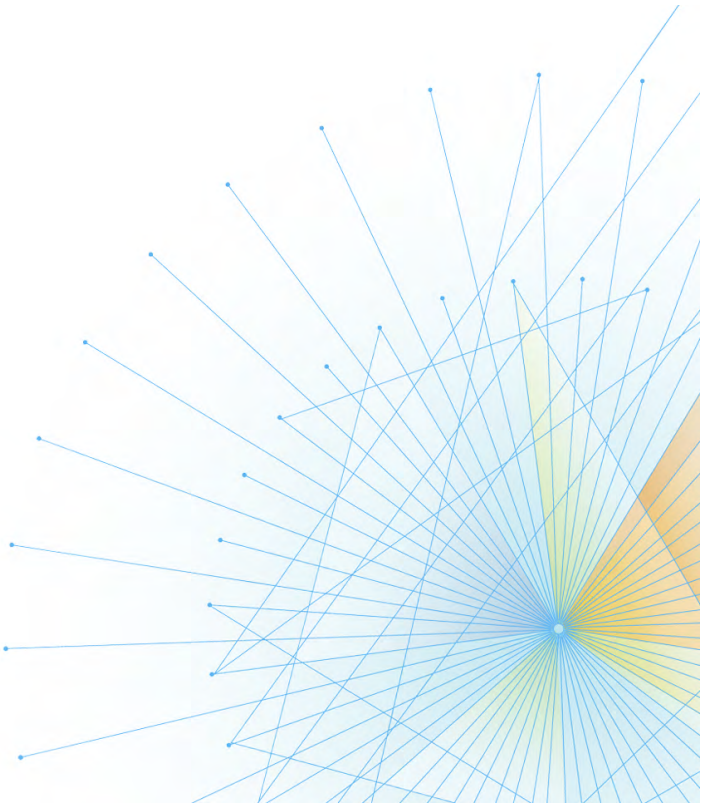
Compuware | The Mainframe Software Partner For The Next 50 Years

Xpediter/CICS Installation and Configuration Guide

Release 17.02



Please direct questions about Xpediter/CICS
or comments on this document to:

Compuware Support Center

https://go.compuware.com/

Doc. APR2019
March 6, 2019

Contents

Introduction

This manual provides information about how to install, customize, and maintain Xpediter/CICS.

Overview

This document is intended to guide you through installing/updating, configuring, deploying, and troubleshooting Xpediter/CICS. Supplemental documentation can be found in the *Xpediter/CICS Advanced Configuration Guide*.

Alerts

The alerts found in this guide include:

A note or tip providing additional information.

If a particular milestone or task doesn't apply to your site—or your site is not licensed for a particular option—you can skip ahead to the next milestone or task by clicking the icon.

Information important to remember.

Caution. Failure to follow these instructions can cause problems.

Indicates which skill set is most likely needed to perform the following task(s).

Additional Resources

Refer to these other sources of information on Xpediter/CICS.

Related Publications

- *Compuware Installer Mainframe Products SMP/E Installation Guide*
- *Xpediter/CICS Advanced Configuration Guide*
- *Xpediter/CICS Release Notes*
- *Xpediter/CICS User Guides for Assembler, C, COBOL, or PL/I*

- *Xpediter/CICS Reference Manual*
- *Xpediter/CICS Quick Reference*
- *Xpediter/CICS Messages and Codes*.

Online Documentation

The Xpediter/CICS product installation package does not include the product documentation. Access the Xpediter/CICS documentation from the Compuware Support Center website at **https://go.compuware.com** in the following electronic formats:

- Release Notes in HTML format
- Product manuals in PDF format
- Product manuals in HTML format.

The product documentation is available for viewing or downloading:

- View PDF files with the free Adobe Reader, available at **http://www.adobe.com**.
- View HTML files with any standard web browser.

Notation Rules

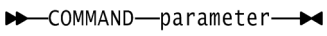
Syntax diagrams define primary command syntax.

A **parameter** is either a keyword or a variable.

All KEYWORDs are shown in uppercase characters and must be spelled exactly as shown. You cannot substitute another value. If any part of a KEYWORD is shown in lowercase characters, that part is optional.

Variables are user-specified values and are printed in lowercase italics. For example, *dataset-name* indicates you are to substitute a value.

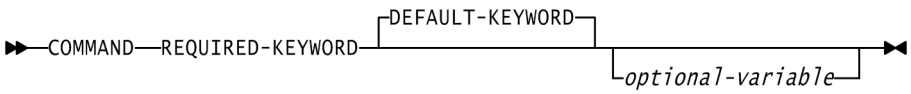
The syntax for commands is described in diagrams that help you visualize parameter use. The following example shows a command and a parameter:



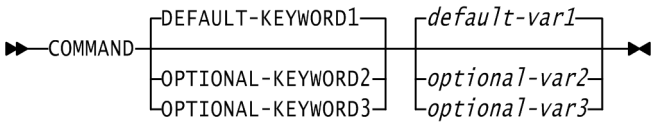
Read the diagrams from left to right and from top to bottom. These symbols help you follow the path of the syntax:

- ▶▶ indicates the beginning of a statement.
- → indicates the statement is continued on the next line.
- ▶ indicates the statement is continued from the previous line.
- →◀ indicates the end of a statement.

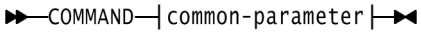
Required parameters appear on the horizontal line (the main path). Optional parameters appear below the main path. Default parameters appear above the main path and are optional. The command executes the same regardless of whether the default parameter is included.



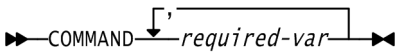
Vertically stacked parameters are mutually exclusive. If you must choose a parameter, one item of the stack appears on the main path. If the parameters are optional, the entire stack appears below the main path. If a parameter in a stack is the default, it appears above the main path.



If the same parameters are used with several commands, their syntax may be documented in a separate diagram. In the command syntax, these common parameters are indicated with separators before and after the parameter name.



An arrow returning to the left indicates a repeatable item. If the arrow contains a comma, separate the repeated items with a comma.



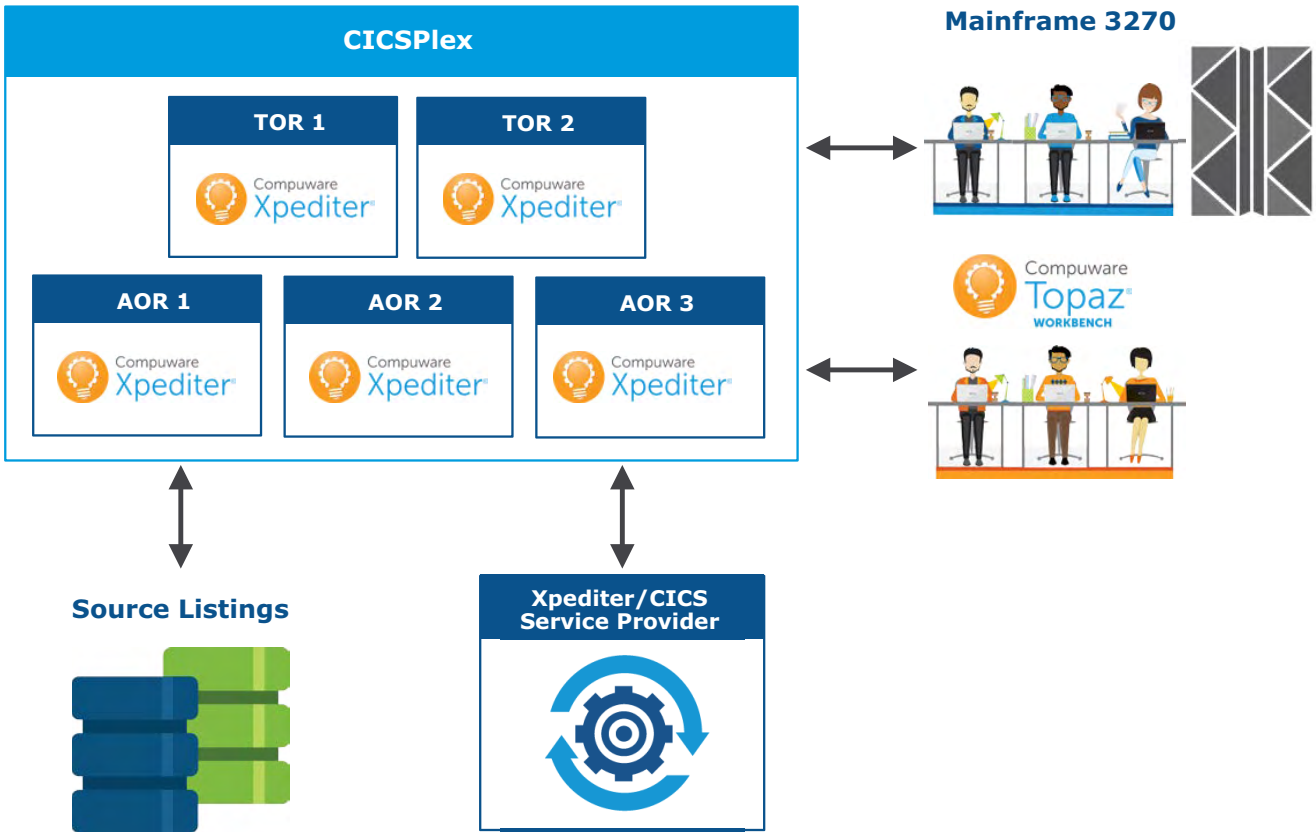
Xpediter/CICS Overview

Xpediter/CICS is the premier product choice for interactive, source-level testing of COBOL, Assembler, C, and PL/I programs. With Xpediter/CICS you can interactively analyze and test at the source code level to understand how programs process data and logic. Programmers can even see the potential impact of program logic changes before any changes are made. Xpediter automates the testing and debugging process within the development life cycle, so your applications move quickly into production - and stay there.

Users drive Xpediter/CICS sessions either through the classic character-based 3270 screens, or through the modernized Eclipse graphical user interface provided by Topaz Workbench. Topaz Workbench is provided at no charge and supports a large subset of Xpediter/CICS functionality.

Product Architecture

Xpediter/CICS is made up of three components: CICS Debug engine, Service Provider, and Eclipse-based modernized alternative user interface.



- CICS Debug engine runs inside of CICS regions and enables debugging of CICS applications programs. Source code support is available for application programs written in COBOL, PL/I, C Language, and HLASM. Along with standard debugging functionality (i.e. setting breakpoints, manipulating variables and memory, etc.), this engine does additional tasks such as CICS storage violation prevention, code coverage statistic collection, and application execution visualization data collection.

- Service Provider is a long running started task that coordinates communications between multiple Xpediter/CICS enabled CICS regions and provides breakpoint management and DTR (dynamic transaction routing) coordination in multi-LPAR CICSPLEXs.
- The Eclipse based interface provides an alternative, modernized user interface to those who prefer to use the product in a Graphical User Interface as opposed to 3270 character based screens. It uses the Eclipse Debugging Framework, so users who are familiar with it (that is, most JAVA developers) will immediately feel at home with this.

Planning

This section provides information related to planning to install or update to Xpediter/CICS 17.02.

Steps Involved

1. Order Xpediter/CICS and its companion products, including the latest maintenance, via Compuware's Product Ordering web page or by contacting Compuware as described in Customer Solutions on page 64.
2. Read this *Installation and Configuration Guide* and complete each of the milestones to:
   a. Ensure companion products have been installed and configured with the latest maintenance and that the license for Xpediter/CICS has been imported.
   b. Perform the SMP/E installation of Xpediter/CICS according to the *Compuware Installer Mainframe Products SMP/E Installation Guide*.
   c. Implement the Compuware PARMLIB.
   d. Configure Xpediter/CICS for either a new installation or an upgrade.
   e. Perform additional configuration for Topaz Workbench and other features.
   f. Verify product installation.
   g. Deploy Xpediter/CICS to additional CICS regions.
   h. Troubleshoot any problems with the installation.

Milestones and Roles

Installation, configuration, verification, and deployment are done in ten milestones. The rows in Table 1 identify the role or skill set required to perform each milestone. This makes it easier to know which people need to be involved at each milestone along the way. With the proper planning, you may be able to have certain tasks performed at the same time.

Table 1. People Required for Each Milestone

Milestone	Companion Product Installer	Xpediter/CICS Installer	z/OS System Programmer	z/OS Security Administrator	CICS System Programmer	Network Administrator	DBA
Milestone 1: Ensure Installation and Configuration of Companion Products	ECC ●	●					
Milestone 2: Install Xpediter/CICS Using SMP/E		●					
Milestone 3: Compuware PARMLIB Implementation		●					
Milestone 4: Configure Xpediter/CICS — New Installation		●	●	●	●		

Table 1. People Required for Each Milestone ***(Continued)***

Milestone	Companion Product Installer	Xpediter/CICS Installer	z/OS System Programmer	z/OS Security Administrator	CICS System Programmer	Network Administrator	DBA
Milestone 5: Configure Xpediter/CICS — Upgrade	Abend-AID for CICS ●	●	●	●	●		IMS ● DB2 ●
Milestone 6: Configure Topaz Workbench Integration	Topaz Workbench ●			●	●	●	
Milestone 7: Specifying Additional Configuration Parameters		●	●				
Milestone 8: Specifying System Facilities and Code Coverage Test Parameters		●					
Milestone 9: Verify Product Installation		●					
Milestone 10: Deployment		●	●		●		

Checklist of Milestones and Tasks

To keep track of your progress, you may want to print the Checklist of Milestones and Tasks on page 65 at the end of this manual, then check off each Milestone and task as it gets completed. (The checklist is only available in the PDF version.)

