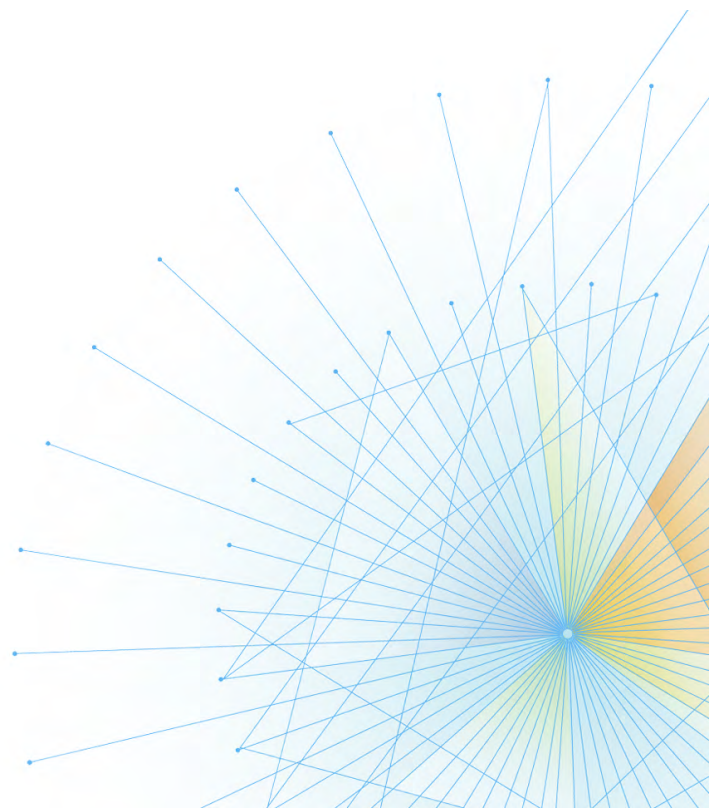




The Mainframe Software Partner For The Next 50 Years

# Xpediter/Code Coverage Installation and Configuration Guide

**Release 17.02**



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

**Compuware Support Center**

**<https://go.compuware.com/>**

This document and the product referenced in it are subject to the following legends:

Copyright 1999 - 2019 Compuware Corporation. All rights reserved. Unpublished rights reserved under the Copyright Laws of the United States.

U.S. GOVERNMENT RIGHTS-Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in Compuware Corporation license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable. Compuware Corporation.

This product contains confidential information and trade secrets of Compuware Corporation. Use, disclosure, or reproduction is prohibited without the prior express written permission of Compuware Corporation. Access is limited to authorized users. Use of this product is subject to the terms and conditions of the user's License Agreement with Compuware Corporation.

Xpediter, Code Coverage, File-AID, Abend-AID, Topaz Workbench, Enterprise Common Components, and Compuware Shared Services are trademarks or registered trademarks of Compuware Corporation.

IBM, CICS, DB2, DFSORT, IMS, Language Environment, MVS, OS/390, and z/OS are trademarks of International Business Machines Corporation.

ACF2, CA-MIM, CA-ROSCOE, ENDEVOR, LIBRARIAN, PANEXEC, PANVALET, and Top Secret are trademarks or registered trademarks of CA Technologies, Inc.

Adobe® Reader® is a trademark of Adobe Systems Incorporated in the United States and/or other countries.

All other company and product names are trademarks or registered trademarks of their respective owners.

# Contents

<b>Introduction</b> .....	<b>7</b>
Overview .....	7
Icons .....	7
<b>Xpediter/Code Coverage Overview</b> .....	<b>9</b>
Product Architecture .....	9
<b>Planning</b> .....	<b>11</b>
Steps Involved .....	11
Milestones and Roles .....	11
Pre-Installation Considerations .....	12
Results Repository Dataset .....	12
Topaz Workbench .....	12
Additional Configuration .....	12
Necessary Information and Who to Ask .....	13
Checklist of Milestones and Tasks .....	13
Prerequisites .....	13
Software and Hardware Requirements .....	13
Hardware Platforms .....	13
Operating Systems .....	13
Major Subsystems .....	13
Languages .....	14
ISV Software .....	14
Corequisites .....	14
<b>Milestone 1: Ensure Installation and Configuration of Companion Products</b> .....	<b>15</b>
Task 1.1 Install/Upgrade Xpediter/TSO and Xpediter/IMS .....	15
Task 1.2 Apply Xpediter/TSO and Xpediter/IMS Maintenance .....	15
Task 1.3 Install/Upgrade Xpediter/CICS .....	15
Task 1.4 Apply Xpediter/CICS Maintenance .....	15
Task 1.5 Import Xpediter/Code Coverage License .....	16
<b>Milestone 2: Install Xpediter/Code Coverage Using SMP/E</b> .....	<b>17</b>
SMP/E Installation .....	17
Task 2.1 Ensure Product Integrity .....	17
Task 2.2 Follow the Compuware Installation Guide .....	18
<b>Milestone 3: Configuration Preparation</b> .....	<b>19</b>
Task 3.1 Provide Users Access to Product Runtime Libraries and Files .....	19
Task 3.2 Allow Access for the Xpediter/Code Coverage Installer .....	19
<b>Milestone 4: Configure Xpediter/Code Coverage — New Installation</b> .....	<b>21</b>
Task 4.1 Install ISPF Support .....	21
Task 4.2 Create a Results Repository Dataset .....	23

Task 4.3 Global Filter File Implementation . . . . .	25
Task 4.4 Limit Record Deletion Capabilities . . . . .	26
<b>Milestone 5: Configure Xpediter/Code Coverage — Upgrade . . . . .</b>	<b>27</b>
Task 5.1 Update ISPF Support . . . . .	27
Task 5.1.1 ISPF Support . . . . .	27
Task 5.2 Upgrade Xpediter/Code Coverage with Xpediter/CICS. . . . .	28
Task 5.2.1 Modify JCL Template Dataset for CICS. . . . .	28
Task 5.2.2 Update the CICS Startup JCL . . . . .	28
Task 5.2.3 Convert Global Table Override Input . . . . .	28
<b>Milestone 6: Configure CICS for Use with Xpediter/Code Coverage . . . . .</b>	<b>29</b>
Preliminary CICS Considerations . . . . .	29
Specification Dataset Access . . . . .	30
XVCC/XVSC Transaction Security . . . . .	30
Dataset Security Requirements . . . . .	30
Overview of Using Code Coverage with Xpediter/CICS . . . . .	31
Task 6.1 Create and Format the Program Inventory for CICS . . . . .	33
Task 6.1.1 Determine Appropriate Size for the Program Inventory Dataset . . . . .	33
Task 6.1.2 Create a Program Inventory Dataset . . . . .	33
Task 6.2 Create an Extract Dataset for CICS . . . . .	36
Task 6.3 Create JCL Template Dataset for CICS . . . . .	37
Task 6.4 Create the XVTCSPEC Dataset for CICS . . . . .	39
Task 6.5 Update the CICS Resource Definitions . . . . .	40
Task 6.6 Update the CICS Startup JCL . . . . .	41
Task 6.7 Review Global Parameters . . . . .	41
Verifying the Code Coverage Installation . . . . .	41
<b>Milestone 7: Using Compuware PARMLIB to Configure Xpediter/Code Coverage with Xpediter/CICS . . . . .</b>	<b>43</b>
Specifying Xpediter/CICS Code Coverage Parameters at Product Initialization . . . . .	43
Parameter Ineligible for XVS1 Transaction . . . . .	44
Global Parameter Default Values and Descriptions . . . . .	44
Parameter Descriptions . . . . .	44
AUTO_SUBMIT_JCL_DURING_EXTRACTION . . . . .	44
CAPTURE_SYSTEM_FLOW_USING_GMT_TIME . . . . .	45
CODE_COVERAGE_PARMLIB_REPORT_ALLOCATION . . . . .	45
CODE_COVERAGE_PARMLIB_REPORT_DDNAME . . . . .	45
CODE_COVERAGE_PARMLIB_REPORT_SYSOUT_CLS . . . . .	45
DATA_EXTRACTION_INTERVAL . . . . .	45
INCREMENT_VERB_COUNTS . . . . .	45
KEEP_PROGRAM_COUNTS_FOR_USERID . . . . .	46
LOG_INFO_MESSAGES . . . . .	46
LOG_TDQ_NAME . . . . .	46
PRIME_OWNER_FILTER_WITH_CURRENT_USERID . . . . .	46
TRANSID_XVCC . . . . .	46
TRANSID_XVKP . . . . .	46

TRANSID_XVSC .....	46
TRANSID_XVSI .....	46
TRANSID_XVTQ .....	47
WRITE_DATA_EXTRACTION_MSGS_TO_JES_LOG .....	47
Overriding Parameters .....	47
Assembling and Link-Editing the XVTCGBL Program .....	47
Overriding Parameters Using the Compuware PARMLIB .....	47
Concatenation of Compuware PARMLIB Members .....	48
Processing Overview .....	48
Xpediter/CICS Code Coverage Parameter Report .....	48
Xpediter/CICS Code Coverage Parameter Log Output .....	50
Defining Output Report Parameters .....	51
Setting Up the Compuware PARMLIB Parameters .....	51
PARMLIB Product Configuration .....	52
Testing the Setup .....	52
Adding Parameters to a Member in the Compuware PARMLIB .....	53
<b>Milestone 8: Verify the Xpediter/Code Coverage Installation .....</b>	<b>55</b>
Task 8.1 Establish Batch COBOL Support .....	55
Test Preparation .....	55
Test Initiation and Execution .....	56
Task 8.2 Establish Online IMS Support .....	58
Test Initiation and Execution .....	58
Task 8.3 Establish CICS Support .....	61
Test Preparation .....	61
Populate Program Inventory .....	62
Test Definition and Execution .....	62
Task 8.4 Set Up Code Coverage Reporting .....	64
<b>Milestone 9: Configure Topaz Workbench Integration .....</b>	<b>67</b>
Task 9.1 Configure the CXSSAS Task .....	67
Task 9.2 Verify the Topaz Workbench Integration .....	67
<b>Milestone 10: Troubleshooting .....</b>	<b>69</b>
Customer Support .....	69
Information for Customer Solutions .....	69
<b>Checklist of Milestones and Tasks .....</b>	<b>71</b>



# Introduction

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

## Overview

This document is intended to guide you through installing/updating, configuring, deploying, and troubleshooting Xpediter/Code Coverage.

## Icons

The icons found in this guide include:



**“Tip”** A note or tip providing additional information.



**“Remember”** Information important to remember.



**“Fast Forward”** 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.



**“Roles”** The individuals required to perform a Milestone or Task.



**“Caution”** Failure to follow these instructions can cause problems.





# Xpediter/Code Coverage Overview

Xpediter/Code Coverage is an advanced application analysis tool that provides the ability to create reports detailing testing efficiency and risk metrics for mainframe COBOL, PL/I, and High-Level Assembler programs. Xpediter/Code Coverage works in conjunction with Compuware's mainframe test and debugging tools Xpediter/CICS and Xpediter/TSO and Xpediter/IMS.

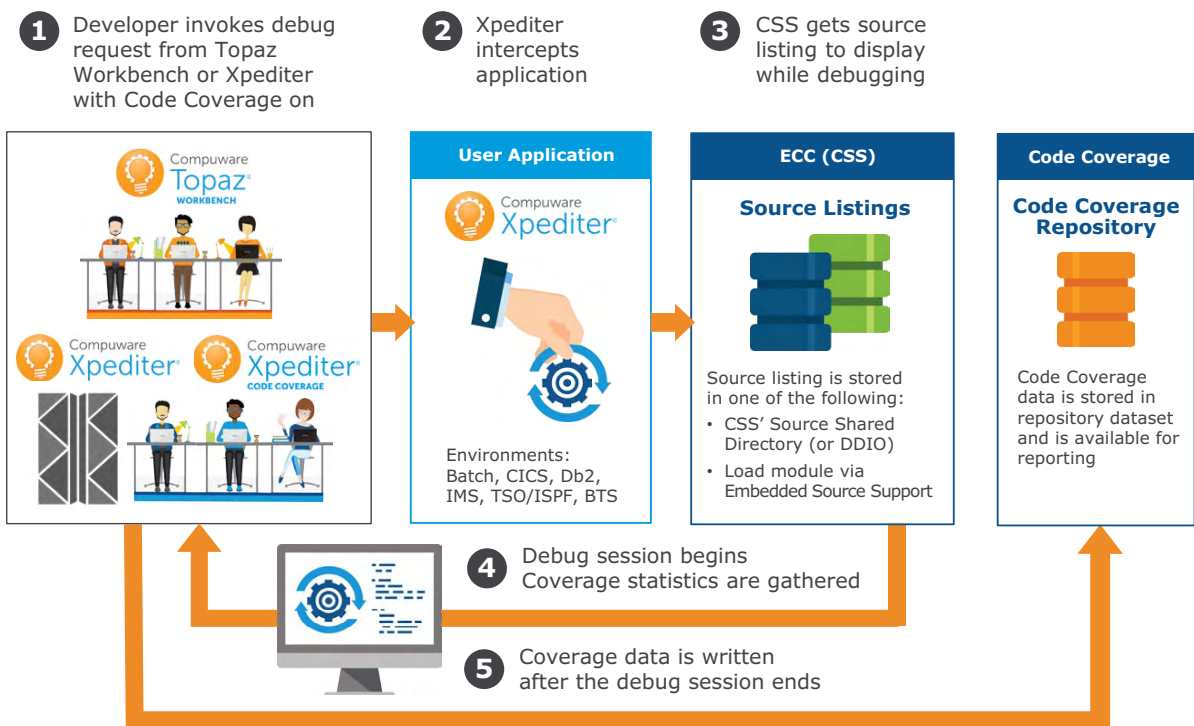
Members of Compuware's Xpediter family of debugging and testing products include:

- Xpediter/TSO and Xpediter/IMS
- Xpediter/CICS
- Xpediter/Code Coverage
- Xpediter/Xchange.

Each of these products has its own installation and configuration documentation.

## Product Architecture

Xpediter/Code Coverage is made up of a number of components accessible from either a standard terminal or from Topaz Workbench, an Eclipse-based modernized user interface.



The numbered steps in an Xpediter/Code Coverage debugging session workflow are:

- 1** A Topaz Workbench or mainframe Xpediter/CICS or Xpediter/TSO and Xpediter/IMS user initiates a debugging request for their user application with Xpediter/Code Coverage turned on.
- 2** Xpediter intercepts the user application for debugging. Supported Topaz Workbench environments include batch, TSO, DB2, and IMS. Supported Mainframe environments include batch, CICS, TSO/ISPF, DB2, IMS, and BTS.

- 3 Compuware Shared Services (CSS)—a component of Enterprise Common Components (ECC)—locates the source listing. It can be either stored in the CSS source shared directory (or DDIO) or embedded in the load module via Embedded Source Support.
- 4 The interactive debugging session begins. Code Coverage statistics are gathered. The source listing is displayed to give the user a familiar context for debugging.
- 5 When the session is completed, Xpediter/Code Coverage data is written to the Code Coverage repository.



