/DC applications, the default is 'C' for CICS/VS.

Figure C-2 M5PARM Keyword Parameters (Page 8 of 9)

Keyword	Explanation
XLOAD	Indicates whether or not the IMS Executor is to be linked with the generated application or dynamically loaded. XLOAD='Y' indicates dynamic loading is to be performed. XLOAD='N' indicates that the Executor is link-edited with the generated application. XLOAD='Y' is the default.

Figure C-2 M5PARM Keyword Parameters (Page 9 of 9)

Note: Specifications from M5PARM are included in the generated application. Modifications to M5PARM have no effect on a generated application unless that application is subsequently regenerated.

The following table contains the keywords for specifying the M5TERMDV parameters.

Keyword	Explanation
SCRNHT	Specifies the number of lines this device is capable of displaying. This entry is used to edit SL statements and in generating both MFS and BMS. SCRNHT='24' is the default.
	Specifies the number of character positions this device is capable of displaying on one line. This entry is used to edit SL statements and in generating both MFS and BMS. SCRNWD='80' is the default.
	Note: The following keywords are used for IMS/DC; they are ignored for CICS.

Figure C-3 M5TERMDV Keyword Parameters (Page 1 of 4)

Keyword	Explanation
TERMDEV	This 16-character literal is used to construct the TYPE= parameter on the MFS DEV statement for this device. Enter the device type exactly as it should appear in the DEV statement. For example, TERMDEV='(3270,2)' results in an MFS DEV statement with TYPE=(3270,2). This keyword is required for IMS/DC; there is no default. VISION:Transact recognizes the following device types when enclosed in single quotation marks: ■ For 3270 displays: (3270,1) (3270,2) 3270-An (n = 1 through 15) ■ For 3270 printers: 3270P (3270P,1) (3270P,2) ■ For SCS1 or SCS2 output: - SCS1 - SCS2 3270 printers, SCS1, and SCS2 devices are treated as output-only devices by VISION:Transact.
NUMPFK	Defines the number of PF keys for a 3270 display device. NUMPFK is ignored and a value of zero is assumed if TERMDEV is not a 3270 display. This keyword is used to determine the number of PF key values to use for the PFK= parameter on the MFS DEV statement for this 3270 display device. NUMPFK='24' is the default.
XCOLOR	Specifies whether or not the device supports the extended color attribute (7 colors). XCOLOR='Y' indicates the device supports extended color; XCOLOR='N' indicates the device does not support extended color. XCOLOR='N' is the default.
XHIL	Specifies whether or not the device supports the extended highlighting attribute. XHIL='Y' indicates the device supports extended highlighting; XHIL='N' indicates the device does not support extended highlighting. XHIL='N' is the default.

Figure C-3 M5TERMDV Keyword Parameters (Page 2 of 4)

Keyword	Explanation
XPSYM	Specifies whether or not the device supports the programmed symbol or EGCS attribute. XPSYM='Y' indicates the device supports programmed symbols or EGCS; XPSYM='N' indicates the device does not support this attribute. XPSYM='N' is the default.
DVPAGE	If this keyword is not supplied, then the MFS DEV statement for this device does not contain a PAGE= parameter. If this keyword is supplied, then its value is used in the PAGE= parameter on the DEV statement. For example, if DVPAGE=(55,EJECT(ENDPP)) is specified, then the DEV statement contains PAGE=(55,EJECT(ENDPP)). Note that the PAGE= parameter on the DEV statement is only used for 3270 printers and SCS1 and SCS2 output devices.
DVWIDTH	If this keyword is not supplied, then the MFS DEV statement for this device does not contain a WIDTH= parameter. If this keyword is supplied, then its value is used in the WIDTH= parameter on the DEV statement. Note that the WIDTH= parameter on the DEV statement is only used for 3270 printers and SCS1 and SCS2 output devices.
DVFEAT	If this keyword is not supplied, then the MFS DEV statement for the device contains FEAT=IGNORE. If this keyword is supplied, then its value is used in the FEAT= parameter on the DEV statement.
DVMISC	This keyword allows you to supply one or more miscellaneous parameters for this device's DEV statement. For example, DVMISC= 'FORMS= "A.DATASET",SLDI=8' causes VISION:Transact to add FORMS= 'A.DATASET',SLDI=8 to the device's DEV statement. If the character string supplied for DVMISC contains an apostrophe, then two consecutive apostrophes must be coded. This is shown surrounding A.DATASET in the above example. Note that the DVMISC keyword should not supply the DEV statement parameters TYPE=, FEAT=, SYSMSG=, PFK=, WIDTH=, or PAGE= as these parameters may be generated by VISION:Transact.
MODPAGE	If MODPAGE='Y' is specified, then the MOD's MSG statement in the MFS for this device contains PAGE=YES which enables operator logical paging. If MODPAGE='N' is specified, then no PAGE= parameter is generated. The default is MODPAGE='Y'. Note if all the devices in a device group are output only devices (3270 printers, SCS1, or SCS2 devices), then MODPAGE='N' should be specified.

Figure C-3 M5TERMDV Keyword Parameters (Page 3 of 4)

Keyword	Explanation
MODIGN	If MODIGN='Y' is specified, then the MOD's MSG statement in the MFS for this device contains the IGNORE keyword in the SOR= parameter. If MODIGN='N' is specified, then the SOR= parameter does not contain the IGNORE keyword. The default is MODIGN='Y'. If device features are supplied with the M5TERMDV DVFEAT keyword, the IGNORE keyword may be inappropriate for the SOR= parameter on the MFS MSG statement.
MIDIGN	If MIDIGN='Y' is specified, then the MID's MSG statement in the MFS for this device contains the IGNORE keyword in the SOR= parameter. If MIDIGN='N' is specified, then the SOR= parameter does not contain the IGNORE keyword. The default is MIDIGN='Y'. If device features are supplied with the M5TERMDV DVFEAT keyword, then the IGNORE keyword may be inappropriate for the SOR= parameter on the MFS MSG statement.
MIDNMOD	If MIDNMOD='Y' is specified, then the MID's MSG statement in the MFS for this device contains the NXT=modname parameter where modname is the name of the MOD for this screen. If MIDNMOD='N' is specified, then the NXT=modname parameter is not generated. MIDNMOD='N' is the default.

Figure C-3 M5TERMDV Keyword Parameters (Page 4 of 4)

```

MP      TITLE 'FIVEPARM  COMPUTER ASSOCIATES, INC.'      00010001
      COPY  M5PARM      00020000
*      00030000
*      M5PARM USED TO GENERATE DEFAULT M5PARM FOR      00040000
*      INSTALLATION.      00050000
*      00060000
*      THE DEVICE DEFINITIONS ARE INTENDED AS EXAMPLES      00070000
*      ONLY.  EACH INSTALLATION SHOULD REVIEW ITS SPECIFIC      00080000
*      NEEDS AND DEFINE ITS OWN DEVICES AND DEVICE GROUPS.      00090000
*      00100000
*      ENTRY SEUSERID      FOR FIVEGEN -REP      00110000
*      00120000
*      M5PARM      00130000
*      00140000
*      00150000
*      THE FOLLOWING COMMENT LINES ARE AN EXAMPLE OF A      00160000
*      CUSTOMIZED M5PARM SPECIFICATION.      00170000
*      00180000
*      M5PARM ERRDEST='MOTERM',DEFTERM='T4380',      X 00190002
*      DELIM=';',TODAY='YYMMDD',      X 00200002
*      CUSTNAM='ABC MANUFACTURING, INC.'      00210000
*      00220000
*      00230000
* D3270 DEFINES A GENERIC 3270 DISPLAY WITH DIMENSIONS 24X80.      00240000
*      00250000
* THIS DEVICE CAN BE USED BY BOTH IMS AND CICS APPLICATIONS      00260000
* AND IS THE DEFAULT DEVICE FOR ALL INSTALLATIONS THAT USE THE      00270000
* INSTALLED M5PARM WITHOUT MODIFICATION.      00280000
*      00290000
D3270  M5TERMDV SCRNT='24',SCRWD='80',TERMDEV='(3270,2)'      00300000
*      00310000
* DP24X80 IS A GROUP THAT CONTAINS A DISPLAY DEVICE AND      00320000
* A PRINTER WITH DIMENSIONS 24X80.      00330000
*      00340000