Prerequisites

Supported Hardware and Software

Hardware Platforms

- z14, z14 ZR1
- z13, z13s
- zEC12, zBC12
- z196, z114
- z10-EC/BC
- z9-EC/BC
- z900, z990
- z800, z890

Operating Systems

- IBM z/OS V2.2, 2.3
- IBM ISPF for the supported z/OS releases

Major Subsystems

- IBM CICS Transaction Server for z/OS V5.1, 5.2, 5.3, 5.4, 5.5
- IBM DB2 for z/OS V11.1, 12.1
- IBM IMS Transaction and Database Servers V14.1, 15.1
- IBM MQ for z/OS V9.0, 9.1

Languages

(Under IBM Language Environment for Supported z/OS Releases and in Compatibility Mode)

- IBM Enterprise COBOL for z/OS V4.2, 5.1, 5.2, 6.1, 6.2
- IBM Enterprise PL/I for z/OS V4.5. 5.1, 5.2
- IBM XL C/C++ for supported z/OS releases. Note: Support is limited to C, not C++.

Non-LE Languages

- IBM High Level Assembler for z/OS, z/VM, and z/VSE V1.6

Milestone 1: Ensure Installation and Configuration of Companion Products

Roles involved:
ECC Installer

Complete the following tasks to install and configure Xpediter/CICS companion products.

Task 1.1 Install/Upgrade Enterprise Common Components

Enterprise Common Components (ECC) version 17.02 or higher must be installed and configured to support Xpediter/CICS 17.02. See the *Enterprise Common Components Installation and Configuration Guide* for instructions on configuring ECC for use with Xpediter/CICS.

Task 1.2 Apply ECC Maintenance

Apply the latest maintenance to ECC 17.02.

Task 1.3 Import Xpediter/CICS License

When you received your Xpediter/CICS product, you also received a license key for it. Import your Xpediter/CICS license into the Compuware License Management System.

Milestone 2: Install Xpediter/CICS Using SMP/E

Roles involved:
Xpediter/CICS Installer

SMP/E Installation

Xpediter/CICS is installed using SMP/E. The milestones in this chapter will guide you through the SMP/E installation of Xpediter/CICS as well as some preparatory tasks you can perform to make configuration of Xpediter/CICS easier.

FMIDs

The FMIDs for each CICS release are listed below:

- QMXD170 - CICS TS 5.1 modules
- RMXD170 - CICS TS 5.2 modules
- SMXD170 - CICS TS 5.3 modules
- TMXD170 - CICS TS 5.4 modules
- UMXD170 - CICS TS 5.5 modules.

If you install two or more CICS-dependent FMIDs for the same release of Xpediter/CICS, there will be an instance of SMXDAUTH, SMXDAAFX, SMXDPDSE, and SMXDSAMP for each FMID. Assuming the same level of maintenance on each FMID, all instances of these datasets are identical. The following example uses SMXDAUTH, but the concept applies equally to SMXDAAFX, SMXDPDSE, and SMXDSAMP.

Consider this example: You presently use the TMXD170 FMID and are installing the UMXD170 FMID for a newer CICS release. Your Xpediter Service Provider has CPWR.TMXD170.SMXDAUTH as the STEPLIB. There is no technical requirement to immediately change the STEPLIB to specify CPWR.UMXD170.SMXDAUTH as the STEPLIB.

There may be an operational reason to transition to use of the UMXD170 instance of SMXDAUTH, at some convenient time. Once all CICS regions have been upgraded to the newer CICS release, you may want to archive or delete the TMXD170 datasets. Obviously, that archive or delete cannot be completed if the TMXD170 instance of SMXDAUTH is still in use.

If you are installing Xpediter/CICS with support for more than one release of CICS, which may include releases your site is planning to install in the future, Compuware recommends installing them simultaneously.

Complete the following task to SMP/E-install Xpediter/CICS.

Task 2.1 Follow the Compuware Installation Guide

1. Follow the instructions in the *Compuware Installer Mainframe Products SMP/E Installation Guide* to install Xpediter/CICS.

2. Once completed, follow the steps in this guide to configure and deploy Xpediter/CICS.

Milestone 3: Compuware PARMLIB Implementation

This milestone contains tasks that help prepare for configuration of Xpediter/CICS for both a new install and an upgrade.

Roles involved:
Xpediter/CICS Installer

Complete the following tasks to prepare for Xpediter/CICS configuration.

Task 3.1 Implement the Compuware PARMLIB

Starting with release 17.02, Compuware mainframe products, including Xpediter/CICS, use parameter libraries (PARMLIBs) in conjunction with the Compuware Mainframe Services Controller (CMSC) to configure each product as well as common components. The CMSC was installed as part of the Enterprise Common Components installation.

In this task, you will implement the Compuware PARMLIB for Xpediter/CICS. This consists of creating default members and updating the CMSC with them. The sample members are suffixed with `00` and should contain the values you want for all CICS ***test*** regions in which Xpediter/CICS and Xpediter/Code Coverage are installed. The sample members suffixed with `PROD` contain suggested values for CICS ***production*** regions in which Xpediter/CICS and Xpediter/Code Coverage are installed.

This task is necessary for both new installation and an upgrade of Xpediter/CICS to release 17.02.

Whenever you modify an existing Compuware PARMLIB member or add a new member, you will need to use the CMSC REFRESH command to update the contents of the CMSC cache.

Task 3.1.1 Create the Default Members

Copy the associated members from the Xpediter/CICS SMXDSAMP library members into your Compuware PARMLIB dataset. The SMXDSAMP library members are set up for the most common settings in a test region.

Table 1. Compuware PARMLIB Members for Test Regions

Product	Contents	Member Prefix	Default Member	SMXDSAMP Library Members
Xpediter/CICS	Xpediter/CICS Parameters	XDGB	XDGB00	XDGB00, XDGBALL
Xpediter/CICS	Xpediter/CICS System Facilities and Code Coverage Test Parameters	XDDB	XDDB00	XDDB00, XDDBALL
Xpediter/Code Coverage	Xpediter/Code Coverage Parameters	XVGB	XVGB00	XVGB00, XVGBALL

If you are planning to install Xpediter/CICS and Xpediter/Code Coverage for use in a production region, copy the following member from your Xpediter/CICS SMXDSAMP library into your Compuware PARMLIB dataset.

Table 2. Compuware PARMLIB Members for Production Regions

Product	Contents	Member Prefix	SMXDSAMP Library Member
Xpediter/CICS	Suggested Xpediter/CICS Parameter settings for production regions	XDGB	XDGBPROD

Task 3.1.2 Update the CMSC with your PARMLIB Members

Use the z/OS MODIFY (F) command to update the CMSC with the PARMLIB members you created.

Refreshing All PARMLIB Members

```
F cmscname,PARMLIB REFRESH
```

Refreshing a Single Parameter Member

```
F cmscname,PARMLIB REFRESH member_name
```

During the refresh process, parameter values are validated. If they are found to be invalid, an error message is written to the FDBDLOG SYSOUT dataset associated with the ECC CMSC instance where the member is being refreshed. If an error is detected, the contents of the in-core member will not be refreshed. You must correct the error and refresh the member before attempting to use it.

Milestone 4: Configure Xpediter/CICS — New Installation

This chapter will guide you through configuration of a new installation of Xpediter/CICS 17.02.

If you are performing an upgrade instead, skip ahead to Milestone 5: Configure Xpediter/CICS — Upgrade.

Roles involved:
z/OS Security Administrator
Xpediter/CICS Installer
z/OS System Programmer
CICS System Programmer.

Complete the following tasks to configure a new installation of Xpediter/CICS.

Task 4.1 External Security Considerations

Roles involved:
z/OS Security Administrator

Task 4.1.1 Transaction Security

To help you to set up security groups for external security, the following tables list the Xpediter/CICS (and Xpediter/Code Coverage) transactions and their uses. The tables also identify started transactions and provide recommended external security settings.

If you are performing an upgrade of Xpediter/CICS, compare the tables below to your existing transaction security profile and set the appropriate security for any new transactions.

The terms used in the “Security Recommended” column are defined as follows:

- **NO** - All users of Xpediter, including the user ID associated with PLTPI programs, should be permitted to execute these transactions.
- **YES** - These transactions provide capabilities beyond the needs of a typical end user of Xpediter. Access may be limited to those responsible for installation, configuration, etc.
- **EITHER** - No recommendation.

Table 3. Xpediter/CICS Transactions

Trans	Program	Use	Started	Security Recommended
DBFL	DBUGFILE	File utility	●	NO
DBPA	DBUGPPM2	PLT startup	●	NO

Table 3. Xpediter/CICS Transactions ***(Continued)***

Trans	Program	Use	Started	Security Recommended
DBXG	DBUGSP00	Remote traps	●	NO
DMAP	DBUGMAPS	Display BMS maps		EITHER
NEWC	DBUGNEWC	Newcopy (PHASEIN) programs		EITHER
XACH	CWDEMACH	Demo program		EITHER
XASL	CWCEEASM	Demo program		EITHER
XASM	CWDEMASM	Demo program		EITHER
XCB2	CWDEMCB2	Demo program		EITHER
XCBN	CWDEMCBN	Demo program		EITHER
XCCC	CWDEMC	Demo program		EITHER
XCCH	CWDEMCCH	Demo program		EITHER
XDBP	DBUGDBPA	Process Xpediter/CICS System Facilities and Code Coverage Test Parameters		YES
XIVP	DBUGINST	Installation Verification Program		EITHER
XLGI	DBUGLOGC	File Utility logging	●	NO
XLOG	DBUGLOGM	Activate, terminate, or switch datasets		YES
XPCH	CWCEMPCH	Demo program		EITHER
XPCI	DBUGCRTL	Remote Operations Command Interface (ROCI)	●	EITHER
XPED	DBUGSP00	Access Xpediter		EITHER
XPFS	DBUGCSFS	SLS Dataset Services	●	NO
XPGD	XDPIMIRS	Secure Xpediter/CICS access from Xpediter/Eclipse		EITHER
XPLE	CWDEMPE	Demo program		EITHER
XPLI	CWDEMPL	Demo program		EITHER
XPN0	DBUGEND0	Automatic session termination	●	NO
XPNC	DBUGCRTL	Newcopy (PHASEIN) program from batch		EITHER
XPND	DBUGEND	Automatic session termination	●	NO
XPOF	DBUGDBXX	Shutdown Xpediter		EITHER
XPRT	DBUGSP00	Access Xpediter		EITHER
XPSP	DBUGSP00	Access Xpediter with system facilities		YES
XPWD	DBUGWRAM	Disable Xpediter's dynamic transaction routing exit		YES
XPWI	DBUGWRAM	Enable Xpediter's dynamic transaction routing exit		YES
XREL	DBUGNEWC	Internal Xpediter transaction		NO
XROI	DBUGROIC	Data collection for Compuware's zAdviser	●	NO
XSIT	DBUGSIT	Process Xpediter/CICS Parameters		YES
XSOC	DBUGSOCK	Xpediter/Eclipse connections		NO
XSTA	DBUGSTA0	Access Xpediter in multiple AORs		EITHER

Table 3. Xpediter/CICS Transactions ***(Continued)***

Trans	Program	Use	Started	Security Recommended
XZCH	CWDEMZCH	Demo program		EITHER

Table 4. Xpediter/Code Coverage Transactions

Trans	Program	Use	Started	Security Recommended
XVCC	XVTCMGR	Code Coverage definition update		YES
XVKP	XVTCEXTR	Internal Code Coverage transaction	●	NO
XVSC	XVTCMGR	Code Coverage definition update		YES
XVSI	XVTCSIT	Process Xpediter/CICS Code Coverage Parameters		YES
XVTQ	XVTCCCIN	Internal Xpediter/Code Coverage quiesce tran ID	●	NO

If external security checking is enabled (SEC=YES and XPCT=YES in the CICS System Initialization Table) for the started transactions identified in the above tables, CICS calls the external security manager using the default CICS resource class name CICSPCT prefixed with A or B. This is done to verify that the user ID associated with the transaction is authorized to use started transactions. Please refer to IBM's Knowledge Center and your external security manual for more information.

- Incorrect security settings for the XPFS transaction can cause a NO SOURCE AVAILABLE condition for all program listings accessed via Shared Directory or DDIO datasets and can also cause the SLS Datasets (9.L) screen to function improperly.
- Incorrect security settings for the XREL transaction can cause erroneous failure of the new copy function for programs where breakpoints have been removed.
- Incorrect security settings for the XSRE transaction can cause erroneous failure of breakpoint, event, and visualization when using Xpediter/Eclipse (Topaz Workbench).
- Incorrect security settings for the XVKP transaction can cause failure of periodic extract of data for Xpediter/Code Coverage.

Started Transaction User Security

The following tables list the recommended user security for Xpediter/CICS and Xpediter/Code Coverage started transactions.

Table 5. User Security for Xpediter/CICS Started Transactions

Trans	Program	Use	UserID
DBFL	DBUGFILE	File utility	Assigned the userID of the terminal accessing the file utility.
DBPA	DBUGPPM2	PLT startup	Assigned the same userID as the CPLT transaction.
DBXG	DBUGSP00	Remote traps	Assigned the userID of the trapping terminal.
XLGI	DBUGLOGC	Activate, terminate, or switch logging	Assigned the userID of the terminal accessing transaction XLOG.
XPCI	DBUGCRTL	Remote Operations Command Interface (ROCI)	Refer to the *Xpediter/CICS Advanced Configuration Guide* for security considerations.
XPFS	DBUGCSFS	SLS Dataset Services	Assigned the userID of the Xpediter/CICS initiator or DFLTUSER. If initiated via PLTPI, assigned PLTPIUSR or the CICS region userID.
XPGD	XDPIMIRS	Xpediter/Eclipse	Refer to Milestone 6: Configure Topaz Workbench Integration.

Table 5. User Security for Xpediter/CICS Started Transactions ***(Continued)***

Trans	Program	Use	UserID
XPN0	DBUGEND0	Automatic session termination	Assigned the userID of the terminal at which it runs (when started via autoinstall terminal delete).
XPND	DBUGEND	Automatic session termination	Same as XPN0.
XREL	DBUGNEWC	Release held programs	Runs when all breakpoints are removed from a program. Assigned the userID of the terminal for which the breakpoints were removed.
XROI	DBUGROIC	Data collection for Compuware's zAdviser	Assigned the userID of the Xpediter/CICS initiator. If initiated via PLTPI, assigned PLTPIUSR or DFLTUSER userID. If initiated via terminal, assigned userID of the terminal.
XSRE	DBUGREMS	Triggers Xpediter/Eclipse remote events	Assigned the userID of the Xpediter/CICS initiator or DFLTUSER.
XTMO	DBUGXTMO	Checks for inactive session termination	Assigned the userID of the Xpediter/CICS initiator or DFLTUSER. If initiated via PLTPI, assigned PLTPIUSR or the CICS region userID.

