

ASG-Encore Installation Guide

Version: 6.0

Publication Number: ENX0300-60

Publication Date: February 2002

The information contained herein is the confidential and proprietary information of Allen Systems Group, Inc. Unauthorized use of this information and disclosure to third parties is expressly prohibited. This technical publication may not be reproduced in whole or in part, by any means, without the express written consent of Allen Systems Group, Inc.

© 1993 - 2002 Allen Systems Group, Inc. All rights reserved.

All names and products contained herein are the trademarks or registered trademarks of their respective holders.



ASG Worldwide Headquarters Naples, Florida USA | asg.com

1333 Third Avenue South, Naples, Florida 34102 USA Tel: 941.435.2200 Fax: 941.263.3692 Toll Free: 1.800.932.5536

ASG Support Numbers

ASG provides support throughout the world to resolve questions or problems regarding installation, operation, or use of our products. We provide all levels of support during normal business hours and emergency support during non-business hours. To expedite response time, please follow these procedures.

Please have this information ready:

- Product name, version number, and release number
- List of any fixes currently applied
- Any alphanumeric error codes or messages written precisely or displayed
- A description of the specific steps that immediately preceded the problem
- The severity code (ASG Support uses an escalated severity system to prioritize service to our clients. The severity codes and their meanings are listed below.)
- Verify whether you received an ASG Service Pack for this product. It may include information to help you resolve questions regarding installation of this ASG product. The Service Pack instructions are in a text file on the distribution media included with the Service Pack.

If You Receive a Voice Mail Message:

- 1 Follow the instructions to report a production-down or critical problem.
- 2 Leave a detailed message including your name and phone number. A Support representative will be paged and will return your call as soon as possible.
- 3 Please have the information described above ready for when you are contacted by the Support representative.

Severity Codes and Expected Support Response Times

Severity	Meaning	Expected Support Response Time
1	Production down, critical situation	Within 30 minutes
2	Major component of product disabled	Within 2 hours
3	Problem with the product, but customer has work-around solution	Within 4 hours
4	"How-to" questions and enhancement requests	Within 4 hours

ASG provides software products that run in a number of third-party vendor environments. Support for all non-ASG products is the responsibility of the respective vendor. In the event a vendor discontinues support for a hardware and/or software product, ASG cannot be held responsible for problems arising from the use of that unsupported version.

Business Hours Support

Your Location	Phone	Fax	E-mail
United States and Canada	800.354.3578	941.263.2883	support@asg.com
Australia	61.2.9460.0411	61.2.9460.0280	support.au@asg.com
England	44.1727.736305	44.1727.812018	support.uk@asg.com
France	33.141.028590	33.141.028589	support.fr@asg.com
Germany	49.89.45716.300	49.89.45716.400	support.de@asg.com
Singapore	65.332.2922	65.337.7228	support.sg@asg.com
All other countries:	1.941.435.2200		support@asg.com

Non-Business Hours - Emergency Support

Your Location	Phone	Your Location	Phone
United States and Canada	800.354.3578		
Asia	65.332.2922	Japan/Telecom	0041.800.9932.5536
Australia	0011.800.9932.5536	Netherlands	00.800.3354.3578
Denmark	00.800.9932.5536	New Zealand	00.800.9932.5536
France	00.800.3354.3578	Singapore	001.800.3354.3578
Germany	00.800.3354.3578	South Korea	001.800.9932.5536
Hong Kong	001.800.9932.5536	Sweden/Telia	009.800.9932.5536
Ireland	00.800.9932.5536	Switzerland	00.800.9932.5536
Israel/Bezeq	014.800.9932.5536	Thailand	001.800.9932.5536
Japan/IDC	0061.800.9932.5536	United Kingdom	00.800.3354.3578
		All other countries	1.941.435.2200

ASG Web Site

Visit <http://www.asg.com>, ASG's World Wide Web site.

Submit all product and documentation suggestions to ASG's product management team at <http://www.asg.com/asp/emailproductsuggestions.asp>.

If you do not have access to the web, FAX your suggestions to product management at (941) 263-3692. Please include your name, company, work phone, e-mail ID, and the name of the ASG product you are using. For documentation suggestions include the publication number located on the publication's front cover.