```

Figure C-4 FIVEPARM Source Code (Page 1 of 2)

```

DP24X80  M5TERMDV  SCRNT='24',SCRNWD='80',TERMDEV='(3270,2)'      00350000
DP24X80  M5TERMDV  SCRNT='55',SCRNWD='120',TERMDEV='(3270P,2)',    X00360000
          DVPAGE='(55,SPACE) '                                     00370000
*                                                                 00380000
* D24X80 IS A GROUP THAT CONTAINS DISPLAY DEVICES WITH          00390000
* MINIMUM DIMENSIONS 24X80.  NOTE THAT THE FIRST ENTRY IN      00400000
* THE GROUP DETERMINES THE DIMENSIONS FOR THE GROUP WHILE      00410000
* SUBSEQUENT ENTRIES MAY BE PHYSICALLY LARGER.                 00420000
*                                                                 00430000
D24X80    M5TERMDV  SCRNT='24',SCRNWD='80',TERMDEV='(3270,2)'      00440000
D24X80    M5TERMDV  SCRNT='24',SCRNWD='80',TERMDEV='3270-A2',      X00450000
          XCOLOR='Y',XHIL='Y',XPSYM='Y'                          00460000
D24X80    M5TERMDV  SCRNT='27',SCRNWD='132',TERMDEV='3270-A7'     00470000
D24X80    M5TERMDV  SCRNT='43',SCRNWD='80',TERMDEV='3270-A4'      00480000
*                                                                 00490000
* P24X80 IS A GROUP THAT DEFINES A GROUP OF PRINTERS           00500000
* WITH DIMENSION 24X80                                         00510000
*                                                                 00520000
P24X80    M5TERMDV  SCRNT='24',SCRNWD='80',TERMDEV='(3270P,2)',    X00530000
          MODPAGE='N',                                           X00540000
          DVPAGE='(24,SPACE) '                                    00550000
P24X80    M5TERMDV  SCRNT='55',SCRNWD='132',TERMDEV='SCS1',        X00560000
          MODPAGE='N',                                           X00570000
          DVPAGE='(24,EJECT(ENDPP)) '                          00580000
P24X80    M5TERMDV  SCRNT='55',SCRNWD='80',TERMDEV='SCS2',        X00590000
          MODPAGE='N',                                           X00600000
          DVPAGE='(24,SPACE) '                                    00610000
*                                                                 00620000
* P55X120 IS A GROUP THAT DEFINES A GROUP OF PRINTERS          00630000
* WITH DIMENSION 55X120                                         00640000
*                                                                 00650000
P55X120   M5TERMDV  SCRNT='55',SCRNWD='120',TERMDEV='(3270P,2)',    X00660000
          MODPAGE='N',                                           X00670000
          DVPAGE='(55,SPACE) '                                    00680000
P55X120   M5TERMDV  SCRNT='55',SCRNWD='132',TERMDEV='SCS1',        X00690000
          MODPAGE='N',                                           X00700000
          XCOLOR='Y',XHIL='Y',XPSYM='Y',                        X00710000
          DVPAGE='(55,EJECT(ENDPP)) ',                          X00720000
          DVMISC='SLDI=8'                                         00730000
END        M5TERMDV                                             00740000
*                                                                 00750000
          END                                                    00760000

```

Figure C-4 FIVEPARM Source Code (Page 2 of 2)

MARKLIBP

COMLIB is a sub component of the Generator component and provides standard default conditions for the library parameters usually determined by the operating environment in each installation. These parameters affect directory blocking, optional library tracking information, object compression, and reserve. Because these parameters are functions of the installation, COMLIB provides you with the capability of changing the default parameters.

A special program module called MARKLIBP is used for this purpose and is supplied with COMLIB. It is made available to you and can be changed to suit your needs. Each condition supplied as the default standard is shown. The module itself is supplied as an assembly language source CSECT and is well documented in its source form. A description of the parameters and a listing of the CSECT follows; the default is clearly indicated for each parameter.

Parameter	Parameter Name	COMLIB Default	
Library directory blocking factor	DIRBLK	0 - allows COMLIB to calculate the optimum directory blocking factor based on the device and size of the library.	
Not applicable to VSAM		A larger value improves decode and library maintenance performance. The table shown indicates maximum values depending on device.	
		<u>Device</u>	<u>Maximum Blocking factor</u>
		3330	372
		3340	243
		3350	561
		3380	733
		3390	800
COMLIB reserve flag for shared DASD	RESERVE	0	
Item tracking flag	ITEMTRAK	0	
Minimum Compress Size	MINCMPSZ	507	
Compression Flag	COMPRESS	0	

```

PBLP4    TITLE 'MARKLIBP - COMPUTER ASSOCIATES, INC.'          00010001
        ISEQ 73,80                                              00020000
*****
*
*          PROPRIETARY AND CONFIDENTIAL INFORMATION OF          * 00040000
*          COMPUTER ASSOCIATES, INC.                            * 00050000
*          USE RESTRICTED BY WRITTEN LICENSE AGREEMENT          * 00060001
*
*          DO NOT REMOVE THIS NOTICE                            * 00070000
*
*          COPYRIGHT (C) COMPUTER ASSOCIATES, INC.              * 00080000
*          AS AN UNPUBLISHED WORK. ALL RIGHTS RESERVED.          * 00090000
*
*          NOTE: U.S. GOVERNMENT DFARS CONTRACTS, RESTRICTED    * 00100000
*          USE, DUPLICATION, OR DISCLOSURE IS SUBJECT TO          * 00110001
*          STATED IN PARAGRAPH (C) (1) (II) OF THE RIGHTS IN      * 00120000
*          AND COMPUTER SOFTWARE CLAUSE AT DFARS 252.227-7013.    * 00130000
*
*****
*          MACRO
*          DIRBLK &FACTOR
DIRBLKSZ EQU  &FACTOR*32
*          MEND
*
MARKLIBP CSECT
MARKLIBP AMODE ANY
MARKLIBP RMODE 24
*
*DIRECTORY SPECIFIES THE NUMBER OF ENTRIES IN A LIBRARY DIRECTORY 00200000
*BLOCKING  BLOCK. LARGER VALUES DECREASE ACCESS TIME FOR LARGE 00210000
*FACTOR    LIBRARIES. THE FACTOR CANNOT EXCEED THE VALUES GIVEN 00220000
*          BELOW FOR THE DEVICE USED. A VALUE OF 0 ALLOWS COMLIB 00230000
*          TO CALCULATE THE OPTIMUM DIRECTORY BLOCKING FACTOR      00240000
*          BASED ON THE DEVICE AND SIZE OF THE LIBRARY.           00250000
*          NOT APPLICABLE TO VSAM LIBRARIES.                      00260000
*
*          3330 372
*          3340 243
*          3350 561
*          3375 549
*          3380 733
*          3390 800 (ALSO APPLIES TO HITACHI DEVICE H6587)
*          9345 692
*
*          DIRBLK 0
*
*DEVICE    THE DEVICE RESERVE FLAG PROVIDES THE ABILITY TO ISSUE 00261001
*RESERVE   A DEVICE RESERVE RATHER THAN AN OPERATING SYSTEM ENQUE. 00262001
*FLAG      ENTER A '1' TO ACTIVATE THE RESERVE FEATURE FOR A
*          MAXIMUM OF ONE LIBRARY IN A GIVEN RUN. ENTER A 2 TO
*          ACTIVATE THE MULTILIB RESERVE FEATURE WHICH SUPPORTS
*          DEVICE RESERVATION FOR MORE THAN ONE LIBRARY IN A
*          GIVEN RUN. (NOT APPLICABLE TO DOS/VS(E))
*
RESERVE EQU 0
*
*ITEM      THE ITEM TRACKING FLAG ENABLES OPTIONAL FEATURES OF THE 00270000
*TRACKING  ITEM TRACKING FACILITY. ENTER A '1' TO REQUIRE THE USE 00280000
*FLAG      OF IT STATEMENTS WITH UPDATER IDENTIFICATION FOR ALL
*          LIBRARY UPDATES. ENTER A '2' TO ENABLE THE DATE-OF-ITEM 00290000
*          USE FACILITY. ENTER A '3' TO ENABLE BOTH.              00300000
*
ITEMTRAK EQU 0
*
*          C011 00310000
*          C011 00320000
*****C011 00330000
*****C011 00340000
* THE FOLLOWING ENTRIES HAVE TO DO WITH THE OPTIONAL COMPRESSION *C011 00350000
* OF OBJECTS STORED IN THE LIBRARY. COMPRESSED OBJECTS MUST BE *C011 00360000
* DE-COMPRESSED WHEN THEY ARE RETRIEVED. THIS WILL ADD SOME *C011 00370000
* MINIMAL OVERHEAD IN EXCHANGE FOR REDUCED I/O COUNTS AND DISK *C011 00380000
* SPACE USAGE. COMPRESSION WILL BE MOST ADVANTAGEOUS IF YOUR *C011 00390000
* LIBRARY OBJECTS TEND TO BE LARGE (MORE THAN 100 FIELDS IN A *C011 00400000