Topaz Workbench integration requires that you follow the instructions in ["Milestone 9: Configure Topaz Workbench Integration"](#).

# Planning

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

## Steps Involved

1. Order Xpediter/Code Coverage and its companion products, including the latest maintenance, via Compuware's Product Ordering web page or by contacting Compuware as described in [Customer Support](#) on page 69.
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/Code Coverage has been imported.
  - b. Perform the SMP/E installation of Xpediter/Code Coverage according to the *Compuware Installer Mainframe Products SMP/E Installation Guide*.
  - c. Implement the Compuware PARMLIB.
  - d. Install the Xpediter/Code Coverage product.
  - e. Verify the installation of the Xpediter/Code Coverage product.
  - f. Perform additional configuration for Topaz Workbench.
  - g. Troubleshoot any problems with the installation.

## Milestones and Roles

Installation, configuration, verification, and deployment are done in 13 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/Administrator	Xpediter/Code Coverage Installer	z/OS System Programmer	z/OS Security Administrator	VTAM Administrator	DBA
" <a href="#">Milestone 1: Ensure Installation and Configuration of Companion Products</a> "	Xpediter/TSO ● Xpediter/IMS ● Xpediter/CICS ●	●	●	●	●	IMS ●
" <a href="#">Milestone 2: Install Xpediter/Code Coverage Using SMP/E</a> "		●				
" <a href="#">Milestone 3: Configuration Preparation</a> "		●	●	●		
" <a href="#">Milestone 4: Configure Xpediter/Code Coverage — New Installation</a> "		●	●			

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

Milestone	Companion Product Installer/Administrator	Xpediter/Code Coverage Installer	z/OS System Programmer	z/OS Security Administrator	VTAM Administrator	DBA
" <a href="#">Milestone 5: Configure Xpediter/Code Coverage — Upgrade</a> "	Xpediter/TSO ● Xpediter/CICS ●	●				
" <a href="#">Milestone 6: Configure CICS for Use with Xpediter/Code Coverage</a> "		●	●			
" <a href="#">Milestone 7: Using Compuware PARMLIB to Configure Xpediter/Code Coverage with Xpediter/CICS</a> "		●				
" <a href="#">Milestone 8: Verify the Xpediter/Code Coverage Installation</a> "		●				
" <a href="#">Milestone 9: Configure Topaz Workbench Integration</a> "	ECC ● Topaz Workbench ●	●				

## Pre-Installation Considerations

Installing Xpediter/Code Coverage consists of registering defaults and providing an access mechanism for users. To facilitate a smooth installation, collect the information identified in this section. You can print the table in this section and use it to write down your site's values.

### Results Repository Dataset

- If a Results Repository dataset from a prior Xpediter/Code Coverage installation exists, you can use it. Make note of the Dataset Name in your copy of [Table 2](#) for future reference.
- If an Results Repository dataset does not exist, or you decide to create a new one, make note of the Dataset Name in your copy of [Table 2](#) for future reference.

### Topaz Workbench

The Topaz Workbench features Code Coverage/Eclipse and Topaz for Total Test utilize Xpediter/Code Coverage. You will modify the SSAS PROC in Enterprise Common Components (ECC) to activate Topaz Workbench support.

### Additional Configuration

Some of the tasks in this *Guide* are only necessary when configuring additional licensed features, or in specialized environments, such as:

- Topaz Workbench
- Xpediter/TSO and Xpediter/IMS
- Xpediter/CICS.

You need to know which options your site is licensed for, as well as what other Compuware products your site is licensed for. You can determine this using the Compuware License Management utility. Your ECC administrator should be able to direct you to this utility.

Information that should be determined before you start to configure Xpediter/Code Coverage is listed in [Table 2](#).

## Necessary Information and Who to Ask

[Table 2](#) lists the information needed and who would be expected to provide it. You may want to print the table, then fill in your site's information to refer to during the installation process.

**Table 2** Information Gathering Worksheet

Who to Contact	What is Needed	Your Information
<b>Required</b>		
Xpediter/Code Coverage SMP/E Installer	High-level qualifiers for Xpediter/Code Coverage SMP/E datasets	
ECC Administrator	Compuware PARMLIB dataset name	
ECC Administrator	Suffix defined/defaulted for Xpediter/Code Coverage in the Compuware PARMLIB	
If re-using Results Repository Dataset, previous Xpediter/Code Coverage Installer. If new Results Repository Dataset, current Xpediter/Code Coverage Installer.	Xpediter/Code Coverage Results Repository Dataset name	
<b>Required for Topaz Workbench</b>		
ECC Administrator	<ul style="list-style-type: none"> <li>• Activate Code Coverage for Topaz Workbench</li> <li>• SSAS PROC modified adding the SLXVLOAD dataset</li> </ul>	

## Checklist of Milestones and Tasks

To keep track of your progress, you may want to print the [“Checklist of Milestones and Tasks”](#) at the end of this manual, then check off each Milestone and task as it gets completed.

## Prerequisites

### Software and Hardware Requirements

#### 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
- IBM DFSORT

#### Major Subsystems

- IBM DB2 for z/OS V11.1, 12.1

- IBM IMS Batch Terminal Simulator for z/OS V4.1
- IBM IMS Transaction and Database Servers V14.1, 15.1

## Languages

### (Under IBM Language Environment for Supported z/OS Releases)

- IBM Enterprise COBOL for z/OS V4.2, 5.1, 5.1.1, 5.2, 6.1, 6.2 (all in compatibility mode)
- IBM Enterprise PL/I for z/OS V4.5, 5.1, 5.2 (all in compatibility mode)
- IBM High Level Assembler for z/OS, z/VM, and z/VSE V1.6
- IBM C for supported z/OS releases. Note: Support is limited to C, not C++.

## ISV Software

- CA Datacom/DB 12, 14
- CA IDMS 18.5
- Hogan
- SAG Adabas 8.3.1, 8.3.2
- SUPRA
- Syncsort
- System 2000
- TIS
- TOTAL

## Corequisites

- Compuware Products and Components:
  - Enterprise Common Components (ECC) 17.02 or newer.

Contact Compuware Customer Support (as described in [Customer Support](#) on page 69) for information on releases not mentioned above.

# Milestone 1: Ensure Installation and Configuration of Companion Products

Xpediter/Code Coverage utilizes information gathered by Xpediter/CICS and/or by Xpediter/TSO and Xpediter/IMS.



Roles involved:  
Xpediter/TSO and Xpediter/IMS Installer  
Xpediter/CICS Installer.

Complete the following tasks to install and configure Xpediter/Code Coverage companion product Xpediter/TSO and Xpediter/IMS.

## Task 1.1 Install/Upgrade Xpediter/TSO and Xpediter/IMS



If your site does not use Xpediter/TSO and Xpediter/IMS, skip ahead to [Install/Upgrade Xpediter/CICS](#).

Xpediter/TSO and Xpediter/IMS version 17.02 or higher must be installed and configured to support Xpediter/Code Coverage 17.02. See the *Xpediter/TSO and Xpediter/IMS Installation and Configuration Guide* for instructions on configuring Xpediter/TSO and Xpediter/IMS for use with Xpediter/Code Coverage.

## Task 1.2 Apply Xpediter/TSO and Xpediter/IMS Maintenance

Apply the latest maintenance to Xpediter/TSO and Xpediter/IMS 17.02.

## Task 1.3 Install/Upgrade Xpediter/CICS



If your site does not use Xpediter/CICS, skip ahead to [Import Xpediter/Code Coverage License](#) on page 16.

Xpediter/CICS version 17.02 or higher must be installed and configured to support Xpediter/Code Coverage 17.02. See the *Xpediter/CICS Installation and Configuration Guide* for instructions on configuring Xpediter/CICS for use with Xpediter/Code Coverage.

## Task 1.4 Apply Xpediter/CICS Maintenance

Apply the latest maintenance to Xpediter/CICS 17.02.

## **Task 1.5 Import Xpediter/Code Coverage License**

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



# Milestone 2: Install Xpediter/Code Coverage Using SMP/E

## SMP/E Installation

Xpediter/Code Coverage is installed using SMP/E. This milestone will guide you through the SMP/E installation of Xpediter/Code Coverage.



Roles involved:  
z/OS Security Administrator  
Xpediter/Code Coverage Installer.

Complete the following tasks to SMP/E-install Xpediter/Code Coverage.

### Task 2.1 Ensure Product Integrity

The following person is required for this task:



z/OS Security Administrator

Your site may already have a set of standards for SMP/E target and distribution libraries.

1. Provide the installation program with FULL access to the following libraries ([Table 3](#)).
2. Provide the individual responsible for product maintenance with UPDATE access.

**Table 3** Access to SMP/E Libraries

Library	Identified By
Distribution	Low-level qualifier prefix ALXVxxxx
Target	Low-level qualifier prefix SLXVxxxx
Run-time	Run-time libraries and target libraries may be the same.



**Do *not* allow the installer to modify the distributed files.**

## Task 2.2 Follow the Compuware Installation Guide

The following person is required for this task:



Xpediter/Code Coverage Installer

1. Follow the instructions in the *Compuware Installer Mainframe Products SMP/E Installation Guide* to install Xpediter/Code Coverage.
2. Once completed, follow the steps in this guide to configure and deploy Xpediter/Code Coverage.

## Milestone 3: Configuration Preparation

This milestone contains tasks that help establish security procedures and prepare for configuration of Xpediter/Code Coverage for both a new install and an upgrade.



Roles involved:  
z/OS Security Administrator.

Complete the following tasks to establish security procedures and prepare for Xpediter/Code Coverage configuration.

### Task 3.1 Provide Users Access to Product Runtime Libraries and Files

Enable user access to the libraries listed in [Table 4](#).

**Table 4** Xpediter/Code Coverage Library Security Requirements

Library	Type	Access Required by Users
INSTALL	Installation library	NO ACCESS
SLXVLOAD	Load library	READ/EXECUTE
SLXVCLIB	CLIST library	READ
SLXVMENU	Messages library	READ
SLXVPENU	Panel library	READ
SLXVSENU	Skeleton library	READ
SLXVTABL	Tables library	READ
SLXVMJPN <sup>1</sup>	Messages library	READ (if installing double-byte character support)
SLXVPJPN <sup>1</sup>	Panels library	READ (if installing double-byte character support)



<sup>1</sup>If you are installing Japanese language support, concatenate SLXVMJPN ahead of SLXVMENU in the ISPMLIB DD and concatenate SLXVPJPN ahead of SLXVPENU in the ISPPLIB DD.

### Task 3.2 Allow Access for the Xpediter/Code Coverage Installer

- The Xpediter/Code Coverage installer requires UPDATE access to the following:
  - Your Compuware PARMLIB
  - Xpediter/Code Coverage installation library SLXVINST.
- The Xpediter/Code Coverage installer also requires READ access to the following:
  - For an upgrade, run-time libraries from prior release of Xpediter/Code Coverage
  - Xpediter/Code Coverage SMP/E controlled libraries
  - Output of the executing Compuware Mainframe Services Controller (CMSC) started task.



# Milestone 4: Configure Xpediter/Code Coverage — New Installation

This milestone will guide you through the configuration of a new installation of Xpediter/Code Coverage 17.02.



If you are performing an upgrade instead, skip ahead to ["Milestone 5: Configure Xpediter/Code Coverage — Upgrade"](#).



Roles involved:  
Xpediter/Code Coverage Installer  
z/OS System Programmer.

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

## Task 4.1 Install ISPF Support

Testing with Xpediter/Code Coverage interactively might require an increase in the default TSO logon region size. Xpediter/Code Coverage requires about 250K.

1. Create an ISPF logon PROC or CLIST.
2. Establish a way of allocating the Xpediter/Code Coverage datasets (such as .SLXVPENU or .SLXVSENU) to each user of Code Coverage.

The following are two suggested methods:

- **Recommended:** Use the LIBDEF command. Copy SLXVSAMP members XPCOVER and XVTTCGDN into a library in your SYSPROC concatenation.

This CLIST library *must* be allocated to your ISPF session.



You can display the libraries in your SYSPROC concatenation by entering the command  
TSO ISRDDN.

By default, the XPCOVER CLIST will use the Simple Deploy method to allocate the dataset names specified in the CSMC PARMLIB DDSN\* member (DDSN00 by default). If you are not using the default member, make sure to point to the DDSN\* member name by setting the DDSNSFX keyword to the one- to four-character PARMLIB member suffix.

For example, to point to PARMLIB member DDSNCPWR, use:

```
DDSNSFX(CPWR)
```

In addition, the primary CMSC will be used by default. If you are not using the default CMSC, specify the CMSC to use by setting the CMSC keyword to the one- to four-character CMSC ID.

For example, to point to CMSC subsystem TEST, use:

```
CMSC(TEST)
```

The XPCOVER CLIST can alternately use High-Level Qualifiers (HLQ) to allocate the Xpediter/Code Coverage datasets it references. See the comments in the CLIST for more information.

- **Alternative:** Modify or create TSO logon PROCs to concatenate Xpediter/Code Coverage libraries to the appropriate DD cards. The Xpediter/Code Coverage datasets listed in [Table 5](#) must be concatenated into the listed ddnames. Replace CPWR.MLXV170 with the values used for the installation of Xpediter/Code Coverage.

**Table 5** ISPF Support Dataset Concatenation

ddname	Dataset Name
SYSPROC	CPWR.MLXV170.SLXVCLIB
ISPLLIB	CPWR.MLXV170.SLXVLOAD
ISPMLIB	CPWR.MLXV170.SLXVMENU
ISPLLIB	CPWR.MLXV170.SLXVPENU
ISPSLIB	CPWR.MLXV170.SLXVSENU
ISPTLIB	CPWR.MLXV170.SLXVTABL

3. If Xpediter/CICS has been installed, add the following datasets for the Program Inventory Utility:

**Table 6** Xpediter/CICS Datasets for use with the Program Inventory Utility

ddname	Dataset Name
ISPLLIB	CPWR.cMXDnnn.SMXDOxxL

4. Replace CPWR.MLXV170 with the values used for the installation of Xpediter/Code Coverage. Replace CPWR.cMXDnnn.SMXDOxxL, with the appropriate load library.

Refer to the table “Load Libraries for CICS Release Support” in the *Xpediter/CICS Installation and Configuration Guide* for the default dataset names.

5. If your site uses any security processing that restricts the programs that can be used as TSO commands, make sure the following programs are available to your users and are not affected by any such restrictions:
  - XVTTACMT
  - XVTTADSN
  - XVTTAENQ
  - XVTTAFGN
  - XVTTAFIL
  - XVTTAFLG
  - XVTTAJCL
  - XVTTAMNT
  - XVTTARGN
  - XVTTASDL
  - XVTTAU01.
6. Provide a way to invoke the Xpediter/Code Coverage product.