Contents

Prefaceiii
About this Publicationiii
Related Publicationsiv
ASG-Existing Systems Workbench (ASG-ESW)	v
Invoking ESW Productsviii
ESW Product Integrationix
Examples	x
Publication Conventions	xii
1 Introduction	1
ASG Service Pack	1
Encore Overview	2
Encore Components	2
Operating Environment	5
Environment	5
COBOL Compilers	5
Preprocessor Support	6
2 Customization	7
Prerequisite	7
Step 1 - Modifying CNTL Library Members	8
Step 2 - Adding Encore Modules to MLPA/PLPA	10
Step 3 - Customizing Panvalet or Librarian	11
Panvalet Customization	11
Librarian Customization	12
Step 4 - Customizing the Module Header Box	13
Step 5 - Setting up the Encore/COBOL/SF Interface	13

Step 6 - Invoking Encore	14
To invoke Encore from an ISPF panel	14
To invoke Encore using a CLIST	14
Step 7 - Validating Encore	15
Glossary	61
Appendix A	
Encore CNTL and CLIST Members	75
Encore CNTL Members	75
Encore CLIST Members	76
Index	77

Preface

This *ASG-Encore Installation Guide* explains the installation and maintenance of ASG-Encore (herein called Encore). Encore is a powerful interactive system that helps analysts and programmers to re-engineer, redevelop, or maintain existing COBOL systems.

Allen Systems Group, Inc. (ASG) provides professional support to resolve any questions or concerns regarding the installation or use of any ASG product. Telephone technical support is available around the world, 24 hours a day, 7 days a week.

ASG welcomes your comments, as a preferred or prospective customer, on this publication or on any ASG product.

About this Publication

This publication consists of these chapters:

- [Chapter 1, "Introduction,"](#) contains an overview of product installation, re-engineering analysis, code extraction analysis, target generation, and operating environments.
- [Chapter 2, "Customization,"](#) describes the customization of specific Encore components, invoking Encore, and validating the Encore installation.

Related Publications

The documentation library for ASG-Encore consists of these publications (where *nn* represents the product version number):

- *ASG-Center Installation Guide* (CNX0300-*nn*) contains installation and maintenance information for ASG-Center, the common set of libraries shared by most ASG-Existing Systems Workbench mainframe products. ASG-Center must be installed before installing ASG-Encore.
- *ASG-Encore Installation Guide* (ENX0300-*nn*) explains the installation and maintenance of ASG-Encore.
- *ASG-Encore Reference Guide* (ENX0400-*nn*) provides detailed information about the pop-ups, screens, and commands used in ASG-Encore.
- *ASG-Encore User's Guide* (ENX0200-*nn*) provides user information about ASG-Encore.

Note: _____

To obtain a specific version of a publication, contact the ASG Service Desk.

This table contains the name and description of each ESW component:

ESW Product	Herein Called	Description
ASG-Alliance	Alliance	The application understanding component that is used by IT professionals to conduct an analysis of every application in their environment. Alliance supports the analysis and assessment of the impact of change requests upon an entire application. Alliance allows the programmer/analyst to accurately perform application analysis tasks in a fraction of the time it would take to perform these tasks without an automated analysis tool. The impact analysis from Alliance provides application management with additional information for use in determining the resources required for application changes.
ASG-AutoChange	AutoChange	The COBOL code change tool that makes conversion teams more productive by enabling quick and safe changes to be made to large quantities of code. AutoChange is an interactive tool that guides the user through the process of making source code changes.
ASG-Bridge	Bridge	The bridging product that enables field expansion for program source code, without being required to simultaneously expand the fields in files or databases. Because programs are converted in smaller groups, or on a one-by-one basis, and do not require file conversion, testing during the conversion process is simpler and more thorough.
ASG-Center	Center	The common platform for all ESW products. Center provides the common Analytical Engine to analyze the source program and store this information in the AKR. This common platform provides a homogeneous environment for all ESW products to work synergistically.

ESW Product	Herein Called	Description
ASG-Encore	Encore	The program re-engineering component for COBOL programs. Encore includes analysis facilities and allows you to extract code based on the most frequently used re-engineering criteria. The code generation facilities allow you to use the results of the extract to generate a standalone program, a callable module, a complement module, and a CICS server. Prior to code generation, you can view and modify the extracted Logic Segment using the COBOL editor.
ASG-Estimate	Estimate	The resource estimation tool that enables the user to define the scope, determine the impact, and estimate the cost of code conversion for COBOL, Assembler, and PL/I programs. Estimate locates selected data items across an application and determines how they are used (moves, arithmetic operations, and compares). Time and cost factors are applied to these counts, generating cost and personnel resource estimates.
ASG-Insight	Insight	The program understanding component for COBOL programs. Insight allows programmers to expose program structure, identify data flow, find program anomalies, and trace logic paths. It also has automated procedures to assist in debugging program abends, changing a computation, and resolving incorrect program output values.
ASG-Recap	Recap	The portfolio analysis component that evaluates COBOL applications. Recap reports provide function point analysis and metrics information, program quality assessments, intra-application and inter-application comparisons and summaries, and historical reporting of function point and metrics information. The portfolio analysis information can also be viewed interactively or exported to a database, spreadsheet, or graphics package.
ASG-SmartDoc	SmartDoc	The program documentation component for COBOL programs. SmartDoc reports contain control and data flow information, an annotated source listing, structure charts, program summary reports, exception reports for program anomalies, and software metrics.