- If automatic session termination is performed via the XSNOFF global exit, XPN0 and XPND must be secured.
- When Xpediter/CICS starts the DBFL and XREL transactions internally, the current user ID will be propagated to either the DBFL or XREL transactions.

Table 6. User Security for Xpediter/Code Coverage Started Transactions

Trans	Program	Use	UserID
XVKP	XVTCEXTR	Code Coverage auto-extract transaction	Assigned the userID of the Xpediter/CICS initiator or DFLTUSER. If initiated via PLTPI, assigned PLTPIUSR or the CICS region userID.
XVTQ	XVTCCCIN	Code Coverage quiesce transaction	Assigned the userID of the Xpediter/CICS initiator or DFLTUSER. If initiated via PLTPI, assigned PLTPIUSR or the CICS region userID.

Task 4.1.2 Security Exits

If you need to control activity in the Xpediter/CICS File Utility, refer to the section entitled "Implement the File Utility Security Exit" in the *Xpediter/CICS Advanced Configuration Guide*.

Task 4.1.3 INQUIRE ASSOCIATION Considerations

The Xpediter/CICS start-of-task exit uses the EXEC CICS INQUIRE ASSOCIATION call to obtain the client IP address associated with the task. This allows the terminal emulator IP address to be identified when a trap/breakpoint is taken. The ASSOCIATION command is subject to CICS command security checking using the resource identifier (ASSOCIATION). Xpediter/CICS will need a profile created that allows all CICS users to have READ access for resource identifier ASSOCIATION.

Task 4.1.4 Compuware PARMLIB Reports

Xpediter/CICS has replaced the z/OS services (dynamic allocation and file open/write/close) used to produce the Compuware PARMLIB Reports with the use of CICS Extrapartition Transient Data Queues and CICS services. As part of this refactoring, three new TDQ definitions have been included in the SMXDSAMP member CSDXDFIL. Table 7 lists the default TDQ names, the Compuware PARMLIB Report associated with the TDQ, and the PARMLIB overrides used to change the TDQ names if required.

Whether you are performing a new install or upgrading to a new release, these queues should be defined to produce the Compuware PARMLIB reports during PARMLIB processing. Review member CSDXDFIL and add these queues to your RDO definitions. If you decide to modify the queue names,

be aware that you must also add the appropriate parameter changes to your Compuware PARMLIB members. If you decide not to install these TDQ definitions, be aware that:

- Additional messages may be written to the CSMT queue indicating that the TDQ definitions are not present.
- The report produced by the Installation Verification Program (transaction XIVP) will display warning messages for any missing queues.

The global tables supplied with the product have also had their default values for the report ddnames changed to reflect the new TDQ queues. If you install the new TDQ definitions and do not change the default names, you should not encounter any errors.

Compuware PARMLIB parameters that are being deprecated by this change are:

- CODE_COVERAGE_PARMLIB_REPORT_ALLOCATION
- CODE_COVERAGE_PARMLIB_REPORT_SYSOUT_CLS
- DBPA_REPORT_ALLOCATION
- DBPA_REPORT_SYSOUT_CLASS
- PARMLIB_REPORT_ALLOCATION
- PARMLIB_REPORT_SYSOUT_CLASS

The following Compuware PARMLIB parameters are being repurposed from supplying a one- to eight-character ddname to providing a one- to four-character TDQ name (if it is necessary to override the new TDQ names):

- CODE_COVERAGE_PARMLIB_REPORT_DDNAME
- DBPA_REPORT_DDNAME
- PARMLIB_REPORT_DDNAME

The reports produced have had minor cosmetic changes made to better describe what the reports represent and the names of the TDQ that they are written to.

Table 7 Compuware PARMLIB Parameter Reports

Queue Name	Report Produced	PARMLIB	
		Member Prefix	Parameter Override
XGBR	Xpediter/CICS Parameters	XDGB	PARMLIB_REPORT_DDNAME
XDBR	Xpediter/CICS System Facilities and Code Coverage Test Parameters	XDDB	DBPA_REPORT_DDNAME
XVGR	Xpediter/Code Coverage Parameters	XVGB	CODE_COVERAGE_PARMLIB_REPORT_DDNAME

Task 4.1.5 Additional Considerations

The following security considerations are covered in more detail in Milestone 6: Configure Topaz Workbench Integration.

Enabling CICS Transaction for XSKL User ID

The user ID associated with transaction XSKL must be permitted to start a CICS transaction as a surrogate for the Topaz Workbench user (for example: read access to *.DFHSTART in CLASS SURROGAT).

Verifying Transaction Definition for Controlling Access from Topaz Workbench

XPGD is the default transaction code for controlling access to Xpediter/CICS from Topaz Workbench. If XPGD is not already defined, a sample is in SMXDSAMP member CSDXDTRN.

Task 4.2 Prepare the ISPF Edit Macro

Roles involved:
Xpediter/CICS Installer

XDUPDATE is an ISPF edit macro that can be used to automate the entry of site-specific JCL parameters. Using this macro saves you from having to repeatedly type in the same information and ensures your JCL parameters are always entered correctly.

To use the edit macro, first copy member XDUPDATE from the SMXDSAMP member into your CLIST library. Then enter your site-specific information in the XDUPDATE macro. The macro includes instructions for entering the information. Your CLIST library must be allocated to your ISPF session. You can verify that your CLIST library is accessible to ISPF by entering the command TSO LISTA.

The edit macro can be used during Xpediter installation each time you are instructed to edit JCL. To run it, first open the JCL member for editing. You can then look at the JCL before any changes are made. Type XDUPDATE in the COMMAND field of the ISPF EDIT screen and press Enter.

XDUPDATE can also be used to update dataset names in CSDXDFIL.

Task 4.3 Integrate Xpediter/CICS with MVS

Roles involved:
z/OS System Programmer

In this step, you will authorize the Xpediter/CICS target library that ends with the qualifier SMXDAUTH. You will also define the Xpediter Service Provider program properties to MVS and define it as an MVS subsystem.

The Xpediter/CICS APF authorized library should not be included in the operating system's link list. This will allow you to protect the library and ensure only approved users can execute the Xpediter Service Provider.

Task 4.3.1 Authorize the Xpediter SMXDAUTH Library

Ensure the Xpediter/CICS 17.02 SMXDAUTH library is APF authorized. The Xpediter/CICS APF authorized library is backward compatible. Compuware suggests you add the dataset to your IEAAPF*xx* or PROG*xx* list, as appropriate, so it will be APF authorized at each system IPL.

If Xpediter/CICS is deployed to additional LPARs, any copies of the SMXDAUTH library must also be APF authorized.

Task 4.3.2 Define the Xpediter Service Provider Program Properties to MVS

1. Add program DBUGSTC to the MVS program properties table (PPT) member in SYS1.PARMLIB using either of the following methods:

   – Copy SMXDSAMP member SCHEDXD to SYS1.PARMLIB, then add **XD** to the SCH=(xx) directive of IEASYSxx in SYS1.PARMLIB. For example, change SCH=00 to **SCH=(00,XD)**.

- Append SMXDSAMP member SCHEDXD to your SCHEDxx member in SYS1.PARMLIB.

2. Update the MVS PPT by issuing one of the following MVS operator commands:

SET SCH=(xx,XD)	where xx is the suffix of the SCHEDxx member used during the last IPL and XD is the suffix of the SCHEDxx member which contains the MVS PPT entry for DBUGSTC.
SET SCH=xx	where xx is the suffix of the SCHEDxx member which contains the MVS PPT entry for DBUGSTC.

Task 4.3.3 Define the Xpediter Service Provider as an MVS Subsystem

You may optionally specify the Xpediter/CICS subsystem ID in SYS1.PARMLIB(IEFSSNxx) for system documentation purposes using either of the following methods:

- Copy SMXDSAMP member IEFSSNXD to SYS1.PARMLIB, then add XD to the SSN=(xx) directive of IEASYSxx in SYS1.PARMLIB. For example, change SSN=00 to SSN=(00,XD).
- Append SMXDSAMP member IEFSSNXD to your IEFSSNxx member in SYS1.PARMLIB.

You are not required to perform an IPL to use the Xpediter Service Provider. If an MVS subsystem entry has not already been defined, Xpediter/CICS will build one dynamically during initialization.

Make sure the MVS Performance Group used by the Xpediter Service Provider is above CICS and below VTAM to prevent runtime abends.

Task 4.4 Update the CICS Resource Definitions

Roles involved:
CICS System Programmer

This task explains how to update your site's CICS resource definitions. Observe the following general considerations:

- Some Xpediter/CICS control blocks are GETMAINed in accordance with the TASKDATALOC specified for the transaction being tested. Xpediter's use of below the line storage can be reduced by specifying TASKDATALOC=ANY for your eligible transactions.
- Xpediter/CICS does not support programs with RELOAD:YES in the program resource definition. Remove this option for programs that you plan to debug with Xpediter.
- Do not define temporary storage queues with an XP prefix as remote.
- In DFHSITxx, do not specify SRT=NO.
- Unless the documentation packaging utility has been disabled by changing the value of global parameter DOCUMENT_PACKAGING_UTILITY_DESTINATION to NONE, either in DBUGGBL or as a global parameter, specify SPOOL=YES as a SIT parameter.
- If your site will be initializing Xpediter/CICS using PLT startup, observe the following conditions:
  - Add the Xpediter/CICS program DBUGPPM2 to the PLT program initialization (PLTPI) table.
  - Shutdown of Xpediter/CICS via the PLT is supported. To enable this support, add program DBUGPLTS to the PLTSD table. In your CICS shutdown PLT, ***before*** the line reading **DFHPLT TYPE=ENTRY,PROGRAM=DFHDELIM**, add the following entry:

```
DFHPLT TYPE=ENTRY,PROGRAM=DBUGPLTS
```

Do ***not*** put DBUGPPM2 in the PLTSD table.

- Modification of the PLTSD will be required if the Xpediter Script Facility or File Utility Audit Trail is enabled later in this installation.
- When Xpediter/CICS starts its DBPA transaction during PLT post-initialization, the user ID assigned is the same as the user ID assigned to the CPLT transaction.

Task 4.4.1 Using RDO to Update Resource Definitions

1. The previous method of using z/OS dynamic allocation and file open/close/write services to produce Compuware PARMLIB reports has been removed from the product and replaced with the use of transient data queues. Table 7 describes the queues, the report written to each queue, and the PARMLIB member prefix and parameter needed to override the queue name (if necessary). The definitions of these queues are in SMXDSAMP member CSDXDFIL. The only modifications you may need to make are to the TDQUEUE name and the ddname. The other definition values should not be changed. Changing the TDQUEUE name will require you to use the appropriate PARMLIB parameter so that the correct queue is used to produce the report during parameter processing. The ddname parameters must not conflict with existing ddnames in your region's JCL or other TDQUEUE definitions.

   While Compuware suggests defining all three queues, you can decide to define only one and use the appropriate PARMLIB parameters to point the reports to that single queue—or use INDirect queues to point to a single queue. The report queues cannot be defined as INTra or Remote.

2. Review the supplied JCL member DBCRDO. Refer to the table below and use the SYSIN in resource members CSDXDFIL, CSDXDPRG, and CSDXDTRN. If Xpediter's program autoinstall is implemented during customization, CSDXDPRG can be omitted.

   - If your site is not using Xpediter/Code Coverage, edit the CSDXDFIL, CSDXDTRN, and CSDXDPRG members and comment out the definitions for Xpediter/Code Coverage datasets, transactions, and programs (prefixed with XV).

   - If your site is using BAS, you may use the furnished JCL member CSD2BAS to create an input file for the BATCHREP view in the CICSPlex SM End User Interface.
   - LSRPOOLNUM or LSRPOOLID may be specified as appropriate. Be aware that for file control update requests, CICS does not support the use of LSRPOOLNUM(NONE) or LSRPOOLID(NONE) while Transaction Isolation is active. This restriction is enforced with an AFDK abend.

3. The Xpediter/CICS resource group created in this step must be added to a CICS group list, which in turn is specified in the system initialization parameter GRPLIST.
4. CSDXDFIL contains an ENQMODEL. Ensure that this ENQMODEL definition is added to all CICS regions that share the same Xpediter/CICS Profile Dataset.
5. Submit JCL member DBCRDO to batch the required CEDA transactions.

- If you changed any of the Xpediter transaction IDs before submitting the JCL, the corresponding configuration parameters must also be changed. For details, see the *Xpediter/CICS Advanced Configuration Guide*.
- The section entitled "Intercommunication Configuration" in the *Xpediter/CICS Advanced Configuration Guide* contains additional transaction definitions that may be necessary for your environment.

Table 8. Resource Members

SYSIN Member for JCL Member DBCRDO or CSD2BAS
CSDXDFIL

Table 8. Resource Members *(Continued)*

SYSIN Member for JCL Member DBCRDO or CSD2BAS
CSDXDPRG (omit if using autoinstall for Xpediter programs)
CSDXDTRN
CSDXDDB2 (DB2ENTRY and DB2TRAN entries for DB2 sites only)
CSDXDDYN (if using Xpediter's CICSPlex TOR support)

Task 4.5 Initialize the Xpediter Service Provider

Roles involved:
z/OS System Programmer

1. Edit sample SMXDSAMP member XDSSPROC to make sure the STEPLIB DD statement specifies the Xpediter/CICS APF authorized dataset (SMXDAUTH) used at your site.
   a. If you change the value of the SSID parameter value, make sure you change the global parameter SERVICE_PROVIDER_SUBSYSTEM_NAME to match. For details on other Xpediter Service Provider parameters, see the *Xpediter/CICS Advanced Configuration Guide*.
   b. Copy member XDSSPROC from SMXDSAMP member to a JCL procedure dataset defined to JES.
2. The Xpediter Service Provider Subsystem can be executed as a batch job or as a started task.
   – To run the subsystem as a batch job:
      1. If you change the value of the SSID parameter value in member XDSSJCL, make sure you change the global parameter SERVICE_PROVIDER_SUBSYSTEM_NAME to match.
      2. Submit XDSSJCL.
   – To run the subsystem as a started task, issue the following operator command:

S XDSSPROC{,SSID=xxxx}	where **{,SSID=xxxx}** is an optional parameter used to override the default subsystem identifier, XDSS, with the identifier used at your site.