- If you pre-allocated all the required Xpediter/Code Coverage libraries, the basic ISPF (internal) subcommand needed to invoke the Code Coverage product is:

```
'SELECT CMD(XVTTCGO) NEWAPPL(XVTC) PASSLIB'
```

- If you tailored the XPCOVER CLIST and put it in the SYSPROC concatenation or activated it with an ALTLIB command, enter:

```
SELECT CMD(%XPCOVER)
```

- If you tailored the XPCOVER CLIST into some other library, enter:

```
SELECT CMD(EXEC 'YOUR.LIBRARY.CLIST(XPCOVER)')
```

You also have options for specifying the SELECT command:

- You can specify it as a part of a command table entry for a verb such as CODECOVR.
- You can establish the command as an option by editing an ISPF panel. If you establish the command as a panel option, the following is recommended for the ISPF menu text:

```
XV Xpediter/Code Coverage - Code Analysis and Program Report Generation Tool
```

The following would be added to the processing section TRANS statement:

```
XV, 'xxxx'
```

where *xxxx* is the parameter of the appropriate SELECT command described above (omitting the SELECT verb). For example, if you tailored the XPCOVER CLIST and put it in the SYSPROC concatenation as shown above, the following statement would be added to the processing section TRANS statement:

```
XV CMD(%XPCOVER)
```

7. If Xpediter/CICS has been installed, Compuware recommends completing the following steps to ensure the availability of the Program Inventory Utility:
  - a. After creating or modifying the logon PROC or CLIST for Code Coverage, enter Code Coverage and choose option 5 (ISPF option 0.5) from the Defaults menu.
  - b. Enter the name of the default program inventory file for all users.
  - c. Enter the name of the excludes file distributed with Xpediter/CICS:

```
'CPWR.cMXDnnn.SMXDSAMP(POPEXCL)'
```

where *nnn* is the release of Xpediter/CICS, and *c* designates the CICS release.

- d. Choose option 6 (Code Coverage option 0.6) from the Defaults menu and enter the names of all the default Xpediter/CICS application load libraries.
  - e. Choose option 7 (Code Coverage option 0.7) from the Defaults menu and enter the names of all the default Xpediter/CICS source listing files.
  - f. Access the ISPF Primary Option Menu, and select option 3.3 (Move, or copy members or datasets). Copy your profile (ISPFPROF) dataset member XVTCPROF to CPWR.MLXV170.SLXVTABL, the new Code Coverage dataset.
8. If Xpediter/TSO is installed and the JCL convert option is necessary, the CPWR.MLXV170.SLXVSENU dataset member XVTTTSJ02 must be tailored with the names of your ISPF datasets and the Xpediter/TSO XOPTIONS dataset name. Change the names to reflect your datasets. These datasets are marked with **\*\*\*TAILOR\*\*\*\*** in the member.

## Task 4.2 Create a Results Repository Dataset



You may postpone this step until the remaining installation steps have been completed, but it must be performed before Code Coverage can be used.

Results Repository datasets can be used to save or accumulate data created through the use of the Xpediter/Code Coverage product. Code Coverage utilities will allow you to maintain your Results

Repository datasets. You can create and delete the Repository datasets. You can list or delete Repository members by system, load module, or individual program entry. You can also export and import Repository members, or merge member records from one Results Repository into another.

The screen shown in [Figure 1](#) can be used to build a batch job that will create or delete an Xpediter/Code Coverage Results Repository dataset. The batch job will perform the actual allocation or deletion of the dataset.

When you select option 1 from the Code Coverage UTILITIES screen and press Enter, the screen shown in [Figure 1](#) will be displayed:

**Figure 1** Code Coverage Repository Utility Screen

```

----- CODE COVERAGE 17.02 - REPOSITORY UTILITY -----
COMMAND ==>

Commands: C (Create file)    D (Delete file)

Repository File ==> 'ACMJETO.REPOSIT'

Management Class ==>          (optional)
Storage Class ==>            (optional)
Volume ==>                   (optional)
Unit ==>                     (optional)

Space Units ==> CYL          (CYL/MB)
Primary Quantity ==> 10

Edit JCL ==> NO             (Yes/No)

ENTER to process or enter END command to terminate

```

Perform the following steps to create or delete a Repository dataset:

1. Enter a **C** on the **COMMAND** line to create a dataset or a **D** to delete a dataset.
2. Enter the name of the dataset to be created or deleted in the **Repository File** field.

The name can be qualified with quotes or it can be unqualified. Unqualified names will have the user prefix added to the beginning of the name.

3. During the create process only, do the following:
  - If your dataset is an SMS managed dataset, enter an SMS management class or storage class in the appropriate field. The Volume and Unit fields should be left blank. The system will automatically assign them.
  - If your dataset is a non-SMS dataset, enter the specific volume serial number or the generic unit name for the dataset in the appropriate field. When a unit name is specified, a sequential dataset is dynamically allocated to a volume on that unit. The IDCAMS batch job will delete that sequential dataset and allocate the VSAM dataset on that volume.
  - Enter cylinders (CYL), records (REC), tracks (TRK), or megabytes (MB) in the Space Units field and the desired number in the Primary Quantity field. The default size of 10 cylinders should provide space for system(s) consisting of about 1000 programs, with each having 800 executable statements. The size of each record is:

$$(8 \times \text{number of executable verbs}) + 160.$$

4. If you would like to review the JCL before submitting it, type **Y** or **YES** in the **Edit JCL** field.



The job card previously specified on the Code Coverage JOB INFORMATION screen is automatically inserted. The JCL can be modified before submitting the job.

### Task 4.3 Global Filter File Implementation

The Global Filter File is intended to provide your site with a common library for filters that can be copied for use across the enterprise. By using your security system, you can control who has access to the Global Filter File.

To create a Global Filter File, use a PDS or PDSE dataset with LRECL=150 and RECFM=FB. The BLKSIZE can be determined based on your site's requirements. Space requirements are dependent on the quantity and size of the filters you save. Each filter is equal to one record in the Filter File member, while the filter description also occupies one record in each Filter File member.



The Retrieve Filters screens that display the Global Filter File name do not allow the name to be modified.

The variable GBLFDSN contains the name of the Global Filter File and can be up to 44 characters long. The initial value of this variable is set to blanks, indicating that the Global Filter File is not being used. To change the value of GBLFDSN so that the Global Filter File will be used, follow the instructions in member JCLGLOBL in the SLXVSAMP library. See [Figure 2](#).

1. Edit the JCL in member JCLGLOBL in the SLXVSAMP library as follows:
  - a. Insert a valid job card.
  - b. Change the high-level qualifier to your site's high-level qualifier for all the libraries pointed to by the **CHANGE** identifier.
  - c. Type the fully-qualified dataset name of the Global Filter File enclosed in quotes immediately following the literal **GBLFDSN=**. The correct line in the JCL is pointed to by the **GLOBAL** identifier.
2. Submit the job. This will create a module that contains the name of your Global Filter File. The job should end with a **CC=0**.

After you have implemented the Global Filter File, if you decide not to use it, change the value of variable GBLFDSN to blanks and resubmit the job. The job should end with a **CC=0**.

Figure 2 JCL to Enable the Global Filter File

```

//*YOUR JOBCARD
/*-----*
/* - ADD A JOB CARD. *
/* *
/* - CHANGE CPWR.XV TO YOUR HIGH LEVEL QUALIFIER FOR THE *
/* CODE COVERAGE LIBRARIES. THE LINES THAT NEED TO BE CHANGED *
/* ARE IDENTIFIED BY THE LITERAL '<=== CHANGE' IN COLUMN 58. *
/* *
/* - CHANGE CPWR.GLOBAL.FILTER.FILE TO YOUR FILTER FILE NAME. *
/* THE LINE THAT NEEDS TO BE CHANGED IS IDENTIFIED BY THE *
/* LITERAL '<=== GLOBAL' IN COLUMN 58. REFER TO THE *
/* XPEDITER/CODE COVERAGE INSTALL MANUAL FOR A DESCRIPTION OF *
/* THE PARAMETER. *
/* *
/* - SUBMIT THE JOB. IT SHOULD END WITH A CC=0. *
/* *
/*-----*
//ASM EXEC PGM=ASMA90,
// PARM='OBJ,RENT,XREF(FULL),MXREF'
//SYSIN DD *
        XVTGLOBL ,
                GBLFDSN='CPWR.GLOBAL.FILTER.FILE' <=== GLOBAL X
//SYSLIB DD DISP=SHR,DSN=SYS1.MACLIB
// DD DISP=SHR,DSN=CPWR.XV.SLXVSAMP <=== CHANGE
//SYSUT1 DD UNIT=SYSDA,SPACE=(TRK,(2,5))
//SYSUT2 DD UNIT=SYSDA,SPACE=(TRK,(2,5))
//SYSLIN DD DISP=(NEW,PASS),
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=4000),
// SPACE=(CYL,(1,1)),UNIT=VIO
//SYSPUNCH DD DUMMY
//SYSPRINT DD SYSOUT=*
/**-----*
//LKED EXEC PGM=HEWL,COND=(0,NE,ASM),
// PARM='MAP,LET,REUS,RENT,AMODE(31),RMODE(ANY)'
//SYSLIB DD DISP=SHR,DSN=CPWR.XV.SLXVLOAD <=== CHANGE
//SYSLMOD DD DISP=SHR,
// DSN=CPWR.XV.SLXVLOAD <=== CHANGE
//SYSOBJ DD DISP=(SHR,PASS),
// DSN=*.ASM.SYSLIN
//SYSUT1 DD DSN=&SYSUT1,UNIT=SYSDA,SPACE=(CYL,(3,2))
//SYSPRINT DD SYSOUT=*
//SYSLIN DD *
INCLUDE SYSOBJ
ENTRY XVTAGBL
NAME XVTAGBL(R)

```

## Task 4.4 Limit Record Deletion Capabilities

If your site wishes to restrict the deletion of records in the Code Coverage List/Delete utility to specific individuals, this function is available through a generic or specific RACF dataset profile. Add ALTER authority to those users that will be allowed to delete any record in the dataset. When Code Coverage attempts to delete a record, it will issue a RACROUTE macro to determine if the user deleting the records has ALTER authority. If the user does not have authority to delete the records, they only will be permitted to delete records containing their own userID.

# Milestone 5: Configure Xpediter/Code Coverage — Upgrade

This milestone will guide you through the configuration of an upgrade to Xpediter/Code Coverage 17.02.



If you are performing a new installation instead, skip ahead to ["Milestone 6: Configure CICS for Use with Xpediter/Code Coverage"](#).



Roles involved:  
Xpediter/Code Coverage Installer

Complete the following tasks to configure an upgrade to Xpediter/Code Coverage.

## Task 5.1 Update ISPF Support



If you are using Xpediter/Code Coverage with Xpediter/TSO and Xpediter/IMS, refer to the *Xpediter/TSO and Xpediter/IMS Installation and Configuration Guide* for instructions on configuring XPLIBDEF with Xpediter/Code Coverage.

### Task 5.1.1 ISPF Support

If you had modified your TSO logon PROC for use with Xpediter/Code Coverage, you need to ensure that the correct (new) datasets are specified.

The recommended method is via the LIBDEF command. Copy SLXVSAMP members XPCOVER and XVTTGDN into a library in your SYSPROC concatenation.

This CLIST library **must** be allocated to your ISPF session.



You can display the libraries in your SYSPROC concatenation by entering the command  
TSO ISRDDN.

By default, the XPCOVER CLIST will use the Simple Deploy method to allocate the dataset names specified in the CSMC PARMLIB DDSN\* member (DDSN00 by default). If you are not using the default member, make sure to point to the DDSN\* member name by setting the DDSNSFX keyword to the one- to four-character PARMLIB member suffix.

For example, to point to PARMLIB member DDSNCPWR, use:

```
DDSNSFX(CPWR)
```

In addition, the primary CMSC will be used by default. If you are not using the default CMSC, specify the CMSC to use by setting the CMSC keyword to the one- to four-character CMSC ID.

For example, to point to CMSC subsystem TEST, use:

```
CMSC(TEST)
```

The XPCOVER CLIST can alternately use High-Level Qualifiers (HLQ) to allocate the Xpediter/Code Coverage datasets it references. See the comments in the CLIST for more information.

## Task 5.2 Upgrade Xpediter/Code Coverage with Xpediter/CICS



If you are not using Xpediter/Code Coverage with Xpediter/CICS, skip ahead to ["Milestone 8: Verify the Xpediter/Code Coverage Installation"](#).

### Task 5.2.1 Modify JCL Template Dataset for CICS

Review the XV.XVTCEXTJ dataset(s) in the STEPLIB. DSNs may need to be updated to specify the new SLXVLOAD and SLCXLOAD datasets.



**A single JCL template dataset can be shared by Xpediter/CICS in multiple CICS regions if the DSN for the template dataset is specified in the resource definition to allow CICS to dynamically allocate and deallocate this file.**

### Task 5.2.2 Update the CICS Startup JCL

Change your site's CICS startup JCL by adding the load library containing Code Coverage to the DFHRPL concatenation. CPWR.MLXV170.SLXVLOAD is the default name for this loadlib.

### Task 5.2.3 Convert Global Table Override Input

If you are a current Xpediter/Code Coverage customer utilizing the XVGBLINP DD to specify global table overrides, be aware that you will have to convert your input to use the Compuware PARMLIB. You **must** move your current override members from the datasets in the XVGBLINP DD into new members with the names of XVGBxxxx in the Compuware PARMLIB. You can remove the XVGBLINP DD, which will no longer be needed, from your CICS JCL at your convenience.

You will now use the Compuware PARMLIB Index Member, XD\$\$\$00, to specify the members and order of concatenation for each CICS region in which you have Xpediter/Code Coverage installed. This member will be read during product initialization and provides the order of members to be read during product initialization or when the XVSI transaction is used to refresh parameter changes.

Refer to the chapter entitled "Milestone 3: Compuware PARMLIB Implementation" in the *Xpediter/CICS Installation and Configuration Guide* and ["Milestone 7: Using Compuware PARMLIB to Configure Xpediter/Code Coverage with Xpediter/CICS"](#) of this manual for additional instructions.

# Milestone 6: Configure CICS for Use with Xpediter/Code Coverage

This milestone will guide you through the configuration of CICS for use with a new installation of Xpediter/Code Coverage 17.02.



If you are not using CICS with Xpediter/Code Coverage, skip ahead to ["Milestone 8: Verify the Xpediter/Code Coverage Installation"](#).



Roles involved:  
Xpediter/Code Coverage Installer  
z/OS System Programmer.