```

Figure C-5 MARKLIBP Source Code (Page 1 of 2)

```

* FILE DEFINITION OR 200 ENTRIES IN A TABLE). IF COMPRESSED *C011 00730000
* OBJECTS HAVE BEEN STORED IN THE LIBRARY, THEY WILL STILL BE *C011 00740000
* RETRIEVED AND DE-COMPRESSED EVEN THOUGH LIBRARY COMPRESSION *C011 00750000
* WAS SUBSEQUENTLY DISABLED. A LIBRARY WITH COMPRESSED OBJECTS *C011 00760000
* MAY BE CONVERTED TO A LIBRARY WITHOUT ANY COMPRESSED OBJECTS *C011 00770000
* BY DUMPING THE LIBRARY AND THEN RESTORING IT WITH COMPRESSION *C011 00780000
* DISABLED. *C011 00790000
* *C011 00800000
* THE COMPRESSION SOFTWARE IS PROVIDED COURTESY OF *C011 00810000
* JEAN-LOUP GAILLY AND MARK ADLER. *C011 00820000
***** *C011 00830000
* *C011 00840000
*MINIMUM THE MINIMUM COMPRESS SIZE SPECIFIES THE MINIMUM *C011 00850000
*COMPRESS SIZE THAT A LIBRARY OBJECT MUST BE BEFORE IT WILL *C011 00860000
*SIZE BE COMPRESSED WHENEVER COMPRESSION IS ENABLED. *C011 00870000
* OBJECTS LARGER THAN THE SPECIFIED SIZE WILL BE *C011 00880000
* COMPRESSED WHENEVER COMPRESSION IS ENABLED. *C011 00890000
* OBJECTS WHOSE SIZE IS LESS THAN OR EQUAL TO THE *C011 00900000
* SPECIFIED SIZE WILL NOT BE COMPRESSED REGARDLESS *C011 00910000
* OF WHETHER COMPRESSION IS ENABLED OR NOT. *C011 00920000
* THE SPECIFIED SIZE SHOULD NEVER BE SMALLER THAN *C011 00930000
* 507 AND MAY BE SPECIFIED AS LARGE AS 65535. *C011 00940000
* *C011 00950000
MINCMPSZ EQU 507 *C011 00960000
* *C011 00970000
*COMPRESSION THE COMPRESSION FLAG ENABLES THE OPTIONAL LIBRARY *C011 00980000
*FLAG OBJECT COMPRESSION. ENTER A '1' TO ENABLE THE *C011 00990000
* COMPRESSION OF LIBRARY OBJECTS WHOSE SIZE EXCEEDS *C011 01000000
* THE 'MINIMUM COMPRESS SIZE' SPECIFIED ABOVE. *C011 01010000
* ENTER A '0' TO DISABLE LIBRARY OBJECT COMPRESSION. *C011 01020000
* *C011 01030000
COMPRESS EQU 0 *C011 01040000
* *C011 01050000
* *C011 01060000
***** *C011 01070000
* USERS MUST NOT MAKE CHANGES FOLLOWING THIS PAGE *C011 01080000
***** *C011 01090000
EJECT *C011 01100000
ORG *C011 01110000
LPRELNO DC CL4'C4.5' *C011 01120000
LPDIRBLK DC Y(DIRBLKSZ) *C011 01130000
LPRESERV DC AL1(RESERVE) *C011 01140000
LPITKFLG DC AL1(ITEMTRAK) *C011 01150000
LPCMPsiz DC AL2(MINCMPSZ) *C011 01160000
LPCMPFLG DC AL1(COMPRESS) *C011 01170000
* *C011 01180000
END *C011 01190000

```

Figure C-5 MARKLIBP Source Code (Page 2 of 2)

Figure C-6 M5LEPARM Source Code

Sample ISPF Startup CLIST

This appendix contains a sample CLIST that illustrates how you can make your ISPF Workbench libraries available to the ISPF environment. You can then invoke this CLIST from the TSO prompt to start ISPF with the ISPF Workbench options included.

You can find the following CLIST in the TRANSACT.WB061.CLIST dataset as member ISPF.

```
PROC 0
CONTROL LIST MSG

FREE F(SYSPROC ISPLLIB ISPMLIB ISPPLIB ISPSLIB ISPTLIB +
      ISPTABL ISPPROF M9LIST)

ALLOC F(SYSPROC) DA( 'SYS1.CLIST'          +
                    'ISR.ISPF.ISRCLIB'      +
                    'TRANSACT.WB061.WBTCLIST') SHR

ALLOC F(ISPLLIB) DA( 'TRANSACT.WB061.WBTLOAD' +
                    'TRANSACT.TR078.TRTGEN'  +
                    'BUILDER.R140.SMPE.T.BLSYSL' +
                    'INFORM.R40.TARGET.INTLOAD' ) SHR

ALLOC F(ISPMLIB) DA( 'TRANSACT.WB061.WBTMSGs' +
                    'ISR.ISPF.ISRMLIB'      +
                    'ISP.ISPF.ISPMLIB') SHR

ALLOC F(ISPPLIB) DA( 'TRANSACT.WB061.WBTPANEL' +
                    'ISR.ISPF.ISRPLIB'      +
                    'ISP.ISPF.ISPPLIB') SHR

ALLOC F(ISPSLIB) DA( 'TRANSACT.WB061.WBTSKELS' +
                    'ISR.ISPF.ISRSLIB'      +
                    'ISP.ISPF.ISPSLIB') SHR

ALLOC F(ISPTLIB) DA( 'ISR.ISPF.ISRTLIB'      +
                    'ISP.ISPF.ISPTLIB') SHR

ALLOC F(ISPTABL) DA('ISR.ISPF.ISRTLIB')

ALLOC F(ISPPROF) DA('&SYSUID..ISPF.PROFILE')

/* NOTE: THE FOLLOWING ALLOCATE STATEMENT SHOWS HOW YOU CAN */
/* PRE-ALLOCATE YOUR WORKBENCH UTILITY LIST DATA SET.    */
/* THIS IS OPTIONAL. IF THIS DATA SET IS NOT PRE-ALLOCATED, */
/* WORKBENCH WILL DYNAMICALLY ALLOCATE IT WHEN NEEDED.      */
```

Figure D-1 Sample ISPF Startup CLIST (Page 1 of 2)

```

ALLOC F(M9LIST) DA('TRANSACTION.&SYSUID..M9LIST1') OLD

/* NOTE: THE FOLLOWING ALLOCATE STATEMENT SHOWS HOW WORKBENCH */
/* CUSTOMERS CAN ALLOCATE A FILE TAILORING OUTPUT DATA SET. */

FREE FI(ISPFILE)

IF &SYSDSN('TRANSACTION.WB061.&SYSUID..FTOUTPUT') = OK THEN +
  ALLOC F(ISPFILE) DA('TRANSACTION.WB061.&SYSUID..FTOUTPUT') SHR
ELSE +
  ALLOC F(ISPFILE) DA('TRANSACTION.WB061.&SYSUID..FTOUTPUT') +
    NEW CATALOG UNIT(SYSDA) +
    SPACE(10,5) DIR(10) +
    DSORG(PO) RECFM(F B) LRECL(80) BLKSIZE(3120)

/* NOTE: THE FOLLOWING ALLOCATE STATEMENT SHOWS HOW WORKBENCH */
/* CUSTOMERS CAN ALLOCATE THE TABLE DATA SET USED BY THE IMPORT */
/* FACILITY. THE DATA SET MUST BE PREDEFINED AND CATALOGED. */
/* (THE "&SYSUID" ELEMENT IS REPLACED BY THE CURRENT USER ID.) */

ALLOC F(DEFTLIB) DA('TRANSACTION.WB061.&SYSUID..TLIB') SHR

/* NOW START ISPF USING THE ISPF PRIMARY MENU PANEL. */

ISPSTART PANEL(ISR@PRIM)

EXIT

```

Figure D-1 Sample ISPF Startup CLIST (Page 2 of 2)

ISPF Workbench Invocation Panels

You can find the following members in the TRANSACT.WB061.PANELS dataset.

XSR@PRIM

The bold lines on the following sample ISPF primary menu panel show how the ISPF Workbench can be invoked from your ISPF primary menu.

```
-->> SAMPLE <<-- ISPF/PDF PRIMARY OPTION MENU VERSION n.n.n -----
%OPTION ==>_ZCMD
%
% 0 +ISPF PARMS - Specify terminal and user parameters +USERID - &ZUSER +
% 1 +BROWSE - Display source data or output listings +PROC - &ZLOGON
% 2 +EDIT - Create or change source data +PF KEYS - &ZKEYS
% 3 +UTILITIES - Perform utility functions +TERMINAL - &ZTERM
% 4 +FOREGROUND - Invoke language processors or script +TIME - &ZTIME
% 5 +BATCH - Submit job for language processing +JULIAN - &ZJDATE
% 6 +COMMAND - Enter TSO command or CLIST +DATE - &ZDATE
% 7 +DIALOG TEST - Perform dialog testing
% 8 +LM UTILITIES- Perform library administrator utility functions
% C +CHANGES - Display summary of changes for this release
% F +FILE-AID - FILE-AID data handling utilities
% M +LDCS - Local Data Center Services Panels
% T +TUTORIAL - Display information about ISPF/PDF
% WB +Workbench -%VISION:Workbench Facility - Release 6.1
% BL +Builder -%VISION:Builder 14.0 Workbench
% TR +Transact -%VISION:Transact 7.8 Workbench
% IN +Inform -%VISION:Inform 4.0 Workbench (Definition Processor)
% X +EXIT - Terminate ISPF using log and list defaults
%
+Enter%END+command to terminate ISPF.
%
)INIT
.HELP = ISR00003
&ZPRIM = YES /* ALWAYS A PRIMARY OPTION MENU */
&ZHTOP = ISR00003 /* TUTORIAL TABLE OF CONTENTS */
&ZHINDEX = ISR91000 /* TUTORIAL INDEX - 1ST PAGE */
VPUT (ZHTOP,ZHINDEX) PROFILE
&M9PRODC = 'Workbench'
)PROC
&ZSEL = TRANS( TRUNC(&ZCMD,('.'))
0,'PANEL(ISFOPTA)'
1,'PGM(ISRBRO) PARM(ISRBRO01)'
2,'PGM(ISREDIT) PARM(P,ISREDM01)'
3,'PANEL(ISRUTIL)'
4,'PANEL(ISRFPA)'
5,'PGM(ISRJB1) PARM(ISRJPA) NOCHECK'
6,'PGM(ISRPTC)'
```