ESW Product	Herein Called	Description
ASG-SmartEdit	SmartEdit	The COBOL editing component that can be activated automatically when the ISPF/PDF Editor is invoked. SmartEdit provides comprehensive searching, inline copybook display, and syntax checking. SmartEdit allows you to include an additional preprocessor (for example, the APS generator) during syntax checking. SmartEdit supports all versions of IBM COBOL, CICS, SQL, and CA-IDMS.
ASG-SmartTest	SmartTest	The testing/debugging component for COBOL, PL/I, Assembler, and APS programs in the TSO, MVS Batch, CICS (including file services), and IMS environments. SmartTest features include program analysis commands, execution control, intelligent breakpoints, test coverage, pseudo code with COBOL source update, batch connect, disassembled object code support, and full screen memory display.

Invoking ESW Products

The method you use to invoke an ESW product depends on your system setup. If you need assistance to activate a product, see your systems administrator. If your site starts a product directly, use the ISPF selection or CLIST as indicated by your systems administrator. If your site uses the ESW screen to start a product, initiate the ESW screen using the ISPF selection or CLIST as indicated by your systems administrator and then typing in the product command on the command line.

The product names can also vary depending on whether you access a product directly or through ESW. See ["ESW Product Integration" on page ix](#) for more information about using ESW.

To initialize ESW products from the main ESW screen, select the appropriate option on the action bar pull-downs or type the product shortcut on the command line.

Product Name	Shortcut	ESW Pull-down Options
Alliance	AL	Understand ▶ Application
AutoChange	CC	Change ▶ Conversion Set
Bridge	BR	Change ▶ ASG-Bridge
Encore (Re-engineer)	EN	Re-engineer ▶ Program
Estimate	ES	Measure ▶ ASG-Estimate
Insight (Understand)	IN	Understand ▶ Program
Recap (Portfolio Analysis)	RC	Measure ▶ Portfolio
SmartDoc (Document)	DC	Document ▶ Program
SmartEdit	SE	Change ▶ Program Or Change ▶ Program with Options
SmartTest	ST	Test ▶ Module/Transaction

ESW Product Integration

Because ESW is an integrated suite of products, you are able to access individual ESW products directly or through the main ESW screen. As a result, you might see different fields, values, action bar options, and pull-down options on a screen or pop-up depending on how you accessed the screen or pop-up.

Certain ESW products also contain functionality that interfaces with other ESW products. Using SmartTest as an example, if Alliance is installed, SmartTest provides a dynamic link to Alliance that can be used to display program analysis information. If Insight is installed and specified during the analyze, the Insight program analysis functions are automatically available for viewing logic/data relationships and execution path. For example, the Scratchpad option is available on the Options pull-down if you have Insight installed. Access to these integrated products requires only that they be installed and executed in the same libraries.

Example 2. [Figure 4](#) shows the File - Analyze Submit pop-up that displays when you access SmartTest directly. [Figure 5](#) shows the File - Analyze Submit pop-up that displays when you access SmartTest through ESW.

Notice that the Analyze features field in [Figure 5](#) lists additional ESW products than shown on [Figure 4](#). This field is automatically customized to contain the ESW products you have installed on your system.

The actions shown on these screens also vary. For example, the D action (ASG-SmartDoc Options) is available on the File - Analyze Submit screen if the SmartDoc product is installed on your system. In [Figure 4](#), the ASG-SmartDoc Options action is not available.

Figure 4 • File - Analyze Submit Screen

```

                                File - Analyze Submit
Command ==> -----
                E - Edit JCL                      S - Submit JCL

Compile and link JCL (PDS or sequential):
  Data set name 'USER12.REL.CNTL(UIAPCOBC)'

Analyze features (Y/N):
  ASG-SmartTest: Y   Extended Analysis: N

AKR data set name 'USER12.GENERAL.AKR'
AKR program name _____ (if overriding PROGRAM-ID)

Analyze options:
-----
-----

Compile? (Y/N) . . . . . Y   (Y if needed by features)
Link load module reusable? (Y/N) Y
  
```

Figure 5 • File - Analyze Submit Screen (Accessed through ESW)

```

                                File - Analyze Submit
Command ==> -----
                E - Edit JCL   S - Submit JCL   D - ASG-SmartDoc