## Preliminary CICS Considerations

Before Code Coverage can be used under Xpediter/CICS, several considerations must be taken into account:

- All dataset members referred to in this section can be found in the Xpediter/CICS SMXDSAMP library.
- A Program Inventory file must be created. This is normally done during Code Coverage installation. See [Create and Format the Program Inventory for CICS](#) on page 33 for more information.
- The Program Inventory file must be populated with the names of all the programs for which you want to collect data. You can use COLLECT control statements from within the Populator to designate specific programs for Code Coverage monitoring, or use the Program Inventory Utility to select covered programs. For more information, refer to Chapter 12, "CICS Populator" and Chapter 13, "Program Inventory Utility" in the *Xpediter/Code Coverage Mainframe User/Reference Guide*.
- You must ensure the value of the Xpediter/CICS global parameter `ACTIVATE_CODE_COVERAGE_AT_INITIALIZATION` is set to YES (the default).
- You must ensure the value of the Xpediter/CICS Code Coverage global parameters `TRANSID_XVCC` and `TRANSID_XVSC` match the transaction codes which default to XVCC and XVSC. Optionally, you may use the global parameters `TRANSID_XVCC` and `TRANSID_XVSC` to change these transaction identifiers. Changing the values requires changing the supplied resource definitions.
- You must ensure the load library containing Xpediter/CICS has been added to the DFHRPL concatenation. This loadlib is provided as `CPWR.cMXDmm.SMXDOxxL`.
- You must ensure the load library containing Code Coverage has been added to the DFHRPL concatenation. `CPWR.MLXV170.SLXVLOAD` is the default name for this loadlib.

- You must ensure a DD statement for the Code Coverage Program Inventory dataset XVTPGINV has been added to your site's startup JCL. **Do not define this file with RDO.**



If this DD does not point to a valid program inventory dataset, you will receive a message stating that Code Coverage is not active. If Code Coverage is not active, refer to the joblog regarding messages related to this dataset.

- You must create an Extract dataset (XVTCEXTR) for the intermediary Code Coverage data. See [Create an Extract Dataset for CICS](#) on page 36 for more information.
- Compuware strongly recommends that you create a JCL Template dataset (XVTCEXTJ) that contains the JCL to write the intermediary Code Coverage data to the Code Coverage repository. See [Create JCL Template Dataset for CICS](#) on page 37 for more information.
- Optionally, you can create a Save Specifications dataset (XVTCSPEC) to provide an archive for Code Coverage test definitions. See [Create the XVTCSPEC Dataset for CICS](#) on page 39 for more information.

## Specification Dataset Access

The XVTCSPEC dataset (Specification Dataset) is accessed from both the Test Definition and Specification Archive screens. On the Specification Archive screen, the user has the ability to delete a record. If a delete of a record is requested, the product programmatically performs a call to see if the user is authorized to use the XVSC transaction. If they are authorized to use this transaction, they are considered to be an "Administrator", and the record will be deleted. If they are not authorized to use the XVSC transaction, they are considered to be a "Standard" user, and the record can only be deleted if their userID matches the ownerID of the record. Refer to the section [XVCC/XVSC Transaction Security](#) for a discussion of product privileges.

## XVCC/XVSC Transaction Security

The Xpediter/CICS component of Xpediter/Code Coverage provides two transactions to gain access to the product to define and/or view collection specifications. A user who has been granted authority to use the XVCC transaction is considered to be a "Standard" user. A user who has been granted authority to use the XVSC transaction is considered to be an "Administrator". The demarcation between a "Standard" user and an "Administrator" is noted in the following table.

**Table 7** Standard and Administrator Transaction Authority

User Type	Transaction Authority		Privileges
	XVCC	XVSC	
Standard	Yes	No	Access to all screens in the Xpediter/CICS component of Xpediter/Code Coverage. Deletion of records in the Specification Dataset is limited to records where their userID matches the ownerID of the record.
Administrator	Yes	Yes	All the same access as a "Standard" user but can delete any record in the Specification Dataset.

## Dataset Security Requirements

[Table 8](#) lists the required security for several Xpediter/Code Coverage datasets.

**Table 8** Dataset Security

Dataset	Security Requirements
XVTCPGINV	Users who will populate the program inventory after a compile will require READ and UPDATE authority.
XVTCEXTR	The hardening utility job will require the userID associated with the job to have READ and UPDATE authority. In the CICS region, authority is associated with the userID of the region, and READ and UPDATE access is required.

**Table 8** Dataset Security

Dataset	Security Requirements
XVTCSPEC	Authority is associated with the userID associated with the CICS region. READ, UPDATE, and DELETE access is required. Refer to the section <a href="#">XVCC/XVSC Transaction Security</a> on page 30 for a discussion of privileges associated with users of those transactions.
XVTCEXTJ	Authority is associated with the userID associated with the CICS region. READ access is required.

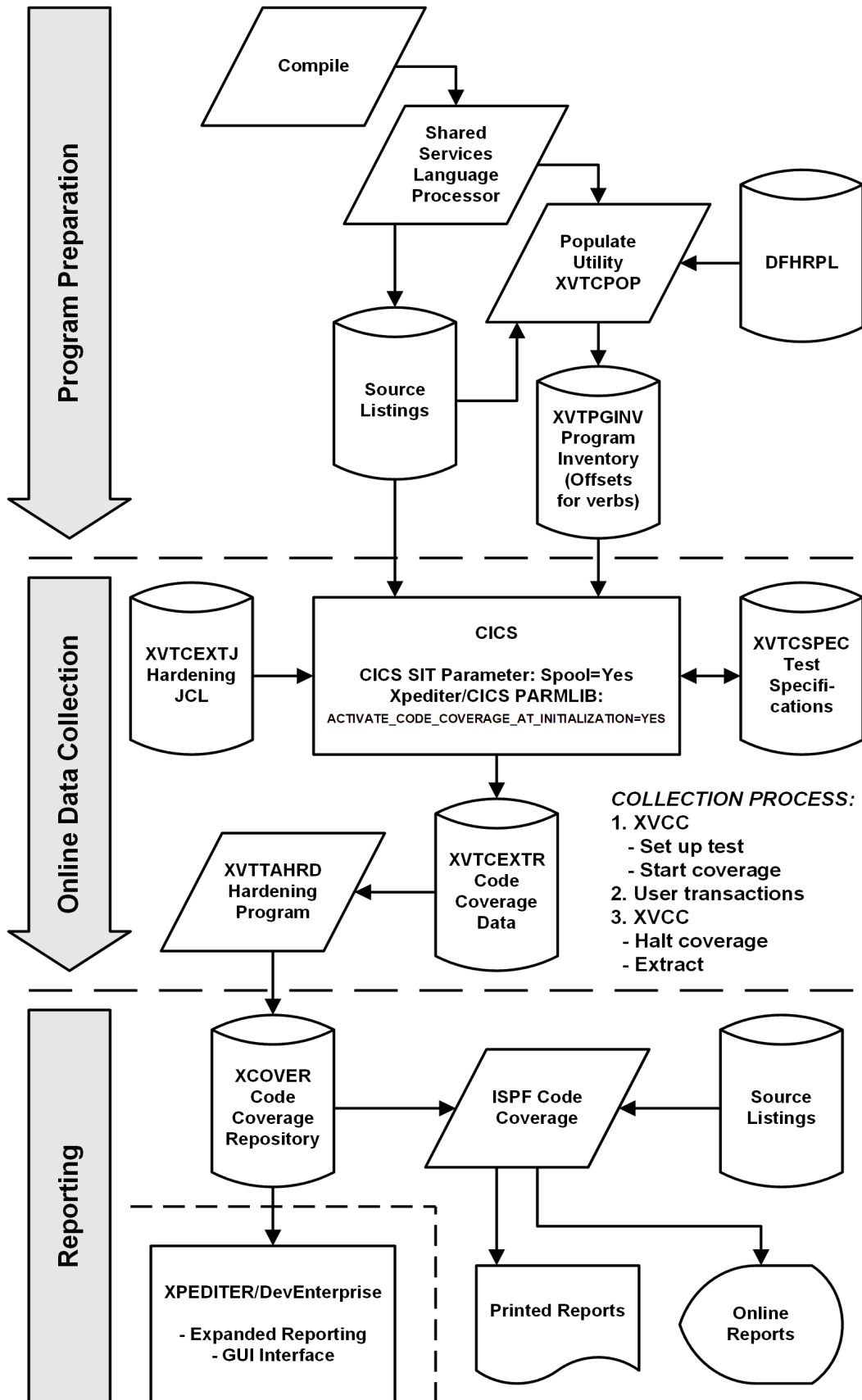
## Overview of Using Code Coverage with Xpediter/CICS

Using Code Coverage with Xpediter/CICS is a three-stage process:

- Program preparation
- Online data collection
- Reporting.

The entire process is illustrated in [Figure 3](#) on page 32. For more information, see the *Xpediter/Code Coverage Mainframe User/Reference Guide*.

Figure 3 Xpediter/Code Coverage Process in a CICS Environment





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

## Task 6.1 Create and Format the Program Inventory for CICS

Xpediter/Code Coverage can collect data on selected CICS programs during testing with Xpediter/CICS. A Program Inventory (PGMINV) designates the programs for which Code Coverage collects data. This Program Inventory also enhances the efficiency of the Code Coverage data collection function in a CICS region. It contains an entry for each CSECT in a CICS region's DFHRPL concatenation that has an Xpediter source listing file entry with a matching date and time stamp.

The JCL used to create a Program Inventory can also be used to reformat an existing one. If an existing PGMINV is reformatted, the dataset's secondary allocation can be used. Keep in mind that when you reformat an existing Program Inventory, all data in the dataset will be erased.

### Task 6.1.1 Determine Appropriate Size for the Program Inventory Dataset

Each member of the Program Inventory dataset will occupy at least one block, assuming half-track blocking. Because one block can accommodate approximately 1750 verbs, 5000 programs of that size or smaller would require roughly 175 cylinders. Larger programs would require roughly one additional block for every additional 1750 verbs.

For an approximation of program sizes, run the Populate Utility for a representative sample of the programs. Then use XVTCUTIL to produce a directory listing containing the verb count for each program. Overall statistics are included at the bottom of the listing.

On the FORMAT statement in member JCLALLOC, BLOCKS should be  $30 \times$  cylinders (with half-track blocking), and DIRENTS should equal the maximum number of members.

### Task 6.1.2 Create a Program Inventory Dataset



Reformatting an existing Program Inventory will erase all data in the dataset.

The Program Inventory dataset must be created and formatted before it can be used by Xpediter/Code Coverage. The necessary JCL is provided in Xpediter/CICS SMXDSAMP member JCLALLOC ([Figure 4](#) on page 35 below). Once you have formatted a Program Inventory, you cannot alter its space allocation (number of TRKS or CYLS) or DCB attributes (RECFM, LRECL, or BLKSIZE). In order to change DCB attributes, a Program Inventory must be re-allocated and formatted.

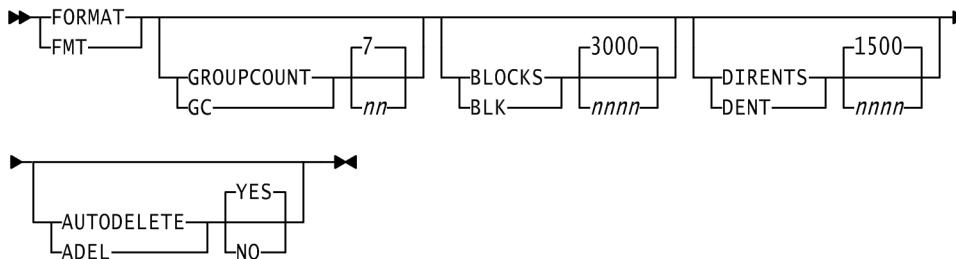
1. Make the necessary edits to the JCL used to create the PGMINV dataset.

The DCB attributes of the Program Inventory dataset are RECFM=FBS, and the LRECL/BLKSIZE must be equal. The BLKSIZE determines the size of a physical record and will default to half-track blocking. The space allocation must allow enough primary and secondary extents to cover the

number of blocks to be formatted. The number of blocks to be formatted is specified on a FORMAT command using the keyword parameters described below.



**The ISPF Program Inventory Utility cannot be used to format a CICS Program Inventory file.**



### GROUPCNT

The GROUPCNT keyword parameter, abbreviated GC, is used to specify the number of blocks which comprise an extent. For example, if GROUPCNT is 7, then space for each program inventory member will be allocated in increments of 7 blocks. This number must be between 7 and 31 (inclusive). The default is 7.

### BLOCKS

The BLOCKS keyword parameter, abbreviated BLK, represents the number of physical records to be formatted. The size of a physical record is specified by the DCB BLKSIZE attribute. Regardless of the JCL SPACE allocation, only this number of BLKSIZE records will be formatted. The default is 3000. When using a half-track blocking factor, this default equates to 1500 tracks or 100 cylinders.

### DIRENTS

DIRENTS, abbreviated DENT, specifies the minimum number of directory entries to be allowed. The value specified should be greater than the maximum number of programs to be tested with Xpediter/Code Coverage. The PGMINV directory is a pre-allocated, contiguous area within the Program Inventory. This number cannot be changed after PGMINV has been formatted. The default is 1500. Any reasonable value for the number of entries (that is, a value not exceeding the available blocks) can be specified.

### AUTODELETE

AUTODELETE, abbreviated ADEL, optionally specifies that old members may be deleted by the Populate Utility if space is needed to add a new member. If NO is specified for AUTODELETE, the attempt to add the new member will fail. If YES is specified for AUTODELETE, old members will be deleted until sufficient space is available for the new member. All deletes are logged by the Populate Utility. The default is YES.

2. When the necessary edits have been performed and a valid job card inserted, submit the JCL to create and format the Program Inventory.

### Notes:

- The installer does not need to load the Program Inventory or designate which programs are to be tested with Code Coverage. These steps should be performed later by the user. For more information, refer to the chapter entitled "CICS Populater" in the *Xpediter/Code Coverage Mainframe User/Reference Guide*.
- A single Program Inventory can be shared by Xpediter/CICS in multiple CICS regions.

- Users of the Program Inventory require UPDATE access.
- Allocating the Program Inventory dataset on multiple DASD volumes is not supported and will result in error XVTC8306E being issued during product initialization.