Figure E-1 Invoking ISPF Workbench from the ISPF Primary Menu (Page 1 of 2)

```

7, 'PGM(ISRYXDR) NOCHECK'
8, 'PANEL(ISRLPRIM)'
C, 'PGM(ISPTUTOR) PARM(ISR00005)'
F, 'PANEL(IFAMU01) NEWAPPL(FAXE)'
M, 'PANEL(MICA)'
T, 'PGM(ISPTUTOR) PARM(ISR00000)'
WB, 'PANEL(M9PRIM)'
BL, 'PGM(M9BOOT) PARM(BDM4) NOCHECK'
TR, 'PGM(M9BOOT) PARM(ODM5) NOCHECK'
IN, 'PGM(M9BOOT) PARM(PMM4) NOCHECK'
' ', ' '
X, 'EXIT'
*, '?' )

&ZTRAIL = .TRAIL
&GVNXTSEL = .TRAIL
)END

```

Figure E-1 Invoking ISPF Workbench from the ISPF Primary Menu (Page 2 of 2)

M9PRIM

This is the ISPF Workbench Selection Menu panel. This is the primary menu panel for entry into the ISPF Workbench.

```
)ATTR  
 )BODY EXPAND(°°)  
 %SELMENU -°-°- VISION:Workbench for ISPF Selection Menu --°-°-----  
 %OPTION ===> _ZCMD  
 %  
 %  
 %          VISION:Workbench Release 6.1+  
 %  
 1 - BL+(M4) Workbench for%VISION:Builder 14.0  
 %  
 2 - TR+(M5) Workbench for%VISION:Transact 7.8  
 %  
 3 - IN+(DA) Workbench for%VISION:Inform 4.0 (Definition Processor)  
 %  
 T+- Introduction To VISION:Workbench for ISPF  
 %  
 X+- Exit the VISION:Workbench  
 +  
 +  
 +  
 +  
 +  
                                     %Computer Associates International, Inc. (CA)  
 )INIT  
 .HELP = M9PRIMH  
 &ZFRIM = YES           /* ALWAYS A PRIMARY OPTION MENU      */  
 &M9PRODUCT = 'Workbench'  
 )PROC  
 &zSEL = TRANS( TRUNC(&ZCMD,'.')  
            1,'PGM(M9BOOT) PARM(BDM4) NOCHECK'  
            BL,'PGM(M9BOOT) PARM(BDM4) NOCHECK'  
            M4,'PGM(M9BOOT) PARM(BDM4) NOCHECK'  
            2,'PGM(M9BOOT) PARM(ODM5) NOCHECK'  
            TR,'PGM(M9BOOT) PARM(ODM5) NOCHECK'  
            M5,'PGM(M9BOOT) PARM(ODM5) NOCHECK'  
            3,'PGM(M9BOOT) PARM(PMM4) NOCHECK'
```

Figure E-2 VISION:Workbench for ISPF Selection Menu Panel (Page 1 of 2)

Figure E-2 VISION:Workbench for ISPF Selection Menu Panel (Page 2 of 2)

Skeleton and User Panel Listings

You can find the following members in the TRANSACT.WB061.SKELS dataset.

M9GCTPU2

```
)ATTR
+ TYPE (TEXT)      INTENS (LOW)  SKIP (&SKIPVAR)
% TYPE (TEXT)      INTENS (HIGH) SKIP (&SKIPVAR)
  TYPE (INPUT)     INTENS (HIGH) CAPS (ON)  JUST (LEFT)
@ TYPE (INPUT)     INTENS (LOW)  CAPS (ON)  JUST (LEFT)  PADC (&PADVAR)
} TYPE (OUTPUT)    INTENS (HIGH)  SKIP (ON)
{ TYPE (OUTPUT)    INTENS (LOW)   SKIP (ON)
)BODY EXPAND (|)
%USRPANEL --- &GVITEMLM -|-|
%COMMAND ==> _ZCMD
+
+ENTER%END+TO PROCESS USING THE OPTION PREVIOUSLY ENTERED.
+ENTER%CANCEL+TO TERMINATE PROCESSING THIS MEMBER.
+
+
%ENTER THE NAME OF THE VISION:TRANSACT GEN LIBRARY:
+VISION:TRANSACT GENLIB ==> _GCUSVL2
+
%LINK OBJECT FILE? ==> _Z + (YES OR NO)
+ LINK TO ==> _GCUSLKL2
+ INCLUDES FROM:
+ VISION:TRANSACT ==> _GCUSMKI2
+ MONITOR ==> _GCUSMNI2
)INIT
.HELP = M9GCTPH0
.ZVARS = ' (GCUSLNK) '
)PROC
VPUT (GCUSVL2 GCUSLNK GCUSLKL2 GCUSMKI2 GCUSMNI2) PROFILE
)END
```

Figure F-1 M9GCTPU2 - VISION:Transact Batch Job Submission User Panel

M9GCTSBG

```

)CM
)CM THIS IS A SAMPLE ISPF FILE TAILORING SKELETON FOR USE WITH
)CM WORKBENCH RELEASE 6.1. IT WILL GENERATE MVS JCL FOR A TRANSACT
)CM BATCH RUN (APPGEN, DEF RUN, OR SSR). THIS FILE TAILORING
)CM SKELETON IS INTENDED TO BE USED WITH THE SAMPLE USER PANEL
)CM 'M9GCTPU2'. THIS PANEL HAS BEEN PROVIDED IN YOUR WORKBENCH
)CM PANEL LIBRARY.
)CM
)CM THERE ARE 4 TYPES OF VARIABLES (WORDS PRECEDED BY AMPERSANDS) USED
)CM IN THIS SKELETON INCLUDING:
)CM   - VARIABLES FROM THE USER PANEL
)CM     YOU CAN CHANGE THESE
)CM   - VARIABLES SET BY WORKBENCH
)CM     YOU CANNOT CHANGE THESE
)CM   - ISPF SYSTEM VARIABLES
)CM   - LOCAL VARIABLES THAT ARE SET AND USED DURING FILE TAILORING
)CM
)CM THE VARIABLES FROM THE USER PANEL INCLUDE:
)CM VARIABLE          USAGE
)CM
)CM   &GCUSVL2          VISION:TRANSACT GEN LIBRARY
)CM   &GCUSLNK          IF THE OBJECT FILE SHOULD BE LINKED
)CM   &GCUSLKL2         LINK LIBRARY
)CM   &GCUSMKI2         VISION:TRANSACT INCLUDE LIBRARY
)CM   &GCUSMNI2         MONITOR INCLUDE LIBRARY
)CM
)CM WORKBENCH RESERVED VARIABLE NAMES INCLUDE:
)CM VARIABLE          USAGE
)CM
)CM   &GSDDNAM          DATA DEFINITION NAME
)CM   &GSDSN            DATA SET NAME
)CM   &GSDISP           DATA SET STATUS AND DISPOSITION
)CM   &GSVOLSER         VOLUME SERIAL NUMBER
)CM   &GSUNIT           UNIT TYPE
)CM   &GSALLOC          FILE ALLOCATION
)CM
)CM ISPF SYSTEM VARIABLES NAMES INCLUDE:
)CM VARIABLE          USAGE
)CM
)CM   &Z                A VARIABLE WHOSE VALUE IS NULL
)CM
)CM
)CM   )SEL &GCBOJCL1 ^= &Z
)CM   &GCBOJCL1
)CM   )ENDSEL
)CM   )SEL &GCBOJCL2 ^= &Z
)CM   &GCBOJCL2
)CM   )ENDSEL
)CM   )SEL &GCBOJCL3 ^= &Z
)CM   &GCBOJCL3
)CM   )ENDSEL
)CM   )SEL &GCBOJCL4 ^= &Z
)CM   &GCBOJCL4
)CM   )ENDSEL
)CM
)CM INCLUDE EXECUTE AND STEPLIB STATEMENTS
)CM
)CM //MISPF EXEC PGM=MARKV,REGION=2M
)CM //STEPLIB DD DSN=&GCUSVL2,DISP=SHR
)CM
)CM LOOP THROUGH THE ISPF TABLE OF DATA SET CHARACTERISTICS ENTERED
)CM ON THE 'BATCHGEN' PANEL GENERATING APPROPRIATE DD STATEMENTS
)CM FOR EACH FILE TO BE USED IN THIS JOB.
)CM
)CM ***** BEGIN DD STATEMENT LOOP *****
)CM DOT M9GDFTB
)CM
)CM CHECK IF DD STATEMENT IS FOR M5LIST
)CM
)CM )SEL &GSDDNAM = M5LIST