If you would like the Xpediter Service Provider to be initialized automatically during MVS startup, add the following start command to SYS1.PARMLIB member COMMNDxx:

S XDSSPROC,SSID=xxxx	where **xxxx** is the subsystem identifier. XDSS is the default. Place the start command for the Xpediter Service Provider after your start of TCPIP.

The MVS Performance Group for the Xpediter Service Provider must be higher than CICS and less than VTAM to prevent runtime abends.

Task 4.6 Allocate a Profile File (DBCDEFPF)

Roles involved:
Xpediter/CICS Installer

Installing the profile file enables you to individually define PF key settings, footing options, scroll values, and trap and trace options.

Submit SMXDSAMP member DBCDEFPF to define the VSAM KSDS profile file.

The resource definition for DBUGPRF was installed as part of Update the CICS Resource Definitions on page 27.

Task 4.7 Set Up Test Assets

Roles involved:
Xpediter/CICS Installer

This task will guide you through setting up test assets that will be used for installation verification and demonstration purposes.

Task 4.7.1 Assemble/Compile and Link-Edit the Test Programs

The tables below show the program name for each test program found in the SMXDSAMP library and the language that it is used for. The tables also include the name of the Assembler subroutine that is called by each of these programs.

For specific information on compiling your programs using the Compuware language processor, refer to the *Enterprise Common Components Installation and Configuration Guide*.

Table 9. Test Programs

Program Name	Language
CWDEMCB2	COBOL
CWDEMCBN	COBOL (w/nested programming)
CWDEMASM	Assembler
CWCDSUBA	Assembler Subroutine
CWDEMPL	PL/I
CWDEMPE	Enterprise PL/I
CWDEMC	C

Table 10. Additional Test Programs Using Channels and Containers

Program Name	Language
CWDEMACH	Assembler
CWDEMCCH	COBOL
CWDEMPCH	Enterprise PL/I
CWDEMZCH	C

Complete the following steps to ensure that the test programs function properly:

1. Assemble program CWCDSUBA. If you have the Assembler option, process it using the Compuware Assembler language processor. The CWCDSUBA routine does not contain any CICS commands and, therefore, does not need to be translated. CWCDSUBA must be included when CWDEMCB2, CWDEMCCH, CWDEMPE, or CWDEMPCH are link-edited.

For the following steps, only the step for your site's primary language needs to be performed. You can perform the steps for any other languages your site has support for as well.

2. If you have the COBOL option, compile the CWDEMCB2 program with the modified compile/link-edit JCL that includes the Compuware COBOL language processor. Use the NODYNAM parameter in the COBOL compile step and make sure that the library for CWCDSUBA is in the SYSLIB dataset. (You may also compile CWDEMCCH.)
3. If you have the PL/I option, compile the CWDEMPE program with the modified compile/link-edit JCL that includes the Compuware PL/I language processor. Make sure that the library for CWCDSUBA is in the SYSLIB dataset. (You may also compile CWDEMPCH.)
4. If you have the Assembler option, assemble CWDEMASM using the modified assembly/link-edit JCL that includes the Compuware Assembler language processor. (You may also compile CWDEMACH.)
5. If you have the C option, compile the CWDEMC program with the modified compile/link-edit JCL that includes the Compuware C language processor. (You may also compile CWDEMZCH.) For detailed information regarding compiling and binding C programs, see the *Xpediter/CICS Advanced Configuration Guide*.

Task 4.7.2 Allocate and Initialize the Employee File (DBCDEFEM)

The employee dataset (keyed sequential) is required to run the test (demonstration) programs.

Define a VSAM KSDS file using the JCL in SMXDSAMP member DBCDEFEM. The JCL loads the file with five records. The test programs require that these records be present.

The resource definition for DBUGEMP was installed as part of Update the CICS Resource Definitions on page 27.

Task 4.8 Update the CICS Startup JCL

Roles involved:
Xpediter/CICS Installer

Change your site's CICS startup JCL as follows:

1. Make the CSS load library accessible to Xpediter/CICS by adding it to the DFHRPL concatenation in the CICS startup JCL. The CSS load library is named CPWR.LCXnnn.SLCXLOAD (nnn is the release of CSS) and is included in the installation of Enterprise Common Components.
2. Add the appropriate Xpediter/CICS load library listed in Table 11 to the DFHRPL concatenation:

Table 11. Load Libraries for CICS Release Support

CICS Release	Dataset Name
CICS TS 5.1	CPWR.QMXD170.SMXDO68L
CICS TS 5.2	CPWR.RMXD170.SMXDO69L
CICS TS 5.3	CPWR.SMXD170.SMXDO70L
CICS TS 5.4	CPWR.TMXD170.SMXDO71L

Table 11. Load Libraries for CICS Release Support ***(Continued)***

CICS Release	Dataset Name
CICS TS 5.5	CPWR.UMXD170.SMXDO72L

3. Define one or more SLS datasets to CICS using one or any combination of the following methods. (The maximum number of ddnames is specified by PARMLIB parameter MAXIMUM_SLS_FILES_TO_SEARCH. For more information, see the chapter entitled "Configuration Parameters" in the *Xpediter/CICS Advanced Configuration Guide*.):

   – Enable dynamic allocation by using PARMLIB parameter ENABLE_DYNAMIC_FILE_DEF_FOR_SLS_FILES and naming the SLS datasets as required for that feature. For more information, see the description of ENABLE_DYNAMIC_FILE_DEF_FOR_SLS_FILES in the chapter entitled "Configuration Parameters" in the *Xpediter/CICS Advanced Configuration Guide*.

   – Add a DD statement and/or RDO definition for each SLS standard DDIO dataset or shared directory to the CICS startup JCL.

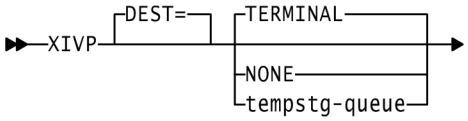
- If you use a non-IBM dynamic allocation product, ensure that the source listing datasets are excluded from automatic opens performed by these products when the dataset is allocated. If you fail to do so, you may be unable to access source listings through Xpediter/CICS.
- The optional datasets described in the remainder of this step have not yet been created.

4. If you are not dynamically allocating CICS datasets, add a DD statement for the profile file created in Allocate a Profile File (DBCDEFPF) on page 30.
5. If you are not dynamically allocating CICS datasets, add a DD statement for the employee file created in Allocate and Initialize the Employee File (DBCDEFEM) on page 31.

Task 4.9 Verify the Basic Xpediter/CICS Installation and Configuration

Roles involved:
Xpediter/CICS Installer

To verify that the Xpediter/CICS installation has been successful, Xpediter/CICS runs the Installation Verification Process (IVP). This procedure, performed automatically when the product is initialized in the CICS region, detects any changes introduced in the region that would cause the product not to function. You may request verification by using the XIVP transaction. This transaction routes the resulting output messages to the destination of your choice. The syntax for XIVP is as follows:



DEST=TERMINAL	Causes the output to go to the terminal at which XIVP is entered. This is the default.
DEST=NONE	Causes all output to go only to transient data destination CSMT.
DEST=tempstg-queue	Causes the output to go to the temporary storage queue name supplied.

The PARMLIB parameter SEVERITY_OF_IVP_MESSAGES_TO_CSMT_QUEUE controls the destination of messages. To change this parameter, refer to Milestone 7: Specifying Additional Configuration Parameters.

All discrepancies detected by XIVP are assigned a severity level of SEVERE, ERROR, WARNING, or INFORMATIONAL.

If severe errors are detected during initialization, they are placed in the temporary storage queue XPEDXIVP, and Xpediter/CICS abends with a code of DBI0. Refer to the chapter entitled "Abend Codes" in the *Xpediter/CICS Messages and Codes* manual for information on the abend code. If the product is being initialized at a terminal where errors are detected, the following messages are directed to the terminal user before the abend is issued:

```
ERRORS WERE DETECTED DURING INITIALIZATION
USE CEBR TO BROWSE TEMPSTG QUEUE XPEDXIVP, OR
EXAMINE TRANSDATA DESTINATION CSMT FOR MESSAGES, OR
EXECUTE TRANSACTION XIVP DEST=TERMINAL TO SEE ERRORS
```

If ERROR level discrepancies are detected during initialization, they are placed in the temporary storage queue XPEDXIVP, and Xpediter/CICS initialization will continue. If the product is being initialized at a terminal, the messages shown below will be directed to the terminal user. If the user responds **NO** to the `CONTINUE INITIALIZATION` prompt, Xpediter/CICS abends with DBI0.

```
              ERRORS DETECTED

THE PRODUCT MAY BE USED, BUT UNPREDICTABLE RESULTS
MAY OCCUR, UP TO AND INCLUDING POSSIBLE REGION
OUTAGES.  CORRECT THESE ERRORS AS SOON AS POSSIBLE.

CONTINUE INITIALIZATION? ==>         (YES/NO)
```

If product is initialized with ERROR level discrepancies, a message "Xpediter initialization error, execute XIVP" will be issued to the terminal when a new Xpediter/CICS session is started. This will alert each user to the fact that there may be problems with the product's configuration. These errors should be corrected as soon as possible.

Task 4.10 Review Automatic Session Termination

Roles involved:
CICS System Programmer

Xpediter/CICS can be tailored to terminate an active debugging session automatically if a terminal is disconnected, logged off, and/or signed off from CICS. Compuware highly recommends that you install automatic session termination to release the resources of sessions that have not been manually terminated.

The following tasks will be sufficient for most sites where terminal autoinstall is used. For other methods, refer to the chapter entitled "Configuring Automatic Session Termination" in the *Xpediter/CICS Advanced Configuration Guide*.

Task 4.10.1 Modify the Terminal Autoinstall Exit DFHZATDX

If your site uses a different autoinstall exit, its name can be found in the SIT parameter AIEXIT.

1. Add the following line of code to DFHZATDX in the "Delete Processing Section" after the IBM **Put Delete Code Here** comment:

```
EXEC CICS START INTERVAL (0) TRANSID ('XPNO') FROM (DELETE_TERM_ID)
```

• Add the above line in the section labeled **DELETE_TERMINAL**. Do ***not*** add the line to the section labeled **DELETE_SHIPPED_TERMINAL**.

• Consider adding "NOHANDLE"—or other appropriate condition handling—for exception conditions such as TRANSIDERR.

2. Reassemble and link edit DFHZATDX (or whatever VTAM terminal autoinstall exit is used at your site).

Milestone 5: Configure Xpediter/CICS — Upgrade

This chapter will guide you through configuration of an upgrade to Xpediter/CICS 17.02.

If you are performing a new installation instead, skip ahead to Milestone 6: Configure Topaz Workbench Integration.

Roles involved:
Xpediter/CICS Installer
z/OS System Programmer
CICS System Programmer
DB2 DBA
IMS DBA
Abend-AID for CICS Installer
z/OS Security Administrator.

Complete the following tasks to upgrade to Xpediter/CICS 17.02.

Task 5.1 Prepare the ISPF Edit Macro

Roles involved:
Xpediter/CICS Installer

XDUPDATE is an ISPF edit macro that can be used to automate the entry of site-specific JCL parameters. Using this macro saves you from having to repeatedly type in the same information and ensures your JCL parameters are always entered correctly.

If you have used the ISPF Edit Macro in the past, replace your existing edit macro rather than trying to update it. This will ensure that you are using an up-to-date edit macro.

To use the edit macro, first copy member XDUPDATE from the SMXDSAMP member (as described in the Note above) into your CLIST library. Then enter your site-specific information in the XDUPDATE macro. The macro includes instructions for entering the information (if you have used the ISPF Edit Macro in the past, it may be helpful to refer to the prior version). Your CLIST library must be allocated to your ISPF session. You can verify that your CLIST library is accessible to ISPF by entering the command TSO LISTA.

The edit macro can be used during Xpediter installation each time you are instructed to edit JCL. To run it, first open the JCL member for editing. You can then look at the JCL before any changes are made. Type XDUPDATE in the COMMAND field of the ISPF EDIT screen and press Enter.

XDUPDATE can also be used to update dataset names in CSDXDFIL.

Task 5.2 Authorize the Xpediter SMXDAUTH Library

Roles involved:
z/OS System Programmer

Ensure the Xpediter/CICS 17.02 SMXDAUTH library is APF authorized. The Xpediter/CICS APF authorized library is backward compatible. Compuware suggests you add the dataset to your IEAAPF*xx* or PROG*xx* list, as appropriate, so it will be APF authorized at each system IPL.

If Xpediter/CICS is deployed to additional LPARs, any copies of the SMXDAUTH library must also be APF authorized.

Task 5.3 Update the CICS Resource Definitions

Roles involved:
CICS System Programmer

Xpediter/CICS might not initialize using the CICS resource definitions from a previous release. Follow the steps below to update the CICS resource definitions.

1. PTFs XDQ021O–XDQ021U have replaced the z/OS services (dynamic allocation and file open/write/close) used to produce the Compuware PARMLIB Reports with the use of CICS Extrapartition Transient Data Queues and CICS services. As part of this refactoring, three new TDQ definitions have been included in the SMXDSAMP member CSDXDFIL. Table 7 on page 25 lists the default TDQ names, the Compuware PARMLIB Report associated with the TDQ, and the PARMLIB overrides used to change the TDQ names if required. For more information, see Task 4.1.4 Compuware PARMLIB Reports on page 24.
2. Review the supplied JCL member DBCRDO. Refer to the table below and use the SYSIN in resource members CSDXDFIL, CSDXDPRG, and CSDXDTRN. If Xpediter's program autoinstall is implemented during customization, CSDXDPRG can be omitted.