**Figure 4 JCL to Create and Format the Program Inventory (JCLALLOC)**

```

/** YOUR JOB CARD HERE
/** * * * * *
/**
/** CODE COVERAGE INSTALL STEP: JCLALLOC
/**
/** * * * * *
/** JCLALLOC: EXECUTE THE CODE COVERAGE UTILITY PROGRAM XVTCUTIL.
/** CODE COVERAGE UNDER CICS USES A SPECIAL FORMAT FILE
/** CALLED THE "PROGRAM INVENTORY FILE". THIS UTILITY IS
/** DESIGNED TO MANIPULATE THE PROGRAM INVENTORY FILE AND
/** THE MEMBERS IT CONTAINS.
/**
/** THIS JCL MEMBER IS USED TO ALLOCATE AND FORMAT A
/** PROGRAM INVENTORY FILE. USE JCL MEMBER JCLCUTIL FOR
/** ALL OTHER UTILITY FUNCTIONS.
/**
/** KEY DDNAMES AND THEIR FUNCTION
/** -----
/** CONTROL - SPECIFIES ACTION(S) TO BE PERFORMED.
/**
/** XVTPGINV - NAMES THE PROGRAM INVENTORY FILE TO BE MANIPULATED.
/**
/**
/** XVTCUTIL CONTROL STATEMENT(S)
/** -----
/** FORMAT GROUPCNT=7,BLOCKS=3000,DIRENTS=1500,AUTODELETE=YES
/** | | | | |
/** | | | | |
/** | | | | |
/** | | | | | - DEL OLDEST WHEN NOSPACE
/** | | | | | - NBR OF MEMBERS IN FILE
/** | | | | | - NBR OF BLOCKS TO FORMAT (DCB=BLKSIZE)
/** | | | | | - NBR OF BLOCKS PER GROUP (FROM 7-31)
/** | | | | | - BUILD/INITIALIZE A "PROGRAM INVENTORY FILE"
/**
/** ONLY SPACE= AND DISP= ARE REQUIRED FOR DATASET ALLOCATION OF A
/** PROGRAM INVENTORY FILE. AUTODELETE DEFAULTS TO "YES", BUT CAN
/** BE SPECIFIED AS "NO". INITIAL ALLOCATION CAN BE PERFORMED IN
/** CONJUNCTION WITH THE "FORMAT" COMMAND.
/**
/** *****
/**
/**STEP1 EXEC PGM=XVTCUTIL
/**STEPLIB DD DISP=SHR,DSN=CPWR.cMXDnnn.SMXD0ccL <== CHECK DSN
/**SYSPRINT DD SYSOUT=*
/**SYSUDUMP DD SYSOUT=*
/**
/**XVTPGINV DD DSN=XPEDITER.CICS.PGMINV, <== CHECK DSN
/** SPACE=(CYL,(100,10),RLSE),
/** UNIT=SYSDA,
/** DISP=(NEW,CATLG)
/**
/**CONTROL DD *
FORMAT GROUPCNT=7,BLOCKS=3000,DIRENTS=1500
/**

```

## Task 6.2 Create an Extract Dataset for CICS

Code Coverage data collecting during Xpediter/CICS testing is written out to an intermediary extract dataset before being entered into the Code Coverage Results Repository.

1. Find the JCL to create the Code Coverage extract dataset.

This dataset is provided in Xpediter/CICS SMXDSAMP member JCLCDEFR and shown [Figure 5](#).

2. Edit the JCL according to the included comments shown in [Figure 5](#), inserting a valid job card and specifying the dataset name, volume, and space values chosen for your site.
3. When the necessary edits have been performed, submit the JCL to create the extract dataset.



**Compuware recommends that the Code Coverage extract dataset be unique for each CICS region collecting Code Coverage data. However, the extract dataset may be shared under the following conditions: each CICS region collecting data intends to write this data to the same Code Coverage repository, and the resource definition specifies the DSN for the extract dataset and Opentime(Firstref) to allow CICS to dynamically allocate and deallocate this file. Also, Code Coverage users will need sufficient authority to add records to this dataset.**

**Figure 5** JCL to Create the Extract Dataset (JCLCDEFR)

```

/* YOUR JOB CARD HERE
/* * * * * *
/*
/* CODE COVERAGE INSTALL STEP:  JCLCDEFR
/*
/* * * * * *
/*
/* THE FOLLOWING JCL WILL ALLOCATE THE XPEDITER/CICS CODE COVERAGE
/* EXTRACT DATASET.  SUPPLY THE VOLSER, UNIT, DSN AND SPACE
/* PARAMETERS.
/* THIS JCL IS IN INTENDED AS AN EXAMPLE.
/* PLEASE CONSULT THE INSTALLATION GUIDE BEFORE RUNNING.
/*
//AMS      EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
DELETE   (XV.XVTCEXTR) PURGE           /* CHECK DSN           */
DEFINE CLUSTER                          /* CHECK DSN           */ -
  (NAME(XV.XVTCEXTR)                    /* <== CHECK VOLUME */ -
   VOLUMES(?????)                      /* <== CHECK SPACE  */ -
   CYLINDERS(10 10)                     /* DO NOT MODIFY    */ -
   RECORDSIZE(8 32760)                  /* DO NOT MODIFY    */ -
   REUSE                                 /* DO NOT MODIFY    */ -
   SPANNED                               /* DO NOT MODIFY    */ -
   NONINDEXED                           /* DO NOT MODIFY    */ -
   SHAREOPTIONS(1 3))                  /* DO NOT MODIFY    */

```

## Task 6.3 Create JCL Template Dataset for CICS

Immediately after Code Coverage data is written out to the extract dataset, JCL is automatically executed to reformat the data and transfer it to the Code Coverage Results Repository. This JCL resides in the Code Coverage JCL template dataset.



The global parameter `AUTO_SUBMIT_JCL_DURING_EXTRACTION` can be used to disable the automatic formatting and transfer of data to the Code Coverage Results Repository. If you prefer to manually submit a job whenever you would like the transfer of data to the Code Coverage Results Repository to occur, refer to the Xpediter/CICS SMXDSAMP member JCLCHRD.

1. Find the JCL to create and load the Code Coverage JCL template dataset. It is provided in Xpediter/CICS SMXDSAMP member JCLCDEFJ as shown in [Figure 6](#).
2. Edit the JCL according to the included comments shown in [Figure 6](#), inserting valid job cards and specifying the dataset names, volume, and space values chosen for your site.
3. When the necessary edits have been performed, submit the JCL to create the JCL template dataset.



**A single JCL template dataset can be shared by Xpediter/CICS in multiple CICS regions if the DSN for the template dataset is specified in the resource definition to allow CICS to dynamically allocate and deallocate this file.**

Figure 6 JCL to Create the JCL Template Dataset (JCLCDEFJ)

```

/** Your Job card here
/** * * * * *
/**
/** Code Coverage install step: JCLCDEFJ
/**
/** * * * * *
/** JCLCDEFJ - IDCAMS JCL to allocate the Xpediter/CICS Code Coverage
/**      JCL template dataset.
/**
/** The following JCL will allocate and load the Xpediter/CICS Code
/** Coverage JCL template dataset.
/**
/** Supply the volser, unit, DSN and space parameters.
/**
/** This JCL is intended as an example. Please consult the
/** installation guide before running. Also, refer to member JCLCHRD.
/**
/** The following symbolics are translated at submit time:
/** %A is replaced by the VTAM applid of the CICS region.
/** %D is replaced by the DSN on the FILE definition for XVTCEXTR.
/** %J is replaced by the CICS jobname.
/** %S is replaced by the SYSID of the CICS region.
/** %U is replaced by the UserID initiating the Extract.
/**
/** ===== CAUTION =====
/** The symbolics may be used anywhere in the template JCL. If using a
/** symbolic for the jobname, be careful to avoid creating a duplicate
/** of the jobname/stcname of the CICS region.
/** ===== CAUTION =====
/**
/**AMS      EXEC PGM=IDCAMS
/**SYSPRINT DD SYSOUT=*
/**SYSIN    DD *
DELETE (XV.XVTCEXTJ) PURGE          /* CHECK DSN      */
DEFINE CLUSTER
  (NAME(XV.XVTCEXTJ)                /* CHECK DSN      */ -
  VOLUMES(?????)                  /* <== CHECK VOLUME */ -
  TRACKS(1 1)                      /* <== CHECK SPACE  */ -
  RECORDSIZE(80 80)                /* DO NOT MODIFY   */ -
  NUMBERED                          /* DO NOT MODIFY   */ -
  SHAREOPTIONS(1 3))               /* DO NOT MODIFY   */

/*          LOAD JCL DATA          */

IF LASTCC EQ 0                      -
THEN REPRO                          -
      INFILE(SYSJIN)                -
      OUTDATASET(XV.XVTCEXTJ)       /* CHECK DSN      */

/*
/**SYSJIN DD DATA,DLM=##
/**EXTRACT JOB ('XPEDITER CODE COVER', <== YOUR JOB CARD HERE
/**          CLASS=A,MSGCLASS=X,NOTIFY=%U
/**%J EXEC PGM=XVTTAHRD
/**STEPLIB DD DISP=SHR,DSN=COMPWARE.MLXVnnn.SLXVLOAD <== CHECK DSN
/**          DD DISP=SHR,DSN=COMPWARE.LCXnnn.SLCXLOAD <== CHECK DSN
/**XEXTRACT DD DISP=SHR,DSN=%D
/**XCOVER   DD DISP=SHR,DSN=YOUR.REPOSITORY <== CHECK DSN
/**SYSUDUMP DD SYSOUT=*
/**SYSPRINT DD SYSOUT=*
/**
/**##
/**

```

## Task 6.4 Create the XVTCSPEC Dataset for CICS

If you intend to archive any of your test specifications from the Test Definition screen, these will be saved in the XVTCSPEC file. XVTCSPEC is an optional file, but must be defined if specification archiving is to be done.

1. Find the JCL to create the Specifications Archive dataset.

The JCL provided in Xpediter/CICS SMXDSAMP member JCLCDEFS as shown in [Figure 7](#).

2. Edit the JCL according to the included comments shown in [Figure 7](#), inserting a valid job card and specifying the dataset name, volume, and space values chosen for your site.
3. When the necessary edits have been performed, submit the JCL to create the XVTCSPEC dataset.



**A single XVTCSPEC dataset can be shared by Xpediter/CICS in multiple CICS regions running on the same MVS image, but *not* across MVS images.**





3. Add program DBUGPLTS to your CICS shutdown PLT so that Code Coverage can extract any collected data during CICS shutdown or region cancel.

- a. To add this program to your CICS shutdown PLT, look for the following line:

```
DFHPLT TYPE=ENTRY,PROGRAM=DFHDELIM
```

- b. Immediately before the line, add the following entry:

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



You may skip the next sub-step if you plan to change the value of the Xpediter/Code Coverage global parameter `AUTO_SUBMIT_JCL_DURING_EXTRACTION` to **NO** when you perform [Review Global Parameters](#) on page 41. If you skip this sub-step, go to [Update the CICS Startup JCL](#) on page 41 immediately below.

4. Modify your CICS SIT parameter SPOOL.

Be sure to set `SPOOL=YES` in your CICS system initialization table (SIT) or SIT overrides. This will enable Code Coverage to automatically submit JCL to reformat the contents of the extract dataset and transfer it to the Code Coverage Results Repository.

## Task 6.6 Update the CICS Startup JCL

Change your site's CICS startup JCL as follows:

1. Add the load library containing Code Coverage to the DFHRPL concatenation. CPWR.MLXV170.SLXVLOAD is the default name for this loadlib.
2. Add a DD statement for the Code Coverage Program Inventory dataset XVTPGINV. **This file is not to be defined with RDO.**



If this DD does not point to a valid program inventory dataset, you will receive a message stating that Code Coverage is not active. If Code Coverage is not active, refer to the joblog regarding messages related to this dataset.

3. An RDO entry for file XVTCEXTR was created in [Update the CICS Resource Definitions](#) on page 40. If that entry did not include a dataset name, add a DD statement to the CICS startup JCL for XVTCEXTR.
4. An RDO entry for file XVTCEXTJ was also created in [Update the CICS Resource Definitions](#) on page 40. If that entry did not include a dataset name, add a DD statement to the CICS startup JCL for XVTCEXTJ.
5. An RDO entry for file XVTCSPEC was also created in [Update the CICS Resource Definitions](#) on page 40. If that entry did not include a dataset name, add a DD statement to the CICS startup JCL for XVTCSPEC. (This step is optional if you do not wish to archive test specification entries.)

## Task 6.7 Review Global Parameters

Compuware recommends that your site install and test Code Coverage with the shipped options. You may, however, want to review the global parameters listed in ["Milestone 7: Using Compuware PARMLIB to Configure Xpediter/Code Coverage with Xpediter/CICS"](#) and make any changes required for your site.

## Verifying the Code Coverage Installation

To verify the installation of Code Coverage you must complete those steps detailed in ["Milestone 8: Verify the Xpediter/Code Coverage Installation"](#) after all the steps in the current chapter are completed.



# Milestone 7: Using Compuware PARMLIB to Configure Xpediter/Code Coverage with Xpediter/CICS



If you are not using Xpediter/Code Coverage with Xpediter/CICS, skip ahead to ["Milestone 8: Verify the Xpediter/Code Coverage Installation"](#).



Refer to "Task 5.5 Migrate Existing Parameters to Compuware PARMLIB" in the *Xpediter/CICS Installation and Configuration Guide* for important information regarding the Compuware PARMLIB.

This milestone outlines the Compuware PARMLIB parameters that can be used to configure your Xpediter/Code Coverage installation in CICS.



Roles involved:  
Xpediter/Code Coverage Installer  
Xpediter/CICS Installer.

There are two methods available for configuring these global parameters:

- Using the Compuware PARMLIB members prefixed with XVGB to supply parameters.



The above is the preferred method.

- Modifying the XVTCGBLM macro parameter values and reassembling and relinking load module XVTCGBL. Its source can be found in member XVTCGBLM of the library CPWR.cMXD*nnn*.SMXDSAMP (where *c* varies by CICS release and *nnn* represents the release of Xpediter).

If this is your first installation of Xpediter/Code Coverage, Compuware suggests that you install and test the product with the shipped global parameters. A default XVTCGBL module is included with the system. The default values supplied for each parameter in XVTCGBLM are shown in [Table 9](#).

## Specifying Xpediter/CICS Code Coverage Parameters at Product Initialization

Xpediter/Code Coverage includes a facility that allows you to override the current parameter settings in the XVTCGBL load module at product initialization. This facility can also be invoked while Xpediter/Code Coverage is active by entering the transaction XVSI. Refer to [Overriding Parameters](#) later in this chapter for complete information.

When overriding the parameters, a Compuware PARMLIB report (Xpediter/CICS Code Coverage Parameters) is written to a transient data queue. The default queue name is XVGR, which should have been defined using RDO when Xpediter/CICS was installed. Refer to Step 1 in “Task 4.4.1 Using RDO to Update Resource Definitions” in the *Xpediter/CICS Installation and Configuration Guide*.

## Parameter Ineligible for XVSI Transaction

The LOG\_TDQ\_NAME parameter can't be overridden with the XVSI transaction. It may however be overridden when Xpediter/Code Coverage begins initialization. For more information, see [Overriding Parameters](#) on page 47.

## Global Parameter Default Values and Descriptions

[Table 9](#) lists all the Compuware PARMLIB parameters related to Xpediter/Code Coverage with Xpediter/CICS, XVTCGBL macro parameter names, and the default values for each. Detailed descriptions of all the parameters are provided in the “Parameter Descriptions” section of this Milestone.

**Table 9** Global Parameters

PARMLIB Parameter	XVTCGBL Macro/PARMLIB Version 1 Parameter	Default in XVTCGBL
AUTO_SUBMIT_JCL_DURING_EXTRACTION	SUBMIT	YES
CAPTURE_SYSTEM_FLOW_USING_GMT_TIME	CTLGMT	YES
CODE_COVERAGE_PARMLIB_REPORT_ALLOCATION	<b>Deprecated</b>	
CODE_COVERAGE_PARMLIB_REPORT_DDNAME	XVGBLRP	XVGR
CODE_COVERAGE_PARMLIB_REPORT_SYSOUT_CLS	<b>Deprecated</b>	
DATA_EXTRACTION_INTERVAL	INTERVAL	42
INCREMENT_VERB_COUNTS	COUNT	YES
KEEP_PROGRAM_COUNTS_FOR_USERID	PROGCNTS	*****
LOG_INFO_MESSAGES	WTDQ	YES
LOG_TDQ_NAME	LOGQ	CSMT
PRIME_OWNER_FILTER_WITH_CURRENT_USERID	FILTDEFU	NO
TRANSID_XVCC	XVCC	XVCC
TRANSID_XVKP	XVKP	XVKP
TRANSID_XVSC	XVSC	XVSC
TRANSID_XVSI	XVSI	XVSI
TRANSID_XVTQ	XVTQ	XVTQ
WRITE_DATA_EXTRACTION_MSGS_TO_JES_LOG	WTO	YES

## Parameter Descriptions

This section describes each parameter. The default values listed are those in XVTCGBL.