```

Figure F-2 M9GCTSBG - VISION:Transact Batch Job Submission Skeleton (Page 1 of 3)

```

)CM
)CM SET DEFAULT FOR SYSPRINT DD IN LINK STEP
)CM
)SET M5SYSOUT = A
)SEL &GSUNIT = SYSOUT
)SET M5SYSOUT = &GSDSN
)ENDSEL
)ENDSEL
)CM
)CM CHECK IF DD STATEMENT IS FOR M5PUNCH
)CM
)SEL &GSDDNAM = M5PUNCH
)SET M5PUNCH = &GSDSN
)ENDSEL
)CM
)CM CHECK IF DD STATEMENT IS FOR SYSOUT DATA SET
)CM
)SEL &GSUNIT = SYSOUT
//&GSDDNAM DD SYSOUT=&GSDSN
)ENDSEL
)CM
)CM GENERATE DD STATEMENTS FOR NON-SYSOUT DATA SETS
)CM
)CM DEFAULT UNIT TO NULL, IF SPECIFIED, SET IT
)CM
)SEL &GSUNIT ^= SYSOUT
)SET M5UNIT1 = &Z
)SEL &GSUNIT ^= &Z
)SET M5UNIT1 = UNIT=&GSUNIT
)ENDSEL
)CM
)CM DEFAULT VOLSER TO NULL, IF SPECIFIED, SET IT
)CM
)SET M5VSR1 = &Z
)SEL &GSVOLSER ^= &Z
)SET M5VSR1 = VOL=SER=&GSVOLSER
)ENDSEL
)CM
)CM DEFAULT ALLOCATION TO NULL, IF SPECIFIED, SET IT
)CM
)SET M5ALL1 = &Z
)SEL &GSALLOC ^= &Z
)SET M5ALL1 = SPACE=(&GSALLOC)
)ENDSEL
)CM
)CM CHECK COMMA PLACEMENTS
)CM
)SET M5C1 = &Z
)SET M5C2 = &Z
)SEL M5VSR ^= &Z | M5UNIT ^= &Z
)SET M5C1 = ,
)ENDSEL
)SEL M5UNIT ^= &Z
)SET M5C2 = ,
)ENDSEL
)CM
)CM USE THIS DD IF DD = DUMMY
)CM
)SEL &GSDSN = DUMMY
//&GSDDNAM DD DUMMY
)ENDSEL
)CM
)CM USE THIS DD IF ALLOCATION, VOLSER, OR UNIT IS SPECIFIED
)CM
)SEL &GSDSN ^= DUMMY
)SEL &M5ALL1 ^= &Z | &M5VSR1 ^= &Z | &M5UNIT ^= &Z
//&GSDDNAM DD DSN=&GSDSN,DISP=(&GSDISP),
)SEL &M5ALL1 ^= &Z
//      &M5ALL1&M5C1
)ENDSEL
)SEL &M5VSR1 ^= &Z
//      &M5VSR1&M5C2
)ENDSEL
)SEL &M5UNIT1 ^= &Z
//      &M5UNIT1

```

Figure F-2 M9GCTSBG - VISION:Transact Batch Job Submission Skeleton (Page 2 of 3)

```
)ENDSEL
)ENDSEL
)ENDSEL
)CM
)CM  USE THIS DD IF ALLOCATION, VOLSER, AND UNIT ARE NOT SPECIFIED
)CM
)SEL  &GSDSN ^= DUMMY
)SEL  &M5ALL1 = &Z && &M5VSR1 = &Z && &M5UNIT = &Z
//&GSDDNAM DD DSN=&GSDSN,DISP=(&GSDISP)
)ENDSEL
)ENDSEL
)CM
)CM  END LOOP ON NON-SYSOUT DATA SET
)CM
)ENDSEL
)CM
)CM  END LOOP ON TABLE
)CM
)ENDDOT
)CM
)CM  SEE IF LINK IS REQUIRED
)CM
)SEL  &GCDDRTYP = A | &GCDDRTYP = G
)SEL  &GCUSLNK = Y | &GCUSLNK = YES
//*
//LINK EXEC PGM=IEWL,REGION=2M,
//      PARM='LET,LIST,XREF,RENT,REUS'
//SYSPRINT DD &M5SYSOUT
//SYSMOD DD DSN=&GCUSLKL2,DISP=SHR
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSLIB DD DSN=&GCUSMKI2,DISP=SHR
//      DD DSN=&GCUSMNI2,DISP=SHR
//IMSLIB DD DSN=&GCUSMNI2,DISP=SHR
//SYSLIN DD DSN=&M5PUNCH,DISP=SHR
)ENDSEL
)ENDSEL
)CM
)CM  ADD EOJ STATEMENT
)CM
//
```

Figure F-2 M9GCTSBG - VISION:Transact Batch Job Submission Skeleton (Page 3 of 3)

M9GCTPU1

```

)ATTR
+ TYPE (TEXT)      INTENS (LOW)  SKIP (&SKIPVAR)
% TYPE (TEXT)      INTENS (HIGH) SKIP (&SKIPVAR)
TYPE (INPUT)       INTENS (HIGH) CAPS (ON)  JUST (LEFT)
@ TYPE (INPUT)     INTENS (LOW)  CAPS (ON)  JUST (LEFT)  PADC (&PADVAR)
} TYPE (OUTPUT)    INTENS (HIGH)  SKIP (ON)
{ TYPE (OUTPUT)    INTENS (LOW)   SKIP (ON)
)BODY EXPAND (||)
%USRPNEL --- &GVITEMLM -|-|
%COMMAND ==> _ZCMD
+
+ENTER%END+TO PROCESS USING THE OPTION PREVIOUSLY ENTERED.
+ENTER%CANCEL+TO TERMINATE PROCESSING THIS MEMBER.
+
+
%ENTER THE NAME OF THE VISION:TRANSACT GEN LIBRARY:
+VISION:TRANSACT GENLIB ==> _GCUSVL1 +
+
%LINK OBJECT FILE? ==> Z + (YES OR NO)
+ LINK TO ==> _GCUSLKL1 +
+ INCLUDES FROM:
+ VISION:TRANSACT ==> _GCUSMKI1 +
+ MONITOR ==> _GCUSMNI1
)INIT
.HELP = M9GCTPH0
.ZVARS = ' (GCUSLNK) '
)PROC
VPUT (GCUSVL1 GCUSLNK GCUSLKL1 GCUSMKI1 GCUSMNI1) PROFILE
)END

```

Figure F-3 M9GCTPU1 - VISION:Transact Foreground Job Execution User Panel

M9GCTSFG

```

)CM
)CM THIS IS A SAMPLE ISPF FILE TAILORING SKELETON FOR USE WITH
)CM WORKBENCH RELEASE 6.1. IT WILL GENERATE CLISTS FOR A TRANSACT
)CM FOREGROUND RUN (APPGEN, DEF RUN, OR SSR). THIS FILE TAILORING
)CM SKELETON IS INTENDED TO BE USED WITH THE SAMPLE USER PANEL
)CM 'M9GCTPU1'. THIS PANEL HAS BEEN PROVIDED IN YOUR WORKBENCH
)CM PANEL LIBRARY.
)CM
)CM THERE ARE 4 TYPES OF VARIABLES (WORDS PRECEDED BY AMPERSANDS) USED
)CM IN THIS SKELETON INCLUDING:
)CM   - VARIABLES FROM THE USER PANEL
)CM     YOU CAN CHANGE THESE
)CM   - VARIABLES SET BY WORKBENCH
)CM     YOU CANNOT CHANGE THESE
)CM   - ISPF SYSTEM VARIABLES
)CM   - LOCAL VARIABLES THAT ARE SET AND USED DURING FILE TAILORING
)CM
)CM THE VARIABLES FROM THE USER PANEL INCLUDE:
)CM VARIABLE          USAGE
)CM
)CM &GCUSVL1           VISION:TRANSACT GEN LIBRARY
)CM &GCUSLNK           IF THE OBJECT FILE SHOULD BE LINKED
)CM &GCUSLKL1          LINK LIBRARY
)CM &GCUSMKI1          VISION:TRANSACT INCLUDE LIBRARY
)CM &GCUSMNI1          MONITOR INCLUDE LIBRARY
)CM
)CM WORKBENCH RESERVED VARIABLE NAMES INCLUDE:
)CM VARIABLE          USAGE
)CM
)CM &GSDDNAM           DATA DEFINITION NAME
)CM &GSDSN             DATA SET NAME
)CM &GSDISP            DATA SET STATUS AND DISPOSITION
)CM &GSVOLSER          VOLUME SERIAL NUMBER
)CM &GSUNIT            UNIT TYPE
)CM &GSALLOC           FILE ALLOCATION
)CM
)CM ISPF SYSTEM VARIABLE NAMES INCLUDE:
)CM VARIABLE          USAGE
)CM
)CM &Z                 A VARIABLE WHOSE VALUE IS NULL
)CM
)CM
)CM
)CM PROC 0
)CM CONTROL LIST MSG
)CM /*
)CM /* ALLOCATE FILES
)CM /*
)CM
)CM LOOP THRU THE TABLE OF FILES TO BE USED IN THIS APPLICATION
)CM AND ALLOCATE EACH FILE AS SPECIFIED
)CM
)CM DOT M9GDFTB
)CM
)CM SAVE M5LIST DSN SO IT CAN BE ALLOCATED AS SYSLST IN THE LINK STEP
)CM
)CM SEL &GSDDNAM = M5LIST
)CM SET M5LIST = A
)CM SEL &GSUNIT = SYSOUT
)CM SET M5LIST = &GSDSN
)CM ENDSEL
)CM SEL &GSDSN = *
)CM SET M5LIST = *
)CM ENDSEL
)CM
)CM
)CM SAVE M5PUNCH DSN SO IT CAN BE ALLOCATED AS SYSLIN IN THE LINK STEP
)CM
)CM SEL &GSDDNAM = M5PUNCH
)CM SET M5PUNCH = &GSDSN
)CM ENDSEL