- If your site is not using Xpediter/Code Coverage, edit the CSDXDFIL, CSDXDTRN, and CSDXDPRG members and comment out the definitions for Xpediter/Code Coverage datasets, transactions, and programs (prefixed with XV).
- If your site is using BAS, you may use the furnished JCL member CSD2BAS to create an input file for the BATCHREP view in the CICSPlex SM End User Interface.
- LSRPOOLNUM or LSRPOOLID may be specified as appropriate. Be aware that for file control update requests, CICS does not support the use of LSRPOOLNUM(NONE) or LSRPOOLID(NONE) while Transaction Isolation is active. This restriction is enforced with an AFDK abend.

3. The Xpediter/CICS resource group created in this step must be added to a CICS group list, which in turn is specified in the system initialization parameter GRPLIST.
4. CSDXDFIL contains an ENQMODEL. Ensure that this ENQMODEL definition is added to all CICS regions that share the same Xpediter/CICS Profile Dataset.

5. Submit JCL member DBCRDO to batch the required CEDA transactions.

- If you changed any of the Xpediter transaction IDs before submitting the JCL, the corresponding configuration parameters must also be changed. For details, see the *Xpediter/CICS Advanced Configuration Guide*.
- The section entitled "Intercommunication Configuration" in the *Xpediter/CICS Advanced Configuration Guide* contains additional transaction definitions that may be necessary for your environment.

Table 12. Resource Members

SYSIN Member for JCL Member DBCRDO or CSD2BAS
CSDXDFIL
CSDXDPRG (omit if using autoinstall for Xpediter programs)
CSDXDTRN
CSDXDDB2 (DB2ENTRY and DB2TRAN entries for DB2 sites only)
CSDXDDYN (if using Xpediter's CICSPlex TOR support)

Task 5.4 Update and Initialize the Xpediter Service Provider

Roles involved:
z/OS System Programmer

For those sites using multiple releases of Xpediter/CICS, this release of the Service Provider is backward compatible with prior releases of Xpediter/CICS.

Update the STEPLIB DD statement in your XDSSPROC to the Xpediter/CICS 17.02 APF authorized dataset (SMXDAUTH) used at your site.

If you change the value of the SSID parameter value, make sure you change the PARMLIB parameter SERVICE_PROVIDER_SUBSYSTEM_NAME to match. For details on other Xpediter Service Provider parameters, see the *Xpediter/CICS Advanced Configuration Guide*.

You can update XDSSPROC, but before you can run it, Xpediter/CICS must be shut down in all CICS regions.

Task 5.5 Migrate Existing Parameters to Compuware PARMLIB

Roles involved:
ECC Installer
Xpediter/CICS Installer

Xpediter/CICS APAR LA00797 has implemented the advanced features of ECC 17.02. As part of this implementation, the keywords associated with all Xpediter/CICS and Xpediter/Code Coverage Global Parameters and DBPA Parameters have changed. Compuware strongly recommends that you upgrade to ECC 17.02 with all current maintenance before continuing with your migration. The following person is required to upgrade to ECC 17.02:

- ECC Installer.

After ECC 17.02 has been upgraded, continue with the applicable migration process below, depending on which of the two scenarios describes your site. The following person is required for this task:

- Xpediter/CICS Installer.

Upgrading Existing Compuware PARMLIB Members:

Proceed directly to Migrate Compuware PARMLIB Members to Compuware PARMLIB Version 2.

Upgrading From a Release Prior to 17.02.

If you have been using the XDGBLINP and/or XDDBPINP DDs to specify parameters, you will instead need to specify these parameters within the Compuware PARMLIB. Follow the steps in Migrate XDGBLINP Global Table Overrides through Migrate Compuware PARMLIB Members to Compuware PARMLIB Version 2 below.

Task 5.5.1 Migrate XDGBLINP Global Table Overrides

XDGBLINP may or may not vary across CICS regions:

- If your current XDGBLINP is common across all CICS regions, copy your current override member into Compuware PARMLIB member XDGB00. In all cases, the XDGB00 member will be the first member read for input.
- If your current XDGBLINP varies across different CICS regions, copy your current override members from the datasets in the XDGBLINP DD into the Compuware PARMLIB with member names of XDGB*xxxx* (where *xxxx* is 1 to 4 valid member name characters). Specify the XDGB*xxxx* members to be read for each CICS region in the Compuware PARMLIB index member XD$$00. The members will be read during product initialization or when the XSIT transaction is used to refresh parameter changes.

Task 5.5.2 Migrate XDDBPINP DBPA Input

XDDBPINP may or may not vary across CICS regions:

- If your current XDDBPINP is common across all CICS regions, copy your current override member into Compuware PARMLIB member XDDB00. In all cases, the XDDB00 member will be the first member read for input.
- If your current XDDBPINP varies across different CICS regions, copy your current override members from the datasets in the XDDBPINP DD into the Compuware PARMLIB with member names of XDDB*xxxx* (where *xxxx* is 1 to 4 valid member name characters). Specify the XDDB*xxxx* members to be read for each CICS region in the Compuware PARMLIB Index Member XD$$00. The members will be read during product initialization or when the XDBP transaction is used to refresh parameter changes. For more information on XD$$00, see the *Xpediter/CICS Advanced Configuration Guide*.

Task 5.5.3 Migrate Compuware PARMLIB Members to Compuware PARMLIB Version 2

ECC 17.02 with all current maintenance supplies a Migration Utility (member MIGDRIVR in ECC SLCXCNTL) that will allow you to migrate your existing Version 1 Compuware PARMLIB members to Version 2. Version 1 PARMLIB members were created when you first installed Xpediter/CICS release 17.02 (or as described in Migrate XDGBLINP Global Table Overrides and Migrate XDDBPINP DBPA Input above).

The ECC Migration Utility includes an Xpediter/CICS-specific module MXDMIGRT as part of this migration. Module MXDMIGRT will convert Version 1 keywords and values to the newly-implemented Version 2 keywords and values. You should run this utility for all your current Compuware PARMLIB members for Xpediter/CICS and for Xpediter/Code Coverage (if using

Xpediter/Code Coverage with Xpediter/CICS). These include members prefixed with XDGB, XDDB, and XVGB, as well as the index member XD$$00. Refer to the *Enterprise Common Components Installation and Configuration Guide*, "Migration Utility" for instructions on how to execute this utility.

The last line in PARMLIB Version 2 members prefixed with XDGB, XDDB, and XVGB, as well as index member XD$$00 is:

```
$$$ V2 END $$$ - DO NOT MODIFY OR REMOVE.
```

This line is used as the required delimiter for PARMLIB Version 2 processing and must ***not*** be removed.

Task 5.6 Update the CICS Startup JCL

Roles involved:
Xpediter/CICS Installer

Change your site's CICS startup JCL as follows:

1. Ensure that the most current CSS load library is in the DFHRPL concatenation. The CSS load library is named CPWR.LCX*nnn*.SLCXLOAD (*nnn* is the release of CSS) and is included in the installation of Enterprise Common Components.
2. Using Table 13, choose the appropriate Xpediter/CICS load library and update the DFHRPL concatenation:

Table 13. Load Libraries for CICS Release Support

CICS Release	Dataset Name
CICS TS 5.1	CPWR.QMXD170.SMXDO68L
CICS TS 5.2	CPWR.RMXD170.SMXDO69L
CICS TS 5.3	CPWR.SMXD170.SMXDO70L
CICS TS 5.4	CPWR.TMXD170.SMXDO71L
CICS TS 5.5	CPWR.UMXD170.SMXDO72L

3. The XDGBLINP and XDDBPINP DD statements will no longer be used and should be removed from your CICS Startup JCL.

Task 5.7 Reassemble the Global Parameters Table

Roles involved:
Xpediter/CICS Installer

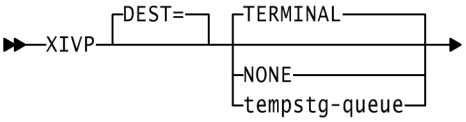
If you reassembled DBUGGBL, the global parameters table, during a previous Xpediter/CICS installation, you will need to review the parameters and make any required changes.

See Milestone 7: Specifying Additional Configuration Parameters for details about specifying parameters within the Compuware PARMLIB.

Task 5.8 Verify the Basic Xpediter/CICS Installation and Configuration

Roles involved:
Xpediter/CICS Installer

To verify that the Xpediter/CICS installation has been successful, Xpediter/CICS runs the Installation Verification Process (IVP). This procedure, performed automatically when the product is initialized in the CICS region, detects any changes introduced in the region that would cause the product not to function. You may request verification by using the XIVP transaction. This transaction routes the resulting output messages to the destination of your choice. The syntax for XIVP is as follows:



DEST=TERMINAL	Causes the output to go to the terminal at which XIVP is entered. This is the default.
DEST=NONE	Causes all output to go only to transient data destination CSMT.
DEST=tempstg-queue	Causes the output to go to the temporary storage queue name supplied.

The PARMLIB parameter SEVERITY_OF_IVP_MESSAGES_TO_CSMT_QUEUE controls the destination of messages. To change this parameter, refer to Milestone 7: Specifying Additional Configuration Parameters.

All discrepancies detected by XIVP are assigned a severity level of SEVERE, ERROR, WARNING, or INFORMATIONAL.

If severe errors are detected during initialization, they are placed in the temporary storage queue XPEDXIVP, and Xpediter/CICS abends with a code of DBI0. Refer to Chapter 1 in the *Xpediter/CICS Messages and Codes* manual for information on the abend code. If the product is being initialized at a terminal where errors are detected, the following messages are directed to the terminal user before the abend is issued:

```
ERRORS WERE DETECTED DURING INITIALIZATION
USE CEBR TO BROWSE TEMPSTG QUEUE XPEDXIVP, OR
EXAMINE TRANSDATA DESTINATION CSMT FOR MESSAGES, OR
EXECUTE TRANSACTION XIVP DEST=TERMINAL TO SEE ERRORS
```

If ERROR level discrepancies are detected during initialization, they are placed in the temporary storage queue XPEDXIVP, and Xpediter/CICS initialization will continue. If the product is being initialized at a terminal, the messages shown below will be directed to the terminal user. If the user responds **NO** to the `CONTINUE INITIALIZATION` prompt, Xpediter/CICS abends with DBI0.

```
              ERRORS DETECTED

THE PRODUCT MAY BE USED, BUT UNPREDICTABLE RESULTS
MAY OCCUR, UP TO AND INCLUDING POSSIBLE REGION
OUTAGES.  CORRECT THESE ERRORS AS SOON AS POSSIBLE.

CONTINUE INITIALIZATION? ==>         (YES/NO)
```

If product is initialized with ERROR level discrepancies, a message “Xpediter initialization error, execute XIVP” will be issued to the terminal when a new Xpediter/CICS session is started. This will alert each user to the fact that there may be problems with the product's configuration. These errors should be corrected as soon as possible.

Task 5.9 Update Support for the DB2 File Utility

Roles involved:
DB2 DBA

The DBRM members for the DB2 File Utility have been updated in release 17.02 of Xpediter/CICS. If you are using the DB2 File Utility, the bind for plan (MXDPLAN) needs to be rerun. The default bind package name has changed in this release to reflect the product code and release (MXD0170).

Perform the following steps to bind the plan:

1. Modify the JCL in member DBBIND10 as described in the comments within that member.
2. The JCL in DBBIND10 must be executed by an authorized DB2 user. Submit the JCL and ensure it completes successfully.
3. The plan bound above must be granted EXECUTION authority of PUBLIC by executing the following SQL statement using SPUFI (or whatever DB2 utility is used at your site):

   ```
   GRANT EXECUTE ON PLAN MXDPLAN TO PUBLIC
   ```

   If you changed the plan name in the JCL above, substitute that plan name in place of MXDPLAN before executing the SQL statement.

Task 5.10 Update Support for the IMS File Utility

Roles involved:
IMS DBA

If you have implemented the Xpediter/CICS IMS File Utility, you will need to update to the new Xpediter/CICS APF authorized library in the DLISAS and IMS control region STEPLIB. The default authorized library is SMXDAUTH.

Task 5.11 Update the File Utility Security Exit

Roles involved:
Xpediter/CICS Installer

If you have implemented the File Utility Security Exit, you will need to reassemble and relink the exit routine. For more information, see the “Customization” section in the chapter entitled “File Update Security Exit” in the *Xpediter/CICS Advanced Configuration Guide*.

Task 5.12 Update the Memory Update Security Exit

Roles involved:
Xpediter/CICS Installer

If you have implemented the Memory Update Security Exit, you will need to reassemble and relink the exit routine. For more information, see the chapter entitled “Memory Update Security Exit” in the *Xpediter/CICS Advanced Configuration Guide*.

Task 5.13 Update the Subsystem Security Exit

Roles involved:
Xpediter/CICS Installer

If you have implemented the Subsystem Security Exit, you can rename DBUGSSXX to DBUGSSX. An example SMP/E USERMOD to rename the module is in member SSXXUMOD in SMXDSAMP. If you implemented your own customized version, you will need to reassemble and relink the exit routine.

Task 5.14 INQUIRE ASSOCIATION Considerations

The Xpediter/CICS start-of-task exit uses the EXEC CICS INQUIRE ASSOCIATION call to obtain the client IP address associated with the task. This allows the terminal emulator IP address to be identified when a trap/breakpoint is taken. The ASSOCIATION command is subject to CICS command security checking using the resource identifier (ASSOCIATION). Xpediter/CICS will need a profile created that allows all CICS users to have READ access for resource identifier ASSOCIATION.

Milestone 6: Configure Topaz Workbench Integration

This chapter will guide you through the tasks necessary to integrate Xpediter/CICS with Topaz Workbench.

If your site

- Is not using—and does not plan to use—Topaz Workbench, ***or***
- Has already configured your CICS environment for Topaz Workbench integration during a previous installation or upgrade,

skip ahead to Milestone 7: Specifying Additional Configuration Parameters.