### AUTO\_SUBMIT\_JCL\_DURING\_EXTRACTION

**Default:** YES

Automatically submit the Code Coverage template JCL during extraction. Enter **NO** to disable the automatic submission during extraction. If AUTO\_SUBMIT\_JCL\_DURING\_EXTRACTION is set to NO,

the JCL to write the extracted data to the Code Coverage repository would need to be submitted manually.

## CAPTURE\_SYSTEM\_FLOW\_USING\_GMT\_TIME

**Default:** YES

Optionally capture System Flow data with local time (CAPTURE\_SYSTEM\_FLOW\_USING\_GMT\_TIME=NO). Specifying CAPTURE\_SYSTEM\_FLOW\_USING\_GMT\_TIME=YES will capture System Flow data with GMT/UTC.

## CODE\_COVERAGE\_PARMLIB\_REPORT\_ALLOCATION

This parameter has been deprecated.

## CODE\_COVERAGE\_PARMLIB\_REPORT\_DDNAME

**Default:** XVGR

A one- to four-character transient data queue name that the Compuware PARMLIB report (Xpediter/CICS Code Coverage Parameters) is written to.

## CODE\_COVERAGE\_PARMLIB\_REPORT\_SYSOUT\_CLS

This parameter has been deprecated.

## DATA\_EXTRACTION\_INTERVAL

**Default:** 42

The number of minutes between automatic data extractions. Valid values are from 1 to 1440.

## INCREMENT\_VERB\_COUNTS

**Default:** YES

Set this option to **NO** when it is not necessary to count each time a verb is executed. When INCREMENT\_VERB\_COUNTS=NO, Xpediter/Code Coverage will count each verb only the first time it is executed. To maximize the potential performance benefit of INCREMENT\_VERB\_COUNTS=NO, rules for Storage Protection and Program Trace are ignored when INCREMENT\_VERB\_COUNTS=NO and a Code Coverage test is active.



When parameter INCREMENT\_VERB\_COUNTS=NO, the System Flow feature of Xpediter/Code Coverage cannot be used. An error message is displayed if you attempt to activate either System Flow or both (Code Coverage plus System Flow).

When INCREMENT\_VERB\_COUNTS is set to **NO**:

- Only one Xpediter/Code Coverage test at a time may be active in the CICS region.
- Each application program you execute is marked as RESIDENT until every verb within it has been executed or the Xpediter/Code Coverage test has been halted. If you encounter a Short on Storage (SOS) situation using INCREMENT\_VERB\_COUNTS=NO, reduce the number of programs referenced by your Xpediter/Code Coverage test.
- CPU consumption may actually increase, depending on the structure of your Xpediter/Code Coverage test. When the same code is executed multiple times, INCREMENT\_VERB\_COUNTS=NO will reduce CPU consumption by a factor related to how many times the code is executed. However, when different code is executed by each transaction and there is very little reuse of the same code, INCREMENT\_VERB\_COUNTS=NO can cause a noticeable increase in CPU consumption.



## KEEP\_PROGRAM\_COUNTS\_FOR\_USERID

**Default:**       \*\*\*\*\*

Specifying KEEP\_PROGRAM\_COUNTS\_FOR\_USERID=\*\*\*\*\* causes program counts for a program to be kept separate by userID. Specifying KEEP\_PROGRAM\_COUNTS\_FOR\_USERID=USER1234 causes program counts to be kept under userID USER1234 for program(s) being covered, regardless of the userID(s) that executed the program(s).

## LOG\_INFO\_MESSAGES

**Default:**       YES

Used to disable the write to transient data queue function for Informational Data Extraction messages (XVTC4mm1). The queue name is specified by the LOG\_TDQ\_NAME parameter. Enter **NO** to disable.

## LOG\_TDQ\_NAME

**Default:**       CSMT

The four-character name of the transient data queue to which Code Coverage is to direct error messages. This parameter cannot be changed when Code Coverage is active. It may be changed at Xpediter/Code Coverage initialization when the Compuware PARMLIB is initially processed.

## PRIME\_OWNER\_FILTER\_WITH\_CURRENT\_USERID

**Default:**       NO

Prime the Owner column filter value with the current userID (PRIME\_OWNER\_FILTER\_WITH\_CURRENT\_USERID=YES). Prime the Owner column filter with asterisks (PRIME\_OWNER\_FILTER\_WITH\_CURRENT\_USERID=NO).

## TRANSID\_XVCC

**Default:**       XVCC

A four-character transaction ID to substitute for the XVCC transaction. Changing the value requires changing the supplied resource definition.

## TRANSID\_XVKP

**Default:**       XVKP

A four-character transaction ID to substitute for the XVKP transaction. Changing the value requires changing the supplied resource definition.



**XVKP is an internal transaction that you should never enter from a terminal. Also, because the XVKP transaction is run asynchronously, security should not be set on it.**

## TRANSID\_XVSC

**Default:**       XVSC

A four-character transaction ID to substitute for the XVSC transaction. Changing the value requires changing the supplied resource definition.

## TRANSID\_XVSI

**Default:**       XVSI

A four-character transaction ID to substitute for the XVSI transaction. Changing the value requires changing the supplied resource definition.

## TRANSID\_XVTQ

**Default:** XVTQ

A four-character transaction ID to substitute for the XVTQ transaction. Changing the value requires changing the supplied resource definition.

## WRITE\_DATA\_EXTRACTION\_MSGS\_TO\_JES\_LOG

**Default:** YES

Used to disable the write-to-operator (WTO) function for the XVTC4001 Data Extraction complete message and other Informational Data Extraction messages (XVTC4nnml). A WTO is normally issued each time a data extraction is performed. Enter **NO** to disable.

## Overriding Parameters

This section explains how to override Xpediter/Code Coverage's parameters specified in the XVTCGBL load module.

## Assembling and Link-Editing the XVTCGBL Program



This method is the less preferred method of overriding the global parameters. Compuware strongly recommends that you use the method described in [Overriding Parameters Using the Compuware PARMLIB](#) on page 47. However, if you elect to override parameters by use of the XVTCGBL program, make a backup copy of the existing XVTCGBL load and source members before modifying.

Review the applicable global parameter default values in [Table 9](#) on page 44. If your site does not use the Compuware PARMLIB and you intend to change the default values supplied, you must enter the desired value in the XVTCGBL global table, assemble it, and link-edit it. The table (XVTCGBL) should be linked with AMODE=24 and RMODE=24 (below the line) and must not be linked as reentrant. Edit and submit the JCL shown in [Figure 8](#) to assemble and link the global parameters table.

**Figure 8** Sample JCL to Assemble and Link XVTCGBL Global Parameters Table

```
//DOIT EXEC ASMHCL
//ASM.SYSLIB DD DSN=CPWR.cMXDnnn.SMXDSAMP,DISP=SHR /*-- CHECK DSN --*/
// DD DSN=SYS1.MACLIB,DISP=SHR
//ASM.SYSIN DD *
TITLE 'XVTC GLOBAL TABLE'
PRINT NOGEN
COPY XVTCGBLM
PRINT ON XVTCGBLM
INTERVAL=20 <--- MODIFY FOR DESIRED OPTIONS
END
/*
//LKED.SYSLMOD DD DSN=CPWR.cMXDnnn.SMXD0ccL(XVTCGBL),DISP=SHR /*-- DSN --*/
```

## Overriding Parameters Using the Compuware PARMLIB

This preferred method reads the desired parameter values from the Compuware PARMLIB during product initialization. This method is easy to set up and maintain, and it makes Xpediter/Code Coverage itself easier to install and maintain. Refer to “Milestone 3: Compuware PARMLIB Implementation” in the *Xpediter/CICS Installation and Configuration Guide* and “PARMLIB Members” in the *Xpediter/CICS Advanced Configuration Guide* for specific information.

In addition, a standalone transaction, XVSI, can be used to read the parameters from the Compuware PARMLIB. This eliminates the need to recycle Xpediter/Code Coverage or the CICS region when you want to make parameter changes. 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.

This section described here includes information on the following topics:

- Processing of the parameters
- Overriding report dataset ddname
- Format of the override report and console log output.

[Defining Output Report Parameters](#) on page 51 contains step by step instructions for defining the attributes for the Report Dataset produced during the override process, using the Compuware PARMLIB.

## Concatenation of Compuware PARMLIB Members

The Compuware PARMLIB Index Member XD\$\$\$00 provides the ability to concatenate members in the same fashion as sequential dataset DDs. Refer to the section entitled “Creating Additional Members for Region Customization” in the *Xpediter/CICS Advanced Configuration Guide*.

## Processing Overview

Override parameters are read from the Compuware PARMLIB.

As the input is processed, a Compuware PARMLIB Report (Xpediter/CICS Code Coverage Parameters) is written to a transient data queue which must be defined as Extrapartition. The default queue name for this report is XGBR. The CODE\_COVERAGE\_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. As each parameter is processed, it is written to the output report. See [Adding Parameters to a Member in the Compuware PARMLIB](#) on page 53 for parameter format and processing rules.

If an error is detected in a parameter override, 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.

## Xpediter/CICS Code Coverage Parameter Report

The Xpediter/CICS Code Coverage Parameter report is written to a transient data queue with a default name of XVGR. If the queue is not defined, processing will continue, but no report will be produced.

[Figure 9](#) and [Figure 10](#) detail a sample Xpediter/CICS Code Coverage Parameter report. The **highlighted** numbered references are discussed following the figures.



Figure 9 Xpediter/CICS Code Coverage Parameters Report Page 1

Compuware PARMLIB Report		Page	1
Xpediter/CICS Code Coverage Parameters			
APPLID: H01AC024	Date: 08 NOV 2017	Time: 16:48:39	USERID: ACMJETO TERM: 1718
Report TDQUEUE(XVGR) (Override from PARMLIB)			[1]
Input Record/Error Message			[2]
-----			
* Processing input from CMSC ID: CMSC			[3]
* CMSC common memory object refreshed from:			[4]
* CW01.CMSC.PARMLIB(XD\$\$\$00)			[5]
-----			
* Attempting to read data from the following parameter members:			[6]
* XVGB00 XVGBNAA0			[7]
-----			
* CMSC common memory object refreshed from:			[8]
* SYS2.CW.CMSC.PARMLIB(XVGB00)			
-----			
INCREMENT_VERB_COUNTS=YES			[8]
* CMSC common memory object refreshed from:			
* SYS2.CW.CMSC.PARMLIB(XVGBNAA0)			
-----			
CODE_COVERAGE_PARMLIB_REPORT_DDNAME=XVGR			
INCREMENT_VERB_COUNTS=YES			
CAPTURE_SYSTEM_FLOW_USING_GMT_TIME=YES			
DATA_EXTRACTION_INTERVAL=42			
LOG_TDQ_NAME=CSMT			
KEEP_PROGRAM_COUNTS_FOR_USERID=*****			
AUTO_SUBMIT_JCL_DURING_EXTRACTION=YES			
LOG_INFO_MESSAGES=YES			
WRITE_DATA_EXTRACTION_MSGS_TO_JES_LOG=YES			
TRANSID_XVCC=XVCC			
TRANSID_XVCP=XVCP			
TRANSID_XVSC=XVSC			
TRANSID_XVSI=XVSI			
TRANSID_XVTQ=XVTQ			
Parameters Successfully Processed.....	15		
Parameters Bypassed Due to Error.....	0		
Parameters Read from PARMLIB Memory Object...	15		

**Figure 10** Xpediter/CICS Code Coverage Parameters Report Page 2

```

Compuware PARMLIB Report                               Page    2
Xpediter/CICS Code Coverage Parameters
APPLID: H01AC024 Date: 08 NOV 2017 Time: 16:48:39 USERID: ACMJETO TERM: 1718
Parameter Settings
-----
AUTO_SUBMIT_JCL_DURING_EXTRACTION=NO
CAPTURE_SYSTEM_FLOW_USING_GMT_TIME=YES
CODE_COVERAGE_PARMLIB_REPORT_DDNAME=XVGL
DATA_EXTRACTION_INTERVAL=42
INCREMENT_VERB_COUNTS=YES
KEEP_PROGRAM_COUNTS_FOR_USERID=*****
LOG_INFO_MESSAGES=NO LOG_TDO_NAME=CSMT
PRIME_OWNER_FILTER_WITH_CURRENT_USERID=NO
TRANSID_XVCC=XVCC
TRANSID_XVKP=XVKP
TRANSID_XVSC=XVSC
TRANSID_XVSI=XVSI
TRANSID_XVTQ=XVTQ
WRITE_DATA_EXTRACTION_MSGS_TO_JES_LOG=NO

```

- [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 XVTCSIT 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.

## Xpediter/CICS Code Coverage Parameter Log Output

Messages are written to the console log whenever the Xpediter/CICS Code Coverage 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 11](#) shows 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 11** Sample Xpediter/CICS Code Coverage Parameter Input, Console Log Output

```
08.06.15 J0436808 +XVTC5000I 07 JUN 2004 08:06:15 H01AC154 447
447 Xpediter/Code Coverage Global Override processing has begun.
08.06.15 J0436808 +XVTC5001I 07 JUN 2004 08:06:15 H01AC154 450
450 Xpediter/Code Coverage Global Override processing has ended.
```

In addition to the console log output above, the input processing program for the Xpediter/CICS Code Coverage Parameters process will display additional informational and error messages in the CICS CSMT output queue. [Figure 12](#) is a sample display of messages that can be output. In this sample, an error was detected. Member XVGBXX was not found in the Compuware PARMLIB.

**Figure 12** Sample Xpediter/CICS Code Coverage Parameter Input, CSMT Output

```
MXDIP0000I 29 APR 2016 10:25:12 H01AC024 Xpediter/CICS Input Provider DBUGINPT processing has begun.
MXDIP0026I 29 APR 2016 10:25:12 H01AC024 Input provider DBUGINPT is processing input for XVTC5001. Members are prefixed with "XVGB".
MXDIP0004I 29 APR 2016 10:25:12 H01AC024 Compuware Mainframe Services Controller problem processing member XVGBXX func=GET
MXDIP0011E 29 APR 2016 10:25:12 H01AC024 Member not found or no data returned. Check index member XD$$$00 for applid H01AC024 entries
MXDIP0001I 29 APR 2016 10:25:12 H01AC024 Xpediter/CICS Input Provider DBUGINPT processing has ended.
```

## Defining Output Report Parameters

This section describes the steps necessary to create parameters to override the parameter settings in the XVTCGBL load module. These parameters will be read during product initialization to allow modification of global table settings. This method eliminates the need to assemble and link-edit the global table XVTCGBL.