```

Figure F-4 M9GCTSFG - VISION:Transact Foreground Job Execution Skeleton
(Page 1 of 3)

```

FREE FI(&GSDDNAM)
)CM
)CM ALLOCATE SYSOUT DATA SETS
)CM
)SEL &GSUNIT = SYSOUT
)SEL &GSDSN ^= *
ALLOC FI(&GSDDNAM) SYSOUT(&GSDSN)
)ENDSEL
)SEL &GSDSN = *
ALLOC FI(&GSDDNAM) DA(&GSDSN)
)ENDSEL
)ENDSEL
)CM
)CM ALLOCATE NON-SYSOUT DATA SETS.
)CM CREATE UNIT, DSN, AND VOLUME STRINGS.
)CM
)SEL &GSUNIT ^= SYSOUT
)SET M5UNIT1 = &Z
)SET M5DSN1 = &Z
)SET M5VOL1 = &Z
)SEL &M5UNIT ^= &Z
)SET M5UNIT1 = UNIT(&GSUNIT)
)ENDSEL
)SEL &GSDSN ^= &Z
)SET M5DSN1 = DA(&GSDSN)
)ENDSEL
)SEL &M5VOL ^= &Z
)SET M5VOL1 = VOLUME(&M5VOL)
)ENDSEL
)CM
)CM ALLOCATE NON-SYSOUT DATA SETS
)CM
ALLOC FI(&GSDDNAM) &GSALLOC &GSDISP &M5UNIT1 &M5VOL1 +
&M5DSN1
)ENDSEL
)ENDDOT
)CM
)CM ALLOCATE STANDARD DATA SETS
)CM
/*
/* EXECUTE VISION:TRANSACTION
/*
SET &&M5 = &&STR(&GCUSVL1)
IF &&SUBSTR(1:1,&&M5) = &&STR('') THEN DO
    SET &&M5 = &&SUBSTR(2:&&LENGTH(&&M5)-1,&&M5)
    SET &&M5 = &&STR(''&&M5.(MARKV)')
END
ELSE DO
    SET &&M5 = &&STR(&&M5.(MARKV))
END
CALL &&M5
)CM
)CM FREE DATA SETS
)CM
)DOT M9GDFTB
    FREE FI(&GSDDNAM)
)ENDDOT
)CM
)CM CHECK FOR LINKEDIT STEP
)CM
)SEL &GCDDRTYP = A && &GCUSLNK = YES
/*
/* LINKEDIT STEP
/*
IF &&LASTCC = 0 THEN
FREE FI(SYSPRINT SYSLMOD SYSUT1 SYSLIB IMSLIB SYSLIN)
)SEL &M5LIST = *
ALLOC FI(SYSPRINT) DA(&M5LIST)
)ENDSEL
)SEL &M5LIST ^= *
ALLOC FI(SYSPRINT) SYSOUT(&M5LIST)
)ENDSEL
ALLOC FI(SYSLMOD) DA(&GCUSLKL1) SHR
ALLOC FI(SYSUT1) UNIT(SYSDA) SPACE(1 1) CYLINDERS

```

Figure F-4 M9GCTSFG - VISION:Transact Foreground Job Execution Skeleton
(Page 2 of 3)

```
ALLOC FI(SYSLIB) +  
DA(&GCUSMKI1 +  
&GCUSMNI1) SHR  
ALLOC FI(IMSLIB) DA(&GCUSMNI1) SHR  
ALLOC FI(SYSLIN) DA(&M5PUNCH) SHR  
CALL 'SYS1.LINKLIB(IEWL)' 'LET,LIST,XREF,RENT,REUSE'  
FREE FI(SYSPRINT SYSLMOD SYSUT1 SYSLIB IMSLIB SYSLIN)  
)ENDSEL  
END
```

Figure F-4 M9GCTSFG - VISION:Transact Foreground Job Execution Skeleton
(Page 3 of 3)

Index

Symbols

&SYSPREF system prefix, 2-30

&SYSUID element entry, 2-27

&SYSUID qualifier, 2-30

Numerics

3270 devices, C-13

3270 printers, C-4, C-13

3290 terminal, 2-30

A

ACCEPT, 2-16

ACCEPT command, 3-2

ACCEPT processing, 1-7

ACCEPT SELECT(NNNNNNNN), 2-18

Advanced Syntax Language (ASL) translator, 2-24

AECICS, B-2

Allocating

 ISPF Log Data Set, 2-36

 new library to ISPLIB, 2-37

 new panel library, 2-36

Allocation requirements, 2-25

APARs, 1-6, 1-7, 2-3, 2-7, 2-11, 2-12, 2-17, 2-18, 2-19,
3-1, 3-3, A-1

 installing, 3-2

APPLY, 2-16

APPLY command, 3-2

APPLY processing, 1-7

APPLY SELECT(NNNNNNNN), 2-17

ASL (Advanced Syntax Language), 2-24

assemble modifications, 2-2

assigning high-level qualifier, 2-1

ASTORZ, C-4

Audience, 1-1

B

BIND, 2-39

bind process, 2-22

BIND REPLACE, 2-22

BL character, 2-32

BMS, C-12

 error display map, B-2

BMS error message map, C-7

BMS map, 2-10

BMS routing, C-7

BUILDER.BL140.SMPE.T.BLSYSL, 2-26

C

C runtime, 2-12

CA-Librarian, 2-39, B-1

- CA-LMP
 - Execution Key information, 1-4
- CA-LMP facility, 1-4
- CA-Panvalet, 2-39, B-1
- CAIRIM, 1-4
 - CA-LMP facility, 1-4
 - GenLevel 9212, 1-4
- Called programs, C-9
- Carriage control, C-10
- CDDX078, 2-11
- CEEUOPT, 2-8, 2-9, B-2, C-1
- CEEUOPT WORKLIB member, 2-2
- check list, 2-1, 2-2
- CICS, 1-7, 2-1, 2-3, 2-9, 2-20
 - 3270 application, 2-2
 - application entry module, B-2
 - COMMAREA program, B-2
 - Executor load library, 1-3
 - instructions, C-3
 - Interface Module, 2-8
 - macro library, 2-15
 - runtime error message module, B-1
 - runtime trace message module, B-2
 - services, 2-8
 - startup JCL, 2-3
 - System Definition, 2-20
 - system load library, 2-15
 - Terminal program, B-2
 - Toolkit load library, 1-3
 - Toolkit source library, 1-3
 - Transaction Dump, 3-4
- CICS Executor, 2-11
- CICS Executor library, 2-10
- CICS Executor load library, 2-4
- CICS Toolkit, 2-12
- CICS Toolkit load library, 2-4
- CICS Toolkit source library, 2-4
- CICS/VS executor
 - called programs, C-9

- CICSKEY, C-5
- CICSLIB, 2-3, 2-20
- CICSTKL, 2-20
- CLASM1, 2-2, 2-7, B-1
- CLEAR key, C-5
- CLIST, 2-20, 2-39, D-1
- CLIST library, 2-25
- COBOL, 2-1
 - utility, 2-3
- COBOL Quick Start utility, B-1
- COMLIB, 2-12, 2-34, 3-1, B-1, C-8, C-17
 - installation parameters, C-1
 - load libraries, 2-7
- COMLIB sub-component parameter module, 2-7
- COMMAREA programs, 2-8
- Common Services CAIRIM, 1-4
- COMPRESS, C-17
- Compression Flag, C-17
- Computer Associates
 - Total License Care (TLC), 1-8
- Connector entry, C-6
- CONREQ, C-6
- contacting, 1-8
 - Total License Care (TLC), 1-8
- copy installation tape files to disk, 2-4
- Copying unprocessed panels, 2-37
- COPYTAPE, 2-2, 2-5, A-1
- create backup copy, 2-7
- CSI, 2-3
- CUSTNAM, C-6
- Customizing job submission skeletons, 2-38

D

- DA names, 2-15
- data set characteristics, 2-27

data sets allocated, 2-13
database administrator, 2-21, 2-22
DATE flag, C-10
DB2, 2-1
 database access, 2-20
 database administrator, 2-21, 2-22
 DBRM utility, 2-3
 preprocessor, 2-22
DB2 Database product, 2-15
DB2 Quick Start utility, B-1
DB2I (DB2 Interactive), 2-22
DBRM, 2-3, 2-39
DDDEF entries, 2-15
DECCHAR, C-6
DEF, 2-24
Default terminal device, C-6
DEFINE, 2-13
Definition Processor, 1-7
DEFTERM, C-6
DEFTLIB, 2-27
DELIM, C-7
DEV statement, C-13, C-14
DEVICE ID name, C-2, C-3, C-4
Device type, C-13
DFHRPL concatenation, 2-3
DIF definition, C-4
DIRBLK, C-17
Distribution Zone, 2-3
DL/I database interface, 2-8, 2-9
documentation
 Advantage VISION
 Transact for CICS Environment Guide, C-5
 Transact Getting Started, 2-39
 Advantage VISION:Transact for CICS Getting Started, 1-7
 Advantage VISION:Transact for CICS Toolkit Reference Guide, 2-2, 2-10

CA Common Services for z/OS and OS/390
 Getting Started, 1-4
IBM DATABASE 2 Application Programming
 Guide manuals, 2-21
IBM DATABASE2 Application Programming
 Guide, 2-22
IBM ISPF Dialogue Management Guide and
 Reference manual, 2-28, 2-29
ISPF Dialogue Management Guide and
 Reference manual, 2-39
Unicenter TNG Framework for z/OS Installation
 and Maintenance Guide, 1-5

DOF definition, C-4
DSNLOAD, 2-15
DUMP, 2-24
DVFEAT, C-14
DVMISC, C-14
DVPAGE, C-14
DVWIDTH, C-14
Dynamic loading, C-12

E

EGCS attribute, C-14
ERRDEST, C-7
ERRDEST keyword, 2-10
error messages, 2-9
Executor, 2-1, 2-12, 2-20, 2-24, 3-1, B-2, C-5
 LE user options/parameters, C-1
 problem reporting, 3-4
Extended color attribute, C-13
Extended highlighting attribute, C-13