If you need to implement Topaz Workbench Integration within a CPSM CICSPlex environment, see the *Xpediter/CICS Advanced Configuration Guide*.

Roles involved:
CICS System Programmer
z/OS Security Administrator
Network Administrator
Topaz Workbench Installer.

Complete the following tasks to configure Topaz Workbench integration.

Task 6.1 Ensure IP CICS Sockets are Defined

Roles involved:
CICS System Programmer

Xpediter/CICS makes use of IP CICS Sockets for integrating with Topaz Workbench. If you do not currently have IP CICS Sockets defined in your CICS regions, follow the installations steps outlined in the *IP CICS Sockets Guide* supplied with IBM's z/OS Communication Server.

Topaz Workbench's integration with Xpediter/CICS requires TCP/IP v4.

Task 6.2 Verify IP CICS Sockets Listener Program and Transaction Definitions

Roles involved:
CICS System Programmer

Xpediter/CICS requires use of the default CICS sockets listener, EZACIC02, delivered with the z/OS Communications Server. If you have customized this program, refer to the *Xpediter/CICS Advanced Configuration Guide* for considerations.

XSKL is the default transaction code for the IP CICS sockets listener. If XSKL is not already defined, a sample is in SMXDSAMP member CSDXDTRN.

Transaction XSKL will run under the following security (If you choose another transaction code, make note of it for later use in Task 6.5 - Define Listener to IP CICS Sockets):

- If invoked from the PLTPI, it will run under the CICS PLT user ID (if the listener is started in the CICS PLT) or the ID of the user that invoked the EZAO transaction (if the listener is started using the EZAO transaction).
- If invoked from a terminal, it will run under the terminal user.
- Or in all cases, it can run under the default user ID specified in the USERID parameter in the IP CICS EZACICD definition.
- As previously mentioned in External Security Considerations on page 21, the user ID associated with transaction XSKL must be permitted to start a CICS transaction as a surrogate for the Topaz Workbench user (for example: read access to *.DFHSTART in CLASS SURROGAT).

If you wish to limit access to Xpediter/CICS by use of Topaz Workbench, you should instead secure transaction code XPGD as described in Task 6.3 - Define Transaction Code for Limiting Access to Xpediter/CICS Using Topaz Workbench below.

Task 6.3 Verify Transaction Definition for Controlling Access from Topaz Workbench

Roles involved:
CICS System Programmer
z/OS Security Administrator.

XPGD is the default transaction code for controlling access to Xpediter/CICS from Topaz Workbench. If XPGD is not already defined, a sample is in SMXDSAMP member CSDXDTRN.

- If your site has CICS SIT parameter XPCT=YES:
  - If you do not want to limit access, make sure you do ***not*** secure this transaction.
  - If your site wants to limit access to Xpediter/CICS debugging facilities through the use of Xpediter/Eclipse, you may do so by limiting access to transaction XPGD as follows:
    - Define any users you want to allow access by adding user profiles in the ACICSPCT class. Any users added should have READ access.
    - A QUERY SECURITY RESTYPE(TRANSACTION) will be performed for this transaction, and any user with READ access will be allowed to continue. If a user does not have READ access, they will not be allowed to continue, and message `MXDSK0026E Access Denied`,

uuuuuuuu Is Not Authorized To Use Xpediter/CICS API (where *uuuuuuuu* is the user ID) will be displayed in the message log.

- If your site has CICS SIT parameter XPCT=NO:
  - If you do not want to limit access, make sure transaction XPGD is ***not*** secured. This can be done by setting parameter RESSEC(NO) on the XPGD transaction definition (the default shipped with the product).
  - If your site wants to limit access to Xpediter/CICS debugging facilities through the use of Xpediter/Eclipse, you may do so by limiting access to transaction XPGD as follows:
    - Set parameter RESSEC(YES) on the XPGD transaction definition.
    - Define any users you want to allow access by adding user profiles in the ACICSPCT class. Any users added should have READ access.
    - A QUERY SECURITY RESTYPE(TRANSACTION) will be performed for this transaction, and any user with READ access will be allowed to continue. If a user does not have READ access, they will not be allowed to continue, and message MXDSK0026E Access Denied, *uuuuuuuu* Is Not Authorized To Use Xpediter/CICS API (where *uuuuuuuu* is the user ID) will be displayed in the message log.

Task 6.4 Define Xpediter/CICS Sockets Security Exit

Roles involved:
CICS System Programmer

Xpediter/CICS provides a sockets security exit, XDPCICSE, which allows users of Topaz Workbench to be authenticated. When establishing a debugging session from Topaz Workbench, a user must provide their mainframe user ID and password.

If you wish to ensure user ID/password authentication and logon, you must specify the parameter SECEXIT=XDPCICSE in the IP CICS Sockets Listener definition in the next section. If you do not require a user to log on to CICS, leave this parameter blank. This will cause the default CICS user ID to be assigned to the task under which the Topaz Workbench session runs.

If program XDPCICSE is not already defined, a sample is in SMXDSAMP member CSDXDPRG.

Task 6.5 Define Listener to IP CICS Sockets

Roles involved:
CICS System Programmer
Network Administrator.

This step provides the definition required to enable the Topaz Workbench components to connect and communicate properly with the Xpediter/CICS Sockets-Based Debugging API. Many of the parameters listed in Figure 1 below must be specified exactly as shown. Modifying them in any way can result in the unavailability of the Xpediter/CICS Sockets-Based Debugging API and complicate problem determination. If you are unsure of any parameter, code it as shown in the example in the figure below.

Execute the EZAC transaction and define the listener using the EZAC DEFine and ALTer commands.

If you are sharing a configuration dataset among several regions, make sure you create identical definitions for all regions you want Topaz Workbench users to be able to connect with. Each definition would require you to modify the APPLID and PORT parameters at a minimum. If you decided above to modify the CICS RDO definition for the XSKL or XSOC transactions, or if you need to create unique definitions by region, you must modify the TRANID and CSTRANid parameters as well. Highlighted parameter values listed in Figure 1 below must be coded as shown.

XSOC is the default transaction code for the Xpediter/CICS sockets interface. If XSOC is not already defined, a sample is in SMXDSAMP member CSDXDTRN.

Transaction XSOC will run under the following security:

- If the listener is started from the PLTPI, XSOC will run under the CICS PLT user ID. If the listener is started using the EZAO transaction, XSOC will run under the ID of the user that invoked the EZAO transaction.
- If the XSKL IP CICS socket definition (EZACICD) specifies a default user ID, that user ID will be used.

The USEREXIT parameter can be used if you want to ensure that the user attempting to use Xpediter/Eclipse can be authenticated. If you want to require authentication, the program XDPCICSE must be specified. If a user cannot be authenticated, access to Xpediter/CICS debugging facilities through Xpediter/Eclipse will be prohibited.

WLM (Work Load Manager) parameters have been explicitly omitted.

Figure 1. Definitions for Enabling Listener to IP CICS Sockets - Upper

```
5/19/2017 11:00:08 AM                          H01AC106              Page 1

 EZAC,DEFine,LISTENER (enhanced listener.  screen 1 of 2)     APPLID = ACMEC123

 Overtype to Enter

 APPLID       ===> ACMEC123            APPLID of CICS System
 TRANID       ===> XSKL                Transaction Name of Listener
 PORT         ===> nnnnn               Port Number of Listener
 AF           ===> INET                Listener Address Family
 IMMEDIATE    ===> YES                 Immediate Startup Yes|No
 BACKLOG      ===> 020                 Backlog Value for Listener
 NUMSOCK      ===> 050                 Number of Sockets in Listener
 ACCTIME      ===> 060                 Timeout Value for ACCEPT
 GIVTIME      ===> 060                 Timeout Value for GIVESOCKET
 REATIME      ===> 300                 Timeout Value for READ
 RTYTIME      ===> 015                 Stack Connection Retry Time
 LAPPLD       ===> NO                  Register Application Data
```

Figure 2. Definitions for Enabling Listener to IP CICS Sockets - Lower

```
5/19/2017 11:00:08 AM                          H01AC106              Page 2

 EZAC,DEFine,LISTENER (enhanced listener.  screen 2 of 2)     APPLID = ACMEC123

 Overtype to Enter

 CSTRANid      ===> XSOC              Child Server Transaction Name
 CSSTTYPe      ===> KC                Startup Method (KC|IC|TD)
 CSDELAY       ===> 000000            Delay Interval (hhmmss)
 MSGLENgth     ===> 061               Message Length (0-999)
 PEEKDATa      ===> NO                Enter Y|N
 MSGFORMat     ===> EBCDIC            Enter ASCII|EBCDIC
 USEREXIT      ===> XDPCICSE          Name of User/Security exit
 GETTID        ===> NO                Get AT-TLS ID  (YES|NO)
 USERID        ===> aaaaaaaa          Listener User ID
```

Task 6.6 Update the HCI Parameter File for CICS Debugging

Roles involved:
CICS System Programmer
Network Administrator.

Edit the TP Configuration section in the HCI Parameter File for CICS. The `CICS_*` statements identify the CICS regions you wish to participate in debugging sessions using Topaz Workbench. For more information on `CICS_*` statements within the HCI Parameter File, see the section "CICS_* Statements for CICS REGIONS FOR DEBUGGING" in the chapter entitled "CSS TP Configuration Section of the HCI Parameter File" in the *Enterprise Common Components Advanced Configuration Guide*.

Specifically:

- CSTRAN in EZAC must match CICS_SOCK in the HCI Parameter File
- PORT in EZAC must match CICS_PORT in the HCI Parameter File.

Task 6.7 Enable Xpediter/CICS Sockets-Based Debugging

Roles involved:
CICS System Programmer

To activate the changes you have made with the EZAC transaction and in the HCI Parameter File, perform the following:

- Activate the new listeners using the EZAO transaction. This is not necessary, however, if you have stopped and started IP CICS Sockets.
- Refresh the HCI parameter member you updated in Task 6.6 Update the HCI Parameter File for CICS Debugging.
- To implement the changes in the HCI, stop and restart the HCI.

Task 6.8 Validate the Topaz Workbench Integration

Roles involved:
Topaz Workbench Installer

Validation of the Topaz Workbench integration requires that Topaz Workbench be installed with the Xpediter/Eclipse feature. For details on installing Topaz Workbench and/or validating Xpediter/CICS integration with Topaz Workbench, see the *Topaz Workbench Installation Guide*.

Milestone 7: Specifying Additional Configuration Parameters

This chapter contains details regarding configuration of Xpediter/CICS to suit your site's needs.

If you are performing a new installation of Xpediter/CICS (as opposed to an upgrade), you may want to skip ahead to Milestone 9: Verify Product Installation before addressing this milestone. If you choose to do so, make sure to perform product verification after performing any additional configuration.

Roles involved:
Xpediter/CICS Installer

Configuration parameters are used to customize Xpediter/CICS. This milestone contains an overview about specifying parameter values, how they are processed, and reporting parameter processing results.

Complete the following tasks to specify additional configuration parameters.

Task 7.1 Specifying Configuration Parameters at Product Initialization

Xpediter/CICS includes a facility that allows you to specify most configuration parameter values using Compuware PARMLIB at product initialization. Most parameter values can be refreshed while Xpediter is active by entering the transaction XSIT.

Restrictions regarding particular parameters that are not eligible for processing from Compuware PARMLIB, and alternative means to specify them, are detailed in the *Xpediter/CICS Advanced Configuration Guide*.

Task 7.2 Enabling Restricted Operating Modes

In addition to its standard operating mode, Xpediter/CICS can be activated in three restricted modes of operation:

- Diagnosis Mode
- Utilities Mode
- Diagnosis/Utilities Mode.

These modes allow a site to tailor its Xpediter implementation to suit the processing integrity and response time requirements of its various CICS regions. In this way, a customer can eliminate unnecessary processing overhead while preventing any potentially disruptive user activity.

Details about configuring restricted operating modes can be found in the *Xpediter/CICS Advanced Configuration Guide*.

Task 7.3 Parameter Processing Overview

Parameters are read from the Compuware PARMLIB.

As the input is processed, a Compuware PARMLIB Report (Xpediter/CICS Parameters) is written to a transient data queue which must be defined as Extrapartition. The default queue name for this report is XGBR. The PARMLIB_REPORT_DDNAME configuration parameter can be used to change the name of the report queue.

Parameters are processed in the order in which they are read. Blank records and comment records are ignored. Parameters are written to the output report.

If an error is detected in a parameter, processing continues and an error message is written to the output report. Refer to the *Xpediter/CICS Messages and Codes* manual for a list of possible messages. See the *Xpediter/CICS Advanced Configuration Guide* for details on how to tailor the report.

Task 7.4 Xpediter/CICS Parameters Report

Figure 3 is a sample Xpediter/CICS Parameters report. The highlighted numbered references are discussed following the figure.

Figure 3. Sample Xpediter/CICS Parameters Report

```
                          Compuware PARMLIB Report                Page     1
                          Xpediter/CICS Parameters
 APPLID: H01AC024 Date: 08 NOV 2017 Time: 16:48:39 USERID: ACMJETO  TERM: 1718        [1]
                 Report TDQUEUE(XGBR) (Override from PARMLIB)                       [2]
                            Input Record/Error Message
       ------------------------------------------------------------------------     [3]
       *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -     [4]
       * Processing input from CMSC ID: CMSC                                       [5]
       *
       * CMSC common memory object refreshed from:
       *   CW01.CMSC.PARMLIB(XD$$00)                                              [6]
       *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
       *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -     [7]
       * Attempting to read data from the following parameter members:
       * XDGB00
       *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
       *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -     [8]
       * CMSC common memory object refreshed from:
       *   MC.DEV.PARMLIB2.PARMLIB(XDGB00)
       *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
ERROR> SERVICE_PROVIDER_SUBSYSTEM_NAME=XPLX                          < ERROR [9]
ERROR> Parameter may not be overridden while Xpediter/CICS is active  < ERROR.

          Parameters Successfully Processed.............  0
          Parameters Bypassed Due to Error..............  1
          Parameters Read from PARMLIB Memory Object...  1

ACTIVATE_CODE_COVERAGE_AT_INITIALIZATION=YES                                      [10]
ALLOW_FORCE_PARM_FOR_XPOF=YES
ALLOW_RESUME_FROM_ABEND=YES AUTO_SELECT_ABENDS_FROM_ANOTHER_TERMINAL=YES
AUTOMATIC_TRAP_ACTIVATION_CHECK_SECURITY=YES
AUTOMATIC_TRAP_ACTIVATION_EXCLUDE_PROGS=DFH*
AUTOMATIC_TRAP_ACTIVATION_EXCLUDE_PROGS=DSN*
AUTOMATIC_TRAP_ACTIVATION_EXCLUDE_PROGS=DB2*
AUTOMATIC_TRAP_ACTIVATION_EXCLUDE_TRANS=CEDA
AUTOMATIC_TRAP_ACTIVATION_EXCLUDE_TRANS=CEMT
AUTOMATIC_TRAP_ACTIVATION_EXCLUDE_TRANS=CESN
```

[1] The third heading line indicates the CICS region applid, date, time, terminal, and userID of the user logged on to the terminal when program DBUGSIT was run.

[2] These heading lines indicate the transient data queue used for the report. In addition, they indicate whether the queue used is the default name or has been overridden by the user.

[3] Input parameters are displayed from here down. This provides a complete audit of the parameters processed.

[4] Comments in the Compuware PARMLIB member are not displayed.

[5] The CMSC ID of the Compuware PARMLIB where the memory objects reside.

[6] The dataset where the memory object for XD$$00 was refreshed from.

[7] The concatenation order of the input members to be processed as derived from the index member XD$$00.

[8] A comment box precedes the data for each CMSC memory object read. It will display the dataset where the memory object was refreshed from.

[9] When an error is detected on an input record, the error is flagged. The line that follows contains a message indicating the error detected and is also flagged. Only the first error detected for a parameter is flagged.

[10] After all parameters have been processed, the report will show values for all available parameters with the specified/default values shown. The figure shows a small sample of parameters.

Task 7.5 Xpediter/CICS Parameters Log Output

Messages are written to the console log whenever the Xpediter/CICS Parameter process is run. In addition to messages indicating the start and end of processing, any errors related to dataset attributes are also written to the console. Figure 4 is a sample of output written to the console.

Messages that do not apply to a particular input parameter are displayed with two lines per message. All messages contain similar information on the first line including a message number, severity level, date, time, and region applid.

Figure 4. Sample Global Table Override Console Log Output

```
11.09.10 JOB00568  +MXDGB0000 I 30 SEP 2003 11:09:10 ACMEC123
                    XPEDITER/CICS GLOBAL TABLE OVERRIDE PROCESSING HAS BEGUN.
11.09.10 JOB00568  +MXDGB0001 I 31 30 SEP 2003 11:09:10 ACMEC123
                    XPEDITER/CICS GLOBAL TABLE OVERRIDE PROCESSING HAS ENDED.
```

In addition to the console log output shown in Figure 4, the input processing program for the parameter override process will display additional informational and error messages in the CICS CSMT output queue. Figure 5 provides a sample display of messages that can be output.

Figure 5. Sample Global Table Override CICS CSMT Output Queue Messages

```
MXDIP0000I 29 APR 2016 16:11:04 ACMEC123 Xpediter/CICS Input Provider DBUGINPT processing has begun.
MXDIP0026I 29 APR 2016 16:11:04 ACMEC123 Input provider DBUGINPT is processing input for DBUGSIT .
                                         Members are prefixed with "XDGB".
MXDIP0001I 29 APR 2016 16:11:05 ACMEC123 Xpediter/CICS Input Provider DBUGINPT processing has ended.
```

Milestone 8: Specifying System Facilities and Code Coverage Test Parameters

This milestone explains how to specify the parameters used to specify the Xpediter/CICS System Facilities and Code Coverage Test parameters (formerly DBPA Input).

Roles involved:
Xpediter/CICS Installer

Complete the following tasks to specify the parameters.

Task 8.1 Specifying System Facilities and Code Coverage Test Parameter Input

Xpediter/CICS uses Compuware PARMLIB members to define System Facilities and Code Coverage Test options during product initialization. The Compuware PARMLIB with the desired parameters is read at Xpediter/CICS startup.

All you need to do is create a member, XDDB00, in the Compuware PARMLIB that contains the System Facilities and Code Coverage Test parameters you wish to process. Comments and blank lines are allowed within your parameters. A report echoes the input parameters and provides any error messages. Refer to Implement the Compuware PARMLIB on page 19.

Transaction XDBP can be used to refresh the parameters from the Compuware PARMLIB. This eliminates the need to recycle Xpediter or the CICS region when you want to make storage protection option changes.

This task includes sections on the following topics:

- Processing of the parameters
- Overriding the report transient data queue name
- Format of the parameter processing report and console log output

If your site uses Xpediter/Code Coverage, this method can also be used to define Code Coverage tests. See the *Xpediter/Code Coverage Mainframe User/Reference Guide* for details.

Task 8.2 Processing Overview

System Facilities and Code Coverage Test parameters are read from the Compuware PARMLIB.

As the input is processed, a Compuware PARMLIB Report (Xpediter/CICS System Facilities and Code Coverage Test Parameters) is written to a transient data queue which must be defined as Extrapartition. The default queue name for this report is XDBR. The DBPA_REPORT_DDNAME configuration parameter can be used to change the name of the report queue.

Parameters are processed in the order in which they are read. Blank records and comment records are ignored. Parameter images are also written to the output report.

If an error is detected in a parameter, the record - along with an error message - is written to the report. Refer to the *Xpediter/CICS Messages and Codes* manual for a list of possible messages.

Task 8.3 Xpediter/CICS System Facilities and Code Coverage Test Parameters Report

The figure below is a sample Xpediter/CICS System Facilities and Code Coverage Test report. The highlighted numbered references are discussed following the figure.

Figure 6. Sample Xpediter/CICS System Facilities and Code Coverage Test Parameters Report

```
                          Compuware PARMLIB Report                Page     1
   Xpediter/CICS System Facilities and Code Coverage Test Parameters
 APPLID: ACMEC123 Date: 08 NOV 2017 Time: 16:48:39 USERID: ACMJET0  TERM: A123     [1]
                Report TDQUEUE(XDBR) (Override from PARMLIB)                     [2]
                             Input Record/Error Message
      ------------------------------------------------------------------------   [3]
      *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -   [4]
      * Processing input from CMSC ID: CMSC                                     [5]
      *
      * CMSC common memory object refreshed from:
      *   AC01.CMSC.PARMLIB(XD$$00)                                            [6]
      *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -   [7]
      * Attempting to read data from the following parameter members:
      * XDDB00
      *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -   [8]
      * CMSC common memory object refreshed from:
      *   SYS2.AC.CMSC.PARMLIB(XDDB00)
      *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DEFINE_SYSTEM_LABELS
        USER_LABEL=EIB
        EXISTING_LABEL=EIS
        OFFSET_HEX=8
        USE_CONTENT=YES
ERROR> END                                                                     [9]
ERROR> **** Entry with errors added - see screen 9.9                           [9]

         Parameters Successfully Processed.............  0
         Parameters Bypassed Due to Error..............  1
         Parameters Read from PARMLIB Memory Object...  1
```

[1] The third heading line indicates the CICS region applid, date, time, terminal, and userID of the user logged on to the terminal when program DBUGDBPA was run.

[2] These heading lines indicate the transient data queue used for the report. In addition, they indicate whether the queue used is the default name or has been overridden by the user.

[3] Input parameters are displayed from here down. This provides a complete audit of the parameters processed.

[4] Comments in the Compuware PARMLIB member are not displayed.

[5] The CMSC ID of the Compuware PARMLIB where the memory objects reside.

[6] The dataset where the memory object for XD$$00 was refreshed from.

[7] The concatenation order of the input members to be processed, as derived from the index member XD$$00.

[8] A comment box precedes the data for each CMSC memory object read. It will display the dataset and member name.

[9] When an error is detected on an input record, the error is flagged. The line that follows contains a message indicating the error detected and is also flagged. Only the first error detected for a parameter is flagged.

Xpediter/CICS System Facilities and Code Coverage Test Log Output

Task 8.4 Xpediter/CICS System Facilities and Code Coverage Test Parameters Log Output

Messages are written to the console log whenever the Xpediter/CICS System Facilities and Code Coverage Test Parameters process is run. Figure 7 is a sample of output written to the console.

Messages that do not apply to a particular System Facilities and Code Coverage Test parameter are displayed with two lines per message. All messages contain similar information on the first line including a message number, severity level, date, time, and region applid. This provides two places that can be checked for error conditions. Messages that apply to a particular System Facilities and Code Coverage Test parameter are displayed with three lines per message. The second line contains the first 71 bytes of the input image. The third line contains the message text.

Figure 7. Sample Xpediter/CICS System Facilities and Code Coverage Test Parameters Input, Console Log Output

```
11.09.24 JOB00568  +MXDPA0000 I 30 SEP 2003 11:09:24 ACMEC123
                    XPEDITER/CICS DBPA TRANSACTION PROCESSING HAS BEGUN.
11.09.24 JOB00568  +MXDPA0001 I 30 SEP 2003 11:09:24 ACMEC123
                    XPEDITER/CICS DBPA TRANSACTION PROCESSING HAS ENDED.
```

In addition to the console log output shown in Figure 7, the input processing program for the global override process will display additional informational and error messages in the CICS CSMT output queue. Figure 8 provides a sample display of messages that can be output.

Figure 8. Sample Xpediter/CICS System Facilities and Code Coverage Test Parameters Input, CICS CSMT Output Queue Messages

```
MXDIP0000I 29 APR 2016 16:11:12 H01AC024 Xpediter/CICS Input Provider DBUGINPT processing has begun.
MXDIP0026I 29 APR 2016 16:11:12 H01AC024 Input provider DBUGINPT is processing input for DBUGDBPA.
                                         Members are prefixed with "XDDB".
MXDIP0001I 29 APR 2016 16:11:13 H01AC024 Xpediter/CICS Input Provider DBUGINPT processing has ended.
```

Milestone 9: Verify Product Installation

This chapter contains tasks to verify that Xpediter/CICS is usable and your compile/assembly procedures were successfully modified.

The XPED level is used to activate Xpediter/CICS. Once activated, you can use one of the following test programs to access source and verify that Xpediter/CICS debugging functions are available:

- CWDEMASM for Assembler (transaction XASM)
- CWDEMC for C (transaction XCCC)
- CWDEMCB2 for COBOL (transaction XCB2)
- CWDEMPE for Enterprise PL/I (transaction XPLE).

Verification is only necessary using the main language you have support for. You can perform verification for any other language you have support for at your discretion.

Figures shown below are representative of COBOL program CWDEMCB2 being used for validation. Keep in mind that these figures will differ for other languages/programs being used for validation. Also, the line numbers shown here may vary from those you see during verification procedures.

Roles involved:
Xpediter/CICS Installer

Complete the following tasks to verify product execution.

Task 9.1 Abend Trapping and Stepping Without Breakpoints

1. On a blank CICS screen, type **XPED** and press Enter. This activates Xpediter/CICS, sets the abend trap option, and displays the Primary Menu as shown in Figure 9. Xpediter/CICS is now ready to intercept any abends that occur in any program executed from your terminal.

Figure 9. Primary Menu (XPED/XPRT)

```
-------------------- XPEDITER/CICS 17.02.00 - PRIMARY MENU ----------------C123
COMMAND ===>
MODULE:          CSECT:

   0  SESSION PROFILE      - Set default session attributes
   1  SESSION CONTROL      - Analyze summary of session events
   2  DEBUGGING FACILITIES - Interactively debug application programs
   5  FILE UTILITY         - Access datasets, temp stg, trans data, DLI, DB2
   7  ABEND-AID FOR CICS   - Interface to Abend-AID for CICS

   C  CODE COVERAGE        - Interface to Xpediter/Code Coverage
   G  XCHANGE/CICS         - Interface to Xpediter/Xchange CICS Facilities
   P  CICSPLEX FACILITIES  - Access CICSPlex Control Facilities
   X  EXIT                 - Exit Xpediter

      To set breakpoints in your program or keep specific data fields,
      enter your program name and use either the SOURCE command or PF key.

      For Online Technical Support refer to: https://go.compuware.com


      NOTICE:  Press PF2/PF14 to display the Copyright/Trade Secret Notice
```

2. Press Clear to return to CICS so that you can begin your test.
3. On a blank CICS screen, type the transaction associated with the language/program you wish to test and press Enter. In this example, type **XCB2** and press Enter to test COBOL program CWDEMCB2.

   The XCB2 Demonstration Transaction screen is displayed as shown in Figure 10.

Figure 10. XCB2 Demonstration Transaction Screen

```
XCB2 _____ - ENTER EMPLOYEE NUMBER                                      C123

  *** COMPUWARE CORPORATION ***
    DEMONSTRATION TRANSACTION

  ENTER DESIRED EMPLOYEE ABOVE:
  00001 - CAUSES ASRA ABEND
  00002 - CAUSES AEIM (AND OTHER ABENDS)
  00003 - CAUSES A WRITE TO TEMPORARY STORAGE
  00004 - STARTS UP XCB2 AS AN ASYNCHRONOUS TASK
  00005 - USED TO SHOW MULTIPLE CSECT SUPPORT
  00333 - CAUSES A STORAGE VIOLATION OF A SAA
  00999 - ENDS NORMALLY
```

4. To cause an ASRA abend, type **00001** and press Enter.

   The test program is intercepted before CICS gets control of the abend, and the Source Listing screen (2.L) is displayed as shown in Figure 11.

- If **DUMP=YES** is specified, an MVS snap dump is taken before Xpediter/CICS gets control.
- If the **NO SOURCE AVAILABLE** message or the **NO TIMESTAMP** message is displayed, use the HELP command (the default PF key is PF1) for detailed information. The Help screen lists the causes for the message and highlights the most probable causes. Also review the output from the Compuware language processor to verify that the source listing was successfully written to the source listing file.