A standalone transaction, XVSI, can be used to refresh the parameters from the Compuware PARMLIB once Xpediter/Code Coverage has completed initialization. This eliminates the need to recycle Xpediter/Code Coverage when you want to make parameter changes. Even if you don't need to modify parameters at this time, Compuware suggests that you use the skeleton member created in this procedure. Nothing will be harmed, and if you later need to modify parameters, they will be ready for use.

Using the Compuware PARMLIB to specify parameters has several advantages. They include ease of installation, flexibility in maintenance of input, and the ability to modify the parameters while the region is active.

## Setting Up the Compuware PARMLIB Parameters

Perform the following steps to establish parameter processing.

1. Create the default member in the Compuware PARMLIB.
  - a. Create a new member, XVGB00, in the Compuware PARMLIB.
  - b. Copy member XVGB00 from the SMXDSAMP library into it.
  - c. Perform a refresh of this member. For more information, see the chapter entitled "Milestone 3: Compuware PARMLIB Implementation" in the *Xpediter/CICS Installation and Configuration Guide* and [Overriding Parameters](#) on page 47 in this manual for additional instructions.

This method of specifying parameters is very flexible. If you wanted to specify parameters for a specific CICS region, you could use the name XVGB $_{sysi}$  where  $_{sysi}$  is the SYSID of the CICS Region. You could also create a single member for all regions. By using member concatenation as described in the section entitled "Creating Additional Members for Region Customization" in the *Xpediter/CICS Advanced Configuration Guide*, you could create a set of members to control the process. You could have a member common to all regions, a member specifically for test regions, and a member for a single region.

For example, XVGB00 contains a set of parameters common to all regions. Member XVGBTEST could contain parameters intended for test regions. Finally, member XVGB0001 would have parameters specifically for region CICS0001.

2. Specify the name of the report transient data queue name.

Make sure the transient data queue (default name XVGR) has been installed. Refer to Step 1 in “Task 4.1.4 Compuware PARMLIB Reports” in the *Xpediter/CICS Installation and Configuration Guide*. If the name was changed, write it in the space below. Otherwise, just write in XVGR.

Report transient data queue name: \_\_\_\_\_

3. Modify the default override member XVGB00.

Add the CODE\_COVERAGE\_PARMLIB\_REPORT\_DDNAME parameters that allow you to easily override the report transient data queue name.

```
CODE_COVERAGE_PARMLIB_REPORT_DDNAME=name
```

Modify the lines as follows:

- a. Change name to the value you wrote down in [Step 2](#) on page 52.
- b. Save the member, refresh it as described in [PARMLIB Product Configuration](#) on page 52, and continue.

## PARMLIB Product Configuration

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

### Refreshing a Single Parameter Member

```
F cmsname,PARMLIB REFRESH member_name
```

### Refreshing All PARMLIB Members

```
F cmsname,PARMLIB REFRESH
```



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.

## Testing the Setup

Setup of the global table parameter override facility is now complete.

When testing Xpediter/Code Coverage after installation, look for the global table override report dataset in your JES output. Check the report, the console log, and the CICS and CSMT log output for any errors that may have occurred while overriding global table parameters. Message formats and explanations are provided in the *Xpediter/Code Coverage Mainframe User/Reference Guide* and the *Xpediter/CICS Messages and Codes* manual.

## Adding Parameters to a Member in the Compuware PARMLIB

Parameters used by your Compuware mainframe product or component are read from the Compuware PARMLIB dataset. Edit the sample parameters to your site's requirements, observing the recommendations in [Table 10](#).

**Table 10** General Parameter Editing Guidelines

PARM Name	Values	Samples
Data columns	1 to 71	N/A
Symbol separator	Underscore	"A_LONG_SYMBOL"
Continuation	Check for "+" character in Column 72.	+
Comments	"*" in column 1. Between columns 1-71, begin with "/*" and end with "*/"	* is a comment /* is a comment */



# Milestone 8: Verify the Xpediter/Code Coverage Installation

This section provides step-by-step instructions for the verification of the Xpediter/Code Coverage installation.



Roles involved:  
Xpediter/Code Coverage Installer.

## Task 8.1 Establish Batch COBOL Support



If your site will not be using Xpediter/Code Coverage for batch applications, skip ahead to [Establish Online IMS Support](#).

### Test Preparation

[Table 11](#) lists the sample programs used for verification of Xpediter/Code Coverage. Check the list for the programs you must compile for your environment. The programs are contained in the Xpediter/TSO samples dataset, SLXTSAMP. Use the COBOL Language Processor (LP) to compile the sample programs in the listed order. Some main routines statically link to subroutines higher in the list. Refer to the *Compuware Shared Services User/Reference Guide* for information on using the COBOL Language Processor. Optionally, you can use the online Compile Facility provided by Xpediter/TSO by selecting option 1 (PREPARE) from the Xpediter/TSO Primary Menu.

**Table 11** Xpediter/TSO COBOL Installation Verification Programs

Program Name	Required to Verify Support for:
TRITST	COBOL
TRIRPT	COBOL
TRIMAIN calls TRITST and TRIRPT	COBOL
TRITST2	COBOL
TRIRPT2	COBOL
TRIMAIN2 calls TRITST2 and TRIRPT2	COBOL

Note the following:

- Code Coverage requires specific compile options which may vary depending on the language type and actual version being used. Always refer to the appropriate *Compuware Shared Services User/Reference Guide* for information related to any required compiler options. An example follows:
  - Options NOTEST and NONUMBER are required for COBOL.

The *Compuware Shared Services User/Reference Guide* is required to assist users in manually converting Compile and Link JCL to execute the CSS Language Processor. The *Guide* also gives users information on how to set up the Language Processor parameters to write a matching source listing to a DDIO file.

- Most of the sample programs are coded to require **APOST** and **LANGLVL(2)**, although Xpediter/TSO fully supports both the **QUOTE** and the **LANGLVL(1)** compiler options for your programs.
- The **SYSLMOD** output from these compiles should not be directed to the Xpediter/TSO **LOADLIB** library, but to a dataset available to your application programmers for training sessions. For example, you could create a dataset named **CPWR.XT.TRAINLIB**.
- Make sure you exercise each of your modified **PROCs** at least once, even if this means compiling some of the sample programs more than once.

To verify that the compile was successful and that a source listing member was created, check the listing under the ddname **CWPERRM** for a message similar to the following:

```
LISTING PGMNAME DATED nn/nn/nn AT nn.nn.nn SUCCESSFULLY WRITTEN TO CWPDDIO
```

## Test Initiation and Execution

1. Select **2** (TSO) from the Xpediter/TSO Primary Menu.
2. Select **1** (STANDARD) from the Environments Menu.



The Environments Menu is automatically displayed if this is your first time to invoke a test session. Otherwise, you can access the Environments Menu by entering **SETUP** from the command line of the displayed test screen and selecting option **0** from the Test Setup Menu.

3. Enter **SETUP** from the Standard test screen.
4. Select **1** (LOADLIBS) from the Setup Menu.
5. On the Load Module Libraries screen, specify the application load library that contains the **TRIMAIN** or **TRIMAIN2** load module.
6. Press Enter.
7. Select **2** (DDIO) from the Setup Menu.
8. On the DDIO Files screen, specify the DDIO dataset name that contains the source listing member of **TRIMAIN** or **TRIMAIN2**.
9. Press Enter.
10. Select **C** (Code Coverage) from the Setup Menu.
11. On the Code Coverage setup screen, specify a user-defined System ID and Test ID (for the purpose of the IVP specify **SYSTEM1** and **TEST1** respectively).
12. Also enter the name of the Code Coverage repository created in [Create a Results Repository Dataset](#). Refer to [Figure 13](#).



**Figure 13** Entering Code Coverage Information

```

----- XPEDITER/TSO - CODE COVERAGE -----
COMMAND ==>

Repository DSNAME ==> 'COMPWARE.CC.REPOSIT'

    System Name ==> SYSTEM1                Test ID ==> TEST1

Test Data Optimization Dataset:      (DSNAME will be generated if blank)
    DSNAME ==>

    Data Class ==>
    Space Units ==> TRK
        Primary ==> 2
        Secondary ==> 2
    Storage Class ==>
        Unit ==>
        Volume ==>

Disposition Options:
    A (Append)
    D (Delete)
    K (Keep)
    ? (Prompt)

Disposition After the Test ==> K      (K or ?)

Disposition Before the Test ==> A      (A, D, or ? Only if DSN specified)

    Press ENTER to Process or Enter END Command to Terminate

```

13. Press Enter.
14. Enter END or press PF3 to return to the Standard screen.
15. Specify only the field values listed on the sample *Standard Test* screen illustrated in [Figure 14](#). Leave all other fields blank. Be sure to populate the Code Coverage Test? field with YES.



If the IVP program you are using is TRIMAIN2, use JCL member TRIJCL2 in the SLXTSAMP library.

**Figure 14** Standard Test Screen

```

Profile: XXX ----- XPEDITER/TSO - STANDARD (2.1) -----
COMMAND ==>

COMMANDS:  SSetup (Display Setup Menu)
           PProfile (Display Profile Selection)
TEST SELECTION CRITERIA:

    Program ==> TRIMAIN
    Entry Point ==>
    Load Module ==>

    Initial Script ==>
    Post Script ==>

    PARM ( Caps = YES ) ==>

    File List/JCL Member ==> 'COMPWARE.XT.SLXTSAMP(TRIJCL)'
    Preview Files? ==> NO
    Code Coverage Test? ==> YES (YES, NO, TDO)  SYSTEM FLOW? ==> NO
    Is This a DB2 Test? ==> NO    Plan ==>          System ==>

    Press ENTER to process or enter END command to terminate

```

16. Press Enter to begin the Xpediter/TSO debugging session.

The message area contains information on the files being allocated. On a cleared screen, a message indicating a test session is being entered appears.

17. Type **GO** on the command line and press Enter.

The execution pauses at the **GOBACK** statement. A message is reported in the area below the command line stating **TEST COMPLETED**.

18. Type **GO** and press Enter to return to the Standard screen.
19. Type **LOG** on the command line to enter the test log.
20. Type **M** and press PF8 to page to the bottom of the log.

A message will indicate that the Code Coverage statistics were successfully written to the repository.

21. Type **END** or press PF3 to exit the log.
22. Type **D** in the Process Option field to delete the log.

## Task 8.2 Establish Online IMS Support



If your site will not be using Xpediter/Code Coverage for online IMS applications, skip ahead to [Establish CICS Support](#).

### Test Initiation and Execution

1. Before verifying Xpediter/IMS, make sure you have included the following information in your IMS control blocks gen:
  - XPIMSDBT database
  - A message processing program PSB named TRIMPP
  - Transaction TRIMPP is associated with the above application
  - The message format control blocks.
2. Compile sample program TRIMPP as part of your system verification, using the appropriate options.

The Xpediter/IMS sample program TRIMPP is shipped in source form with in SLXTSAMP. This sample program must be compiled as a part of your system verification procedure.

Verify your Xpediter/IMS compile PROCs. If your site supports multiple types of compile PROCs, you must verify all of them using the sample programs.



Module TRIMPP must be link-edited with entry point DLITCBL.

Xpediter requires specific compile options which may vary depending on the language type and actual version being used. A user should always refer to the appropriate *Compuware Shared Services User/Reference Guide* for information related to any required compiler options. For example: (1) Options NOTEST and NONUMBER are required for COBOL. The guide is required to assist users in manually converting Compile and Link JCL to execute the CSS Language Processor. The guide also gives users information on how to set up the Language Processor parameters to write a matching source listing to a source listing file.

Compile the sample programs with COBOL. To verify that the compile was successful and that a source listing was created, check the listing under the ddname CWPERRM for a message similar to the following:

LISTING PGMNAME DATED nn/nn/nn AT nn.nn.nn SUCCESSFULLY WRITTEN TO CWPDDIO

3. Once you have a valid load module and source listing dataset, log on to TSO and invoke the Primary Menu. Use whatever procedures you selected while installing this Primary Menu.
  - Select **2** (TSO) to display the Environments Menu.



The Environments Menu is automatically displayed if this is your first time to invoke a test session. Otherwise, you can access the Environments Menu by entering **SETUP** from the command line of the displayed test screen and selecting option **0** from the Test Setup Menu.

- Select either **8** (MPP) or **9** (BMP/IFP) from the Environments Menu.
- From the test screen, enter the command **SETUP** to display the Setup Menu.
- Select **A** (All) to review all your setup selections.
  - Load libraries—the user program libraries allocated as STEPLIB. Verify that the order of concatenation is correct.
  - DDIO dataset—the library name should be the dataset name of the DD CWPDDIO in your Xpediter compile step.
  - Test script libraries.
  - Test session log dataset.
  - Test session script dataset.
  - DB2 system names and DSNLOAD datasets.
  - PANEXEC defaults (if installed).
  - PSB and DBD libraries (GSAM only).
  - IMS preload list.
  - IMS authorized load libraries.
  - IMS region ID and PARM strings—PARM fields set up at install time can be changed.
- Press **PF3** from the Setup Menu to return to the test screen.
- Select **C** (Code Coverage) from the Setup Menu

The Xpediter/IMS sample source program TRIMPP is found in the SLXTSAMP dataset. TRIMPP is the driver module. The program issues a GU call to the message queue and reads in the three sides of a triangle that were entered from the IMS terminal through the format TRIMPP. The program then issues a GU call to the database to check whether this particular configuration of values has been entered before. If a matching record is found, the program retrieves the type of triangle data and sends the response back to the IMS terminal. If not, the program calls the subprogram TRITST to determine the type of triangle, stores the information in the database for future reference, and sends the response back to the IMS terminal.

4. On the Code Coverage setup screen, specify a user-defined System ID and Test ID (for the purpose of the IVP specify SYSTEM1 and TEST1 respectively).
5. Also enter the name of the Code Coverage repository created in [Create a Results Repository Dataset](#). Refer to [Figure 15](#).

**Figure 15** Entering Code Coverage Information

```

----- XPEDITER/TSO - CODE COVERAGE -----
COMMAND ==>

Repository DSNAME ==> 'COMPWARE.CC.REPOSIT'

    System Name ==> SYSTEM1                Test ID ==> TEST1

Test Data Optimization Dataset:      (DSNAME will be generated if blank)
    DSNAME ==>

    Data Class ==>
    Space Units ==> TRK
        Primary ==> 2
        Secondary ==> 2
    Storage Class ==>
        Unit ==>
        Volume ==>

Disposition Options:
    A (Append)
    D (Delete)
    K (Keep)
    ? (Prompt)

Disposition After the Test ==> K      (K or ?)

Disposition Before the Test ==> A      (A, D, or ? Only if DSN specified)

    Press ENTER to Process or Enter END Command to Terminate

```

6. Press **Enter**.

7. Type **END** or press **PF3** from the Setup Menu to return to the test screen.

8. Select **2** (TSO) from the Xpediter/TSO Primary Menu. This invokes the Environments Menu from which you select **8** (MPP).