F

Field designator, C-7
Field overflow, C-10
File Definition Conversion utilities, 2-39
file definition conversion utilities, 2-1

file tailoring output data set, 2-28
FIVCE07, 2-10, B-2
FIVCE07 WORKLIB member, 2-2
FIVCM078, 2-9, B-1
FIVCM078 WORKLIB member, 2-2
FIVCT078, 2-9, B-2
FIVCT078 WORKLIB member, 2-2
FIVEPARM, 2-7, 2-10, 2-35, B-2, C-1, C-2, C-3
FIVEPARM WORKLIB member, 2-2
FIVESQL, 2-21, 2-22, B-2
FIVESQL module, 2-3
FLDDSGN, C-7

G

GE, C-6
GEN, 2-24
General System Modification (GSM), 2-17
Generate, 2-27
Generator, 2-1, 2-10, 2-11, 2-12, 2-20, 2-21, 2-22, 2-24, 2-26, 3-1, B-2, C-17
 COMLIB installation parameters, C-1
 component parameter modules, B-2
 DB2 Interface module, B-2
 DB2 plan, B-2
 installation parameters, C-1
 LE options/parameters, C-1
 load library, 2-4
 problem reporting, 3-3
Generator component, 2-7
Generator load library, 1-3
GenLevel 9212, 1-4
GENLIB, 2-20
GENLIB load library, 2-7
Grouping character, C-7
GRPCHAR, C-7
GSM (General System Modification), 2-17

GT, C-6

H

Help Primary Command, 3-5
high-level qualifier, 2-1
Highlight character, C-7
HILDSGN, C-7

I

IDCAMS, 2-13
IDCAMS control statements, 2-13
IEBCOPY, 2-11, 2-20
IEBCOPY job, 1-6
IGNORE keyword, C-15
Implicit temporary field, C-11
IMPORT function, 2-27
IMPORT option, 2-27
IMS
 Executor load library, 1-3
 instructions, C-4
 Toolkit load library, 1-3
 Toolkit source library, 1-3
IMS Database, 2-15
IMS Executor, 2-11
IMS executor, C-12
IMS Executor load library, 2-4
IMS Toolkit, 2-12
IMS Toolkit load library, 2-4
IMS Toolkit source library, 2-4
IN character, 2-32
Indirect Libraries, 1-7
INFORM.R40.TARGET.INTLOAD, 2-26
INIT, 2-13, 2-24
install PTFs, 2-3
Installation

- data sets, 2-6
- installation
 - basic, 1-6
 - check list, 2-1, 2-2
 - coding and integrating your licensing key, 1-4
 - licensing requirements, 1-4
 - load system tape, 1-6
 - overview, 1-3
 - setups, 1-7
 - SMP/E setup, 1-6
 - tape contents review, 2-4
 - verify, 2-1
- installation tape
 - copying, 2-1
- Installation Verification Process (IVP), 1-7
- installing, 2-4
 - data sets, 2-5
 - JCL samples, 2-5
 - tape contents, 2-4
- internal work files, 2-29, 2-30
- INVALID, C-8
- Invalid field character, C-8
- ISPF
 - Copy utility, 2-37
 - file-tailoring skeletons, 2-26
 - log, 2-37
- ISPF LIBDEF service, 2-25, 2-28
- ISPF library concatenations, 2-25
- ISPF primary menu, 2-31, E-1
- ISPF primary menu panel, 2-34
- ISPF startup CLIST, 2-25, 2-27, 2-29, 2-31, 2-33
- ISPF Workbench, 2-1, 2-12, 2-20, 2-25, 2-26, 2-27, 2-29, 2-30, 2-31, 2-32, 2-33, 2-34, 2-35, 2-36, 2-37, 2-38, 3-1, 3-5, E-1
 - libraries, D-1
 - library allocations, 2-27
 - load library, 2-28
 - operational libraries, B-2
 - optional setup, 2-35
 - panel library, 2-26

- panels, 2-36
- problem reporting, 3-4
- runtime data sets, 2-3
- Selection Menu panel, E-2
- start method, 2-3
- subsection menu, 2-34
- ISPF/PDF environment, 2-33
- ISPF/PDF Facility, 2-25
- ISPF/PDF facility, 2-31
- ISPF FILE, 2-27, 2-28
- ISPLLIB, 2-25, 2-26, 2-28
- ISPMLIB, 2-26
- ISPPLIB, 2-26, 2-37
- ISPPREP, 2-35
- ISPPREP panel, 2-36
- ISPPREP utility, 2-37
- ISPSLIB, 2-26
- ISPSTART command, 2-31, 2-33
- ISPSTART command specifications, 2-34
- IT statement, 2-24
- Item Tracking, 2-24
- Item tracking flag, C-17
- ITEMTRAK, C-17
- IVP (Installation Verification Process), 1-7

J

- JCLIN, 2-11
- Job submission skeletons, 2-38
- job submission skeletons, 2-3

K

- Keep option, 2-39
- KEYS member, 1-5

L

Language Environment, C-1
 Assembler macro library, 2-15
Language Environment (LE), 2-24
Language Environment parameter module, 2-7
LE, C-6
LE (Language Environment), 2-24
Left hand delimiter, C-8
LEFTMRK, C-8
LIBOPNA, C-8
libraries, 2-4
 CICS Executor load, 1-3
 CICS Toolkit load, 1-3
 CICS Toolkit source, 1-3
 Generator load, 1-3
 IMS Executor load, 1-3
 IMS Toolkit load, 1-3
 IMS Toolkit source, 1-3
 SMP/E control, 1-3
 Source and JCL, 1-3
 Workbench for ISPF CLIST, 1-3
 Workbench for ISPF JCL skeleton, 1-3
 Workbench for ISPF load, 1-3
 Workbench for ISPF messages, 1-3
 Workbench for ISPF panel, 1-3
License Management Program (LMP), 1-2
licensing, 1-8
licensing (international), 1-8
licensing (U. S.), 1-8
licensing key, 1-4
licensing requirements, 1-4
Link edit called programs, C-9
linkedit modifications, 2-2
LIST, 2-24
List Data Set, 2-29
LKEDCIS, C-9
LKEDIMS, C-9

load system tape, 1-6
LOADLIB, 2-20
Logical expression, C-6
Logical operators
 EQ, C-6
LSTHGHT, C-9
LSTWDTH, C-9
LT, C-6
LTERM, C-7

M

M4PARAMS, 2-35
M5INPUT, 2-24
M5LEPARM, 2-7, B-2, C-1
M5LEPARM WORKLIB member, 2-2
M5LIB, 2-24
M5LIST, 2-30
M5PARM, C-12
 ASTORSZ, C-4
 CICSKEY, C-5
 CONREQ, C-6
 CUSTNAM, C-6
 DATFM, C-6
 DECCHAR, C-6
 DEFTERM, C-6
 DELIM, C-7
 ERRDEST, C-7
 FLDDSGN, C-7
 GRPCHAR, C-7
 HILDSGN, C-7
 INVALID, C-8
 keyword parameters, C-3
 LEFTMRK, C-8
M5TERMDV, C-2, C-3, C-4, C-6
 keyword parameters, C-12
M5TERMDV parameters, C-12
M9BGTS, 2-38

M9BGUPNL, 2-38
M9BOOT, 2-33
M9DATPMI, 2-30
M9DATPMV, 2-30
M9FGTS, 2-38
M9FGUPNL, 2-38
M9FTOUT, 2-28
M9GCTPU1, 2-38, F-5
M9GCTPU2, 2-38, F-1
M9GCTSBG, 2-38, F-2
M9GCTSFG, 2-38, F-6
M9LST1, 2-30
M9LST2, 2-30
M9LST3, 2-30
M9LST4, 2-30
M9LSTn, 2-30
M9PRIM, 2-32, 2-33, 2-34, E-2
MARKLIBP, 2-7, 2-24, C-1, C-17
MARKLIBP WORKLIB member, 2-2
MCS (Modification Control Statements), 1-7, 2-16
MCS member, 2-11
Message
 MK5CG01, 3-3
 MK5CG02, 3-3
message IEW2646W, 2-21
Message processing option, C-10
MFS, C-12
 DEV statement, C-13, C-14
 MSG statement, C-14
MID, C-15
MIDIGN, C-4, C-15
MIDNMOD, C-4, C-15
MINCMPSZ, C-17
Minimum Compress Size, C-17
Missing field character, C-10
MK5CG01, 3-3

MK5CG02, 3-3
MNHTTAB, C-10
MOD, C-14, C-15
Modification Control Statements (MCS), 1-7, 2-16
Modifying panels after preprocessing, 2-37
MODIGN, C-4, C-15
MODPAGE, C-4, C-14
Month abbreviations, C-10
MSG statement, C-14, C-15
MSGPROC, C-10
MSGS, 2-20
MVS services, 2-28
MZCICS, 2-8, 2-9, B-2
 relevant source code, 2-8
MZCICS WORKLIB member, 2-2
MZCOB, B-2
MZCOMM, 2-8, 2-9, B-2
MZCOMM WORKLIB member, 2-2

N

NE, C-6
NOEXIST, C-10
Number of PF keys, C-13
NUMPFK, C-13

O

OPTLIB data set, 1-5
 KEYS member, 1-5
OPTLIB DD statement, 1-4
Overflow field character, C-10
OVERFLW, C-10