Figure 11. COBOL Source Listing Screen Showing ASRA

```
---------------------- XPEDITER/CICS - SOURCE LISTING (2.L) ----------------C123
COMMAND ===>                                                  SCROLL ===> CSR
MODULE: CWDEMCB2 CSECT: CWDEMCB2          COMPILED: 25 JUN 2002 - 16.58.54
  LV ----- COBOL DATANAME KEEPS ---- -- ATTRIBUTES -- ----+---10----+---20--->
  77 CURR-PAY                        9(5)V99 NUM-DIS  0000000
  02 WA-HOURS                        999 NUM-DIS      $$$
  02 WA-RATE                         9(3)V99 NUM-DIS  00950
  **END**

------   ----------------------------- ASRA (DATA EXCEPTION) at CWDEMCB2.495 ->
000490                          MOVE '*** EMPLOYEE NOT ON FILE ****' TO PAYPROMP
000491                          GO TO 600-SEND-PAY-MAP.
000492
000493                      300-EMPLOYEE-PAY-RTN.
000494                          IF WA-TYPE EQUAL 'N' OR 'I' OR 'S'
=====>           1                   COMPUTE CURR-PAY   EQUAL WA-HOURS * WA-RATE
000496           1                   COMPUTE CURR-TAXES EQUAL CURR-PAY * WA-TAX-
000497           1                   ADD CURR-PAY   TO WA-YTD-GRS
000498           1                   ADD CURR-TAXES TO WA-YTD-TAX.
000499
000500                          IF PAYEMP1 EQUAL '00001'
000501           1                   MOVE WORK-AREA TO PAYROLL-DATA-EMP001.
000502
000503                          IF PAYEMP1 EQUAL '00999'
```

The module name, CSECT name, compile date, and compile time of program execution are displayed at the top of the screen to indicate which program is executing.

The keep window is displayed directly below the program information. All data names from the current statement are automatically displayed in this window. These are called *automatic keeps* and are valid for Assembler, COBOL, and PL/I programs.

Data names that were selected with the KEEP command are also displayed in this window. These are called *explicit keeps* and are indicated by a K next to the data name.

To scroll the information in the keep window, position the cursor in the window and use the PF keys assigned to scroll up, down, left, and right. The default PF keys are PF7, PF8, PF10, and PF11 respectively.

The STATUS line contains messages and is displayed immediately following the keep window. The example message shown in Figure 11 indicates that a data exception ASRA abend was intercepted at statement 495 in CWDEMCB2.

The source code follows the STATUS line. This section can be scrolled by positioning the cursor anywhere on the screen outside of the keep window. An arrow (=====>) indicates the current statement.

Notice the value of WA-HOURS in the keep window. The bad data in this field ($$$) is causing the ASRA.

5. To correct this data, position the cursor on $$$ in the keep window, type **040**, and press Enter.
6. Type **GO 1** on the COMMAND line (the default PF key is PF9) and press Enter to execute one line of source.

   Notice the changes to the values of the data fields in the keep window, as shown in Figure 12.

Figure 12. Source Listing Screen After Entering GO 1 Command

```
---------------------- XPEDITER/CICS - SOURCE LISTING (2.L) ----------------C123
COMMAND ===>                                                  SCROLL ===> CSR
MODULE: CWDEMCB2    ****** STATEMENT 000495 EXECUTED         STEP=00001 *******
  LV ----- COBOL DATANAME KEEPS ---- -- ATTRIBUTES -- ----+---10----+---20--->
  77 CURR-TAXES                     9(5)V99 NUM-DIS  0000000
* 77 CURR-PAY                       9(5)V99 NUM-DIS  0038000
  02 WA-TAX-RAT                     9(3)V9 NUM-DIS   0200
  **END**

------   ------------------------------------------------ Before CWDEMCB2.496 ->
000490                         MOVE '*** EMPLOYEE NOT ON FILE ****' TO PAYPROMP
000491                         GO TO 600-SEND-PAY-MAP.
000492
000493                     300-EMPLOYEE-PAY-RTN.
000494                         IF WA-TYPE EQUAL 'N' OR 'I' OR 'S'
000495           1                 COMPUTE CURR-PAY   EQUAL WA-HOURS * WA-RATE
=====>           1                 COMPUTE CURR-TAXES EQUAL CURR-PAY * WA-TAX-
000497           1                 ADD CURR-PAY   TO WA-YTD-GRS
000498           1                 ADD CURR-TAXES TO WA-YTD-TAX.
000499
000500                         IF PAYEMP1 EQUAL '00001'
000501           1                 MOVE WORK-AREA TO PAYROLL-DATA-EMP001.
000502
000503                         IF PAYEMP1 EQUAL '00999'
```

7. Type **GO** on the COMMAND line (the default PF key is PF12) and press Enter to continue the test.

   When execution successfully completes, the Transaction Complete screen is displayed as shown in Figure 13.

Figure 13. Transaction Complete Screen

```
*** COMPUWARE CORPORATION ***                                            C123
  DEMONSTRATION TRANSACTION

 EMPLOYEE NUMBER: 00001
 EMPLOYEE NAME:   MR. DAVID ABEND
 HOURS WORKED:    040
 HOURLY RATE:       9.50
 GROSS PAY:          380.00

*** TRANSACTION COMPLETE ***
```

   To release all resources, you need to tell Xpediter/CICS that you have completed your debugging session.

8. On a blank CICS screen, type **XPND** and press Enter.

Task 9.2 Additional Verification

While not required, you may want to perform further verification by exercising additional features of Xpediter/CICS (for example, setting keeps and breakpoints, stepping through a program, and so on). For details on additional verifications and how to perform them, see the appropriate section in the Xpediter/CICS *User Guide* for the specific language you want to test.

Milestone 10: Deployment

This chapter contains considerations to be aware of and tasks to perform during deployment of Xpediter/CICS.

Depending on your environment, you should review Milestone 4: Configure Xpediter/CICS — New Installation if you are deploying a new install or Milestone 5: Configure Xpediter/CICS — Upgrade if you are deploying an upgrade.

Roles involved:
Xpediter/CICS Installer

Complete the following task to deploy Xpediter/CICS.

Task 10.1 Target Library Deployment

If you installed more than one FMID, you need only copy a single instance of SMXDAUTH, SMXDAAFX, SMXDPDSE, and SMXDSAMP when deploying to multiple LPARs. Compuware recommends using the FMID that corresponds to the highest level of CICS.

You will also need to copy the SMXDO*xx*L target libraries for each release of CICS supported in the destination LPAR.

Task 10.2 Exclude Xpediter/CICS Transactions from Monitoring

If your site uses CICS performance monitors that limit the duration of CICS transactions, exclude Xpediter's background transactions from monitoring. Long-running functions may exceed the monitor's duration parameters. The transactions to be excluded are:

- XPFS
- XSKL
- XSOC
- XVKP.

The transaction names listed above are those supplied by Compuware. If alternative names were established during Xpediter/CICS configuration, make sure those alternative names are excluded.

Troubleshooting

This troubleshooting information can help you diagnose some common installation problems. For most situations, start troubleshooting by running transaction XIVP and reviewing the CSMT log for any diagnostic messages describing errors.

Typical Errors

DBI6 Abend Encountered

Indicates an error occurred trying to connect to the Xpediter Service Provider Subsystem. Specific details will appear in the prior XDCP*nnnn*S message.

Suggestions

- Ensure the XDSS subsystem has been started.
- Check that the subsystem is using an SMXDAUTH from the highest level of Xpediter/CICS you have installed on your system.

DBI0 Abend Encountered

Suggestion

Xpediter's Install Verification Program (IVP) detected a severe error. Review the CSMT log for the CICS region to review the specific error(s).

Other DBI*x* Abends Encountered

Suggestion

Refer to the *Xpediter/CICS Messages and Codes* manual for an explanation of the specific Xpediter initialization abend.

Error Message "MXDPP0016E *dd mmm yyyy hh:mm:ss* Xpediter/CICS cannot be started. Wrong Xpediter loadlib for release of CICS."

Suggestion

Ensure the target load library *c*MXD170.SMXDO*xx*L in your DFHRPL concatenation is the correct one for the version of CICS you are running.

Message "Breakpoint disallowed by global table" Encountered When Attempting to Set a Breakpoint.

Suggestion

This message means that the Xpediter global parameter ENABLE_SETTING_BREAKPOINTS is set to NO, ENABLE_UTILITIES_MODE is set to YES, or CAN_UPDATE_DATA_IN_DIAGNOSIS_MODE is not enabled for the current Xpediter transaction ID. Set these parameters accordingly: ENABLE_SETTING_BREAKPOINTS=YES, ENABLE_UTILITIES_MODE=NO, and

CAN_UPDATE_DATA_IN_DIAGNOSIS_MODE=(XP*xx*_USER), where XP*xx* corresponds to the transaction ID used to invoke Xpediter.

Message "SEVERE ERRORS DETECTED IN REQUIRED RESOURCES:" Indicating Program(s) CWPIDRVE, CXRELSMP, or CXTRSRVC

Suggestion

Verify that the Enterprise Common Components (ECC) load library (SLCXLOAD) is defined in your DFHRPL concatenation.

Message "XSP0002E Service Provider entered in non-authorized state" When Trying to Start the Xpediter/CICS Service Provider Subsystem

Suggestion

Ensure the SMXDAUTH library in you STEPLIB is APF authorized. Make sure the dataset is authorized after each IPL.

Abend ASRA/AKEA in Transaction XPED (or Your Selected Transaction ID as Defined in the Global Parameters)

Suggestions

- Ensure the target load library *c*MXD170.SMXDO*xx*L in your DFHRPL concatenation is the correct one for the version of CICS you are running.
- Check JESMSGLG for any additional associated messages that help indicate the cause.

S0C4 Abend Occurs in Program DBUGSP00

Suggestion

Ensure the target load library *c*MXD170.SMXDO*xx*L in your DFHRPL concatenation is the correct one for the version of CICS you are running.

Xpediter Starts But Breakpoints Are Not Working

Suggestion

Check for security messages regarding transactions DBXG or XREL. Ensure the user is allowed to run these background transactions.

Customer Solutions

Visit the Compuware Support Center, https://go.compuware.com, to find product documentation, knowledge articles, and other technical resources. You can open a case with the Customer Solutions team, order products, and much more.

Contact Customer Solutions by phone:

- USA and Canada: 1-800-538-7822 or 1-313-227-5444.
- All other countries: Contact your local Compuware office. Contact information is available at https://go.compuware.com.

Visit Compuware on the web at http://www.compuware.com for additional product information.

Checklist of Milestones and Tasks

- ❑ Milestone 1: Ensure Installation and Configuration of Companion Products
  - ❑ Task 1.1 Install/Upgrade Enterprise Common Components
  - ❑ Task 1.2 Apply ECC Maintenance
  - ❑ Task 1.3 Import Xpediter/CICS License
- ❑ Milestone 2: Install Xpediter/CICS Using SMP/E
  - ❑ Task 2.1 Follow the Compuware Installation Guide
- ❑ Milestone 3: Compuware PARMLIB Implementation
  - ❑ Task 3.1 Implement the Compuware PARMLIB
    - ❑ **Task 3.1.1 Create the Default Members**
    - ❑ **Task 3.1.2 Update the CMSC with your PARMLIB Members**
- ❑ Milestone 4: Configure Xpediter/CICS — New Installation
  - ❑ Task 4.1 External Security Considerations
    - ❑ **Task 4.1.1 Transaction Security**
    - ❑ **Task 4.1.2 Security Exits**
    - ❑ **Task 4.1.3 INQUIRE ASSOCIATION Considerations**
    - ❑ **Task 4.1.4 Compuware PARMLIB Reports**
    - ❑ **Task 4.1.5 Additional Considerations**
  - ❑ Task 4.2 Prepare the ISPF Edit Macro
  - ❑ Task 4.3 Integrate Xpediter/CICS with MVS
    - ❑ **Task 4.3.1 Authorize the Xpediter SMXDAUTH Library**
    - ❑ **Task 4.3.2 Define the Xpediter Service Provider Program Properties to MVS**
    - ❑ **Task 4.3.3 Define the Xpediter Service Provider as an MVS Subsystem**
  - ❑ Task 4.4 Update the CICS Resource Definitions
    - ❑ **Task 4.4.1 Using RDO to Update Resource Definitions**

- ❏ Task 6.1 Ensure IP CICS Sockets are Defined
- ❏ Task 6.2 Verify IP CICS Sockets Listener Program and Transaction Definitions
- ❏ Task 6.3 Verify Transaction Definition for Controlling Access from Topaz Workbench
- ❏ Task 6.4 Define Xpediter/CICS Sockets Security Exit
- ❏ Task 6.5 Define Listener to IP CICS Sockets
- ❏ Task 6.6 Update the HCI Parameter File for CICS Debugging
- ❏ Task 6.7 Enable Xpediter/CICS Sockets-Based Debugging
- ❏ Task 6.8 Validate the Topaz Workbench Integration

❏ Milestone 7: Specifying Additional Configuration Parameters

- ❏ Task 7.1 Specifying Configuration Parameters at Product Initialization
- ❏ Task 7.2 Enabling Restricted Operating Modes
- ❏ Task 7.3 Parameter Processing Overview
- ❏ Task 7.4 Xpediter/CICS Parameters Report
- ❏ Task 7.5 Xpediter/CICS Parameters Log Output

❏ Milestone 8: Specifying System Facilities and Code Coverage Test Parameters

- ❏ Task 8.1 Specifying System Facilities and Code Coverage Test Parameter Input
- ❏ Task 8.2 Processing Overview
- ❏ Task 8.3 Xpediter/CICS System Facilities and Code Coverage Test Parameters Report
- ❏ Task 8.4 Xpediter/CICS System Facilities and Code Coverage Test Parameters Log Output

❏ Milestone 9: Verify Product Installation

- ❏ Task 9.1 Abend Trapping and Stepping Without Breakpoints
- ❏ Task 9.2 Additional Verification

❏ Milestone 10: Deployment

- ❏ Task 10.1 Target Library Deployment
- ❏ Task 10.2 Exclude Xpediter/CICS Transactions from Monitoring