The Environments Menu is automatically displayed if this is your first time to invoke a test session. Otherwise, you can access the Environments Menu by entering **SETUP** from the command line of the displayed test screen and selecting option **0** from the Test Setup Menu.

9. On the MPP Test screen illustrated in [Figure 16](#), type **TRIMPP** as the PROGRAM to be verified, or **TRIMPP** as the TRANCODE, or both and press Enter. Be sure to populate the Code Coverage Test? field with YES.

**Figure 16** MPP Test Screen

```

Profile: XXX ----- XPEDITER/TSO - MPP (2.8) -----
COMMAND ==>

COMMANDS:  SSetup (Display Setup Menu)
           INter (Display Intercepts)
           PROfile (Display Profile List)           DDown (Scroll Down)

                                     INTERCEPTS                               Row 1 of 3

PROGRAM  TRANCODE  INITSCR  POSTSCR  START  MAX  -----  DATA  -----
> TRIMPP
>
>

    IMS USERID ==> PFHABCO                Retain Breakpoints? ==> NO

                                     NBA ==> 2                (Normal Buffer Allocation)
                                     OBA ==> 3                (Overflow Buffer Allocation)

File List/JCL Member ==>
Unattended ==> NO
Code Coverage Test? ==> YES (YES, NO, TDO)    System Flow? ==> NO
Is This a DB2 Test? ==> NO                    System ==>
    Press ENTER to process or enter END command to terminate

```

10. If the intercepts are set, the following messages appear on the screen:

```
*** THE IMS INTERCEPTS ARE BEING SET ***
*** INTERCEPTS SET - STARTING THE IMS xxx_REGION ***
*** THE TEST TRANSACTION CAN BE ENTERED ***
```

The terminal is then locked. If an error is encountered, however, the intercept is not set and an appropriate error message appears on the message line or in the log.



Xpediter automatically starts the message region. You do not have to submit any MSG region JCL for testing under Xpediter.

11. Go to an IMS terminal connected to an associated IMS/DC control region and type **/FOR TRIMPP**. Press **Enter**.
12. On the formatted data entry screen, type the values **9 8 7** for the three sides of a triangle and press **Enter**.
13. Go to the TSO terminal and wait for the source display of program TRIMPP.  
The symbolic representation of the source code is displayed, beginning with the Procedure Division. The third header line contains the program name of the module you are currently in; i.e., PROGRAM: TRIMPP. The arrow pointing to the ENTRY 'DLITCBL' USING IOPCB, DBPCB statement indicates that execution starts at this entry point. The execution arrow also corresponds to what is in the execution status message area; i.e., Before TRIMPP. The Keep window automatically displays the value of the data items on the current line.
14. The B in column 9 of the ENTRY 'DLITCBL' line is set automatically by Xpediter/IMS. Press **PF8** to scroll down. The A in column 9 of statement number 149 (GOBACK) is also set by Xpediter/IMS. Press **PF6** or enter **LOCATE \*** to return to the execution arrow.
15. Type **GO** on the command line and press **Enter**. The execution pauses at the GOBACK statement. A message is reported in the area below the command line stating **TEST COMPLETED**.
16. Type **EXIT** or press **PF4** to return to the Standard screen.
17. Type **LOG** on the command line to enter the test log.
18. Type **M** and press **PF8** to page to the bottom of the log. At this point, there will be message indicating the Code Coverage statistics were successfully written to the repository.
19. Type **END** or press **PF3** to exit the log.
20. Type **D** in the Process Option field to delete the log.

### Task 8.3 Establish CICS Support



If your site will not be using Xpediter/Code Coverage for CICS applications, skip ahead to [Set Up Code Coverage Reporting](#) on page 64.

If your site will not be using Xpediter/Code Coverage for CICS applications, this section may be skipped.

### Test Preparation

The test programs are used to verify proper installation and to demonstrate the product. [Table 12](#) shows the program name for each test program and the language that it is used for. The table also includes the name of the Assembler subroutine that is called by this program.

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

**Table 12** Test Programs

Program Name	Language
CWDEMCB2	COBOL
CWCDSUBA	Assembler Subroutine



If you have previously compiled the test programs as part of the Xpediter/CICS installation, you do not need to recompile them.

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 is link-edited.
2. 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.

## Populate Program Inventory

1. Edit the member JCLCPOP in the Xpediter/CICS SMXDSAMP dataset.
2. Add a valid jobcard.
3. Modify the steplib to contain the Xpediter/CICS and Compuware Shared Services libraries.
4. Modify the DFHRPL DD to contain the load library that contains CWDEMCB2.
5. Modify the SLSF001 DD to contain the Source Listing Library that contains the source listing for CWDEMCB2.
6. Modify the XVTPGINV DD to contain the program inventory defined to the CICS region.
7. Modify the EXCLUDES DD to point to the excludes member (POPEXCL) in the Xpediter/CICS SMXDSAMP dataset.
8. Add the following statements to the CONTROL DD to populate the program inventory file with the new copy of the program, delete all old copies of the program, and set the program for collection by Xpediter/Code Coverage:

```
REPLACE ALL
COLLECT ON
DEFINE CWDEMCB2 *
```

9. Submit the JCL.

## Test Definition and Execution

1. On a blank CICS screen type **XPED** and press Enter. This activates Xpediter/CICS and presents the Xpediter/CICS Primary Menu.
2. Type **C** (Code Coverage) on the command line.
3. Press Enter
4. Type **1** to select the Test Definition screen.
5. Press Enter.
6. Specify a user-defined System Name and Test Identifier (for the purpose of the IVP specify SYSTEM1 and TEST1 respectively). Refer to [Figure 17](#).

**Figure 17** Entering Code Coverage Information

```

----- XPEDITER/CODE COVERAGE - Test Definition (1) -----C123
COMMAND ==>
Userid: ACMJETO
Entry 1 of 1
Available Commands are highlighted:
(S) Specify Collection (H) Halt Collection (V) View Collection Activity
(SC) Start Code Coverage (SF) Start System Flow (SB) Start Both
(E) Extract Entry (ES) Extract System Next Full Extract: 08:42:36
(D) Delete Entry (DS) Delete System Current Time: 08:27:31
(B) Browse (A) Archive
CMD System Name Test Identifier Owner Status Code System
Coverage Flow
-----
_ SYSTEM1 TEST1 ACMJETO Incomplete
-----
PF1 Help PF3 End Extract All (Y/N): N

```

7. Press Enter.
8. Type **S** in the line command field next to your newly created system.
9. Press Enter.
10. On the Collection Specification screen:
  - a. Type your userID in the first User column.
  - b. Type **XCB2** in the Tran field.
  - c. Type **CWDEMCB2** in the program field.
  - d. Type an asterisk (\*) in the CSECT field.
11. Press Enter.
12. Press PF4 to validate the wildcarded collection specification.
13. Verify the collection specifications. The Collection Validation screen identifies the two entries shown in [Figure 18](#). The Load Library (Lib), Program Inventory (INV) and Member Listing indicators should be set to 'Y', and the Collection Status set to ON.

**Figure 18** Using the Collection Validation Screen to Validate Specifications

```

----- XPEDITER/CODE COVERAGE - Collection Validation (1) -----C123
COMMAND ==>                                     SCROLL ==> CSR
Userid: ACME123 ***** XVT0136I Bottom of data reached *****

System Name:  SYSTEM1          TestID: TEST1          Owner: ACME123
Program/CSECT: CWDEMCB2/*****
Collection Matches: 00002 of 00002          Validating 01 of 01 masks

----- Program -----   ----- Compile -----   --- Source ---   Coll
Module  Lib Csect   Inv    Date      Time      Member  Listing  Status
-----
CWDEMCB2 Y CWCDSUBA Y    2003/07/17 16:24:00  CWCDSUBA Y   ON  <----
CWDEMCB2 Y CWDEMCB2 Y    2003/07/17 15:52:15  CWDEMCB2 Y   ON  <----
**END**

PF1 Help      PF3 End

```

14. Type **END** or press PF3 to return to the Test Definition/Collection Specification screen.
15. Type **END** or press PF3 to return to the Test Definition Update screen.
16. Type **SC** in the line command field next to your newly created system to start Code Coverage collection.
17. Press Clear to return to CICS so that you can begin your test.
18. On a blank CICS screen, type **XCB2** and press Enter. The XCB2 Demonstration Transaction screen is displayed.
19. To allow the transaction to end normally, type **00999** and press Enter.
20. Press Clear to return to CICS.
21. Type **XPED** and press Enter.
22. Type **C** (Code Coverage) on the command line and press Enter.
23. Type **1** to return to the Test Definition screen.
24. Type **H** in the line command field next to your newly created system to write the Code Coverage statistics to the repository.

## Task 8.4 Set Up Code Coverage Reporting

1. From the Code Coverage Primary Menu, type **0** to enter the Code Coverage Defaults Menu.
2. Type **2** (Repository) and press Enter.
3. Enter the name of the repository that is defined in the CICS extraction JCL.
4. Press Enter.
5. Type **END** or press PF3 to return to the Defaults Menu.
6. Type **3** (Source Listing File) and press Enter.
7. Enter the name of the Source Listing File that contains the listing of CWDEMCB2.
8. Press Enter.
9. Type **END;END** or press PF3 twice to return to the Code Coverage Primary Menu.
10. Type **1** to enter the Reports screen.



11. Enter the name of your system under the System Name column and your test id under the Test ID column (for the purpose of the IVP it should be SYSTEM1 and TEST1).
12. Type **S** on the line command field next to your system name to execute the JCL that creates the Code Coverage reports.
13. Review the reports in the output of the job on the held cue. There should be execution statistics for each program that was executed with Code Coverage.



# Milestone 9: Configure Topaz Workbench Integration

This milestone will guide you through the tasks necessary to integrate Xpediter/Code Coverage with Topaz Workbench.



If your site is not using—and does not plan to use—Topaz Workbench, skip ahead to ["Milestone 10: Troubleshooting"](#).



Roles involved:  
Enterprise Common Components (ECC) Installer  
Topaz Workbench Installer  
Xpediter/Code Coverage Installer.



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

Complete the following tasks to configure Xpediter/Code Coverage integration with Topaz Workbench.

## Task 9.1 Configure the CXSSAS Task

The following people are required for this task:

- Xpediter/Code Coverage Installer
- ECC Installer.

To configure Topaz Workbench integration, follow the instructions the *Enterprise Common Components Installation and Configuration Guide* “Milestone 8: Configure Topaz Workbench Integration”. You must modify the SSAS PROC (job CXSSAS) to uncomment the Xpediter/Code Coverage library (SLXVLOAD) in the STEPLIB concatenation and update it with the fully-qualified name.

## Task 9.2 Verify the Topaz Workbench Integration

The following person is required for this task:

- Topaz Workbench Installer.

Contact your Topaz Workbench Installer to verify Topaz Workbench integration with Xpediter/Code Coverage via the Topaz Workbench Code Coverage/Eclipse feature.



# Milestone 10: Troubleshooting

## Customer Support

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.

## Information for Customer Solutions

If problems arise, please check your manual for assistance. If problems persist, please obtain the following information before calling Compuware for assistance. This information will help determine the exact cause of the problem as quickly as possible.

1. Identify the release number of Compuware product(s) in use.
2. Identify the operating system.
3. Identify the release of CICS Transaction Server that is being used.
4. If an abend occurs, note the displacement and the module in which it occurs. If possible, obtain a copy of the system dump.
5. Note the sequence of steps (including all commands issued) that resulted in the problem. Also note any variable data types and programming languages involved.
6. To receive product fixes electronically, be ready to provide your email address.



# Checklist of Milestones and Tasks

- ❑ Milestone 1: Ensure Installation and Configuration of Companion Products
  - ❑ Task 1.1 Install/Upgrade Xpediter/TSO and Xpediter/IMS
  - ❑ Task 1.2 Apply Xpediter/TSO and Xpediter/IMS Maintenance
  - ❑ Task 1.3 Install/Upgrade Xpediter/CICS
  - ❑ Task 1.4 Apply Xpediter/CICS Maintenance
  - ❑ Task 1.5 Import Xpediter/Code Coverage License
- ❑ Milestone 2: Install Xpediter/Code Coverage Using SMP/E
  - ❑ Task 2.1 Ensure Product Integrity
  - ❑ Task 2.2 Follow the Compuware Installation Guide
- ❑ Milestone 3: Configuration Preparation
  - ❑ Task 3.1 Provide Users Access to Product Runtime Libraries and Files
  - ❑ Task 3.2 Allow Access for the Xpediter/Code Coverage Installer
- ❑ Milestone 4: Configure Xpediter/Code Coverage — New Installation
  - ❑ Task 4.1 Install ISPF Support
  - ❑ Task 4.2 Create a Results Repository Dataset
  - ❑ Task 4.3 Global Filter File Implementation
  - ❑ Task 4.4 Limit Record Deletion Capabilities
- ❑ Milestone 5: Configure Xpediter/Code Coverage — Upgrade
  - ❑ Task 5.1 Update ISPF Support
    - ❑ Task 5.1.1 ISPF Support
  - ❑ Task 5.2 Upgrade Xpediter/Code Coverage with Xpediter/CICS
    - ❑ Task 5.2.1 Modify JCL Template Dataset for CICS
    - ❑ Task 5.2.2 Update the CICS Startup JCL
    - ❑ Task 5.2.3 Convert Global Table Override Input

- ❑ **Milestone 6: Configure CICS for Use with Xpediter/Code Coverage**
  - ❑ **Task 6.1 Create and Format the Program Inventory for CICS**
    - ❑ **Task 6.1.1 Determine Appropriate Size for the Program Inventory Dataset**
    - ❑ **Task 6.1.2 Create a Program Inventory Dataset**
  - ❑ **Task 6.2 Create an Extract Dataset for CICS**
  - ❑ **Task 6.3 Create JCL Template Dataset for CICS**
  - ❑ **Task 6.4 Create the XVTCSPEC Dataset for CICS**
  - ❑ **Task 6.5 Update the CICS Resource Definitions**
  - ❑ **Task 6.6 Update the CICS Startup JCL**
  - ❑ **Task 6.7 Review Global Parameters**
- ❑ **Milestone 7: Using Compuware PARMLIB to Configure Xpediter/Code Coverage with Xpediter/CICS**
- ❑ **Milestone 8: Verify the Xpediter/Code Coverage Installation**
  - ❑ **Task 8.1 Establish Batch COBOL Support**
  - ❑ **Task 8.2 Establish Online IMS Support**
  - ❑ **Task 8.3 Establish CICS Support**
  - ❑ **Task 8.4 Set Up Code Coverage Reporting**
- ❑ **Milestone 9: Configure Topaz Workbench Integration**
  - ❑ **Task 9.1 Configure the CXSSAS Task**
  - ❑ **Task 9.2 Verify the Topaz Workbench Integration**
- ❑ **Milestone 10: Troubleshooting**