P

PA key, C-5

Panel

- M9BGUPNL, 2-38
- M9FGUPNL, 2-38
- M9GCTPU1, 2-38
- M9GCTPU2, 2-38
- PANELID command, 3-4
- PANELS, 2-20
- Parameter Module
 - source code members, C-1
- PARM entry, 2-35
- PCB, C-7
- PEND statement, 2-5
- Performing SMP/E installation steps, 2-11
- PF keys, C-13
- PR statements, C-6
- preprocess utility
 - running, 2-36
- Primary parameter module, 2-7
- Print lines, C-9
- Print positions, C-9
- Problem reporting, 3-3
- PROC section, 2-32, 2-33
- Process List Data Set panel, 2-30
- product licensing, 1-8
- Product tailoring steps, 2-7
- Programmed symbol, C-14
- PRTCTL, C-10
- PSW value, 3-5
- PTFs, 1-6, 1-7, 2-3, 2-7, 2-11, 2-12, 2-17, 2-18, 2-19, 3-1, 3-3, A-1
 - apply in ascending sequence, 2-19
 - install, 2-3
 - installing, 3-2

Q

QSCOBEX, B-1

QSCOBLL, B-1
QSCOBPL, B-1
QSDB2BND, 2-39, B-1
QSDB2EX, B-1
QSINQEX, B-1
QSRESEX, B-1
QSRESLL, B-1
QSRESPL, B-1
quick start utilities, 2-39
Quick start utility, B-1

R

RC statement, C-5, C-8, C-9
REBIND, 2-22
RECEIVE, 2-16
RECEIVE command, 3-2
RECEIVE operation, 1-6
REJECT command, 3-2
Relative Files, 1-7
REPLACE, 2-20, 2-21
Reporting problems, 3-3
Request processing statements, C-6
RESERVE, C-17
RESLIB, 2-15
RESTORE, 1-7
RESTORE command, 3-2
Restricted System Modification (RSM), 2-17
Restricted System Modifications, 3-1
Right hand delimiter, C-10
RIGHTMK, C-10
RMODE conflicts, 2-21
RP statements, C-9
RSMs, 3-1
Running the preprocess utility, 2-36

runtime programs, 2-3

S

SAS, 2-12

Save Statistics option, 2-36

SC statements, C-6

SCEEMAC, 2-15

Screen control statements, C-6

Screen height, C-12

Screen literals, C-7

Screen width, C-12

SCRNHT, C-3, C-4

SCRNWD, C-3, C-4

SCS1, C-4

SCS1 devices, C-13

SCS2, C-4

SCS2 devices, C-13

SD statement, C-2, C-6

SDFHLOAD, 2-15

SDFHMAC, 2-15

SET command, 2-13

SET command parameters, 2-13, 2-15, 2-16, 2-17, 2-18

SETB operand value, 2-8, 2-9

setting up VISION

 Transact, 2-20

setups, 1-7

Single separator, C-10

site ID, 1-8

Skeleton

 M9BGTS, 2-38

 M9GCTSBG, 2-38

 M9GCTSFG, 2-38

Skeleton M9FGTS, 2-38

Skeletons, 2-38

SKELS, 2-20

SL statement, C-7

SL statements, C-7, C-12

SM (System Modification), 2-17

SMP control library file, 2-2

SMP/E

 ACCEPT processing, 1-7

 APPLY processing, 1-7

 Consolidated Software Inventory (CSI) library, 1-6

 control library, 1-3, 2-4

 CSI, 2-13

 data sets, 2-16

 datasets, 2-3

 Distribution zone, 2-3

 framework, 2-7

 Global zone, 2-3

 global zone, 2-17

 installation approach, 1-7

 installation steps, 2-11

 knowledge, 1-1

 target and distribution libraries, 1-6

 Target zone, 2-3

 work data sets, 1-6

SMP/E facility, 1-2

SMP/E setup, 1-6

SMP_CNTL data set, 3-2, A-1

SMPJCLIN job stream, 1-7

SMPJOB01, 2-3, 2-11, 2-13, A-1

SMPJOB02, 2-3, 2-11, 2-15, A-1

SMPJOB03, 2-3, 2-11, 2-16, A-1

SMPJOB04, 2-3, 2-11, 2-16, A-1

SMPJOB05, 2-3, 2-11, 2-16, A-1

SMPMCS0, 2-11

SMPMCS1C, 2-11

SMPMCS1G, 2-11

SMPMCS1I, 2-11

SMPMCS2, 2-12

SMPMCS3, 2-12
SMPMCS4, 2-12
SMPMCS5C, 2-12
SMPMCS5I, 2-12
SMPRPT listing, 2-17, 2-18, 2-19
SMs, 3-1
SNGLSEP, C-10
Source and JCL library, 1-3
Source listing, C-6, C-10
STEPLIB, 2-28
STEPLIB allocations, 2-25
STORCLASS, 2-13
Support, 3-3
SYSCOLUMNS table, 2-21
SYSIBM.SYSCOLUMNS, 2-21
SYSIBM.SYSSYNONYMS, 2-21
SYSPROC, 2-25
SYSSYNONYMS, 2-21
System delimiter, C-7
system link libraries, 2-25
system link pack area, 2-25
System Modification (SM), 2-17
System Modification Program Extended (SMP/E), 1-1
System Modifications, 3-1
system tape, 1-3
system tape unload, 1-6
SYSUDUMP, 3-3

T

Target Zone, 2-3
Teleprocessing environment, C-11
Temporary fields, C-11
TERMDEV, C-13
TERMDEV keyword, C-4

Terminal devices, C-13
TF statements, C-11
TFREQ, C-11
TLC (Total License Care), 1-8
TODAY, C-11
TODAY flag, C-11
Toolkit, 2-1
Toolkit routines, 2-10, 2-20
Total License Care (TLC), 1-8
TPMON, C-11
TR character, 2-32
TR00nnn, 2-12
trace messages, 2-9
TRACICEM, 2-2, 2-9, B-1
TRACICS1, 2-2, 2-9, B-2
TRACICS2, 2-2, 2-9, B-2
TRACICS3, B-2
TRACICTM, 2-2, 2-9, B-2
TRANSACT.CICS.FILEn, 2-4
TRANSACT.SMPE.SMPCNTL, 1-6, A-1
TRANSACT.TR078.SMPCNTL, 2-5, 2-11, 2-13, 2-15, 2-16, 2-17, 2-18
TRANSACT.TR078.SMPCNTL PDS, 2-5
TRANSACT.TR078.TRTGEN, 2-26
TRANSACT.TR078.WORKLIB, 2-20
TRANSACT.TR078.WORKLIB data set, B-1, C-1
TRANSACT.WB061.CLIST dataset, D-1
TRANSACT.WB061.PANELS dataset, E-1
TRANSACT.WB061.SKELS dataset, F-1
TRANSACT.WB061.WBTCLIST, 2-25
TRANSACT.WB061.WBTLOAD, 2-26, 2-35
TRANSACT.WB061.WBTMSGs, 2-26
TRANSACT.WB061.WBTPANEL, 2-26
TRANSACT.WB061.WBTSKELS, 2-26
Transient Data Queue, 2-20

TRASM1, 2-2, 2-7, B-2
TRBMSE, 2-2, 2-10, B-2
TRCOPY1, 2-20, B-2
TRCOPY2, 2-20, B-2
TRCSDUP, 2-3, 2-20, B-2
TRDB2A, 2-3, 2-21, B-2
TRDB2B, 2-3, 2-22, B-2
TRIVP, 2-24, B-2
TSO, 2-25, 2-30, 2-31, 2-33, 2-34
 command line, 2-36
 logon procedure, 2-25, 2-31
TSO terminal monitor program, 2-22

U

Unexpected Error panel, 3-5
UNIT, 2-13
Unprocessed panels
 copying, 2-37
User Code problems, 3-5

V

verify installation, 2-1
VISION:Builder, 2-26, 2-28, 2-31, 2-34, 2-35, 2-38, 3-2
VISION:Inform, 2-26, 2-31, 2-34, 2-35
VISION:Inquiry, 2-1, 2-39
 Quick Start utility, B-1
VISION:Results, 2-1, 2-3, 2-39
 Quick Start utility, B-1
VISION:Transact, 2-1
 components, 2-1
 library, C-8
VISION:Transact development facility, 2-30
VOLUMES, 2-13

W

WB specifications, 2-32
WB00nnn, 2-12
WBCOPY1, 2-20, B-2
WBCOPY2, 2-20, B-2
Work library, 2-4
Workbench for ISPF
 CLIST library, 1-3, 2-4
 JCL skeleton library, 1-3
 load library, 1-3, 2-4
 message library, 2-4
 messages library, 1-3
 panel library, 1-3, 2-4
 skeleton library, 2-4
Workbench selection menu, 2-34
WORKLB data set, B-1
WORKLIB data set, 2-39, C-1

X

XCOLOR, C-13
XHIL, C-13
XLOAD, C-12
XPSYM, C-14
XSR@PRIM, 2-32, E-1
XSR@PRIM panel, 2-31

Z

z/OS SMP/E facility, 1-2
ZACCEPT, 2-3, 2-11, 2-17, 2-18, 2-19, 3-2, A-1
ZAPPLY, 2-3, 2-11, 2-17, 2-18, 2-19, 3-2, A-1
ZONEINDEX entry, 2-15
ZRECEIVE, 2-3, 2-11, 2-17, 2-18, 2-19, 3-2, A-1
ZREJECT, 3-3, A-2
ZRESTORE, 3-3, A-2

ISPF Workbench, 2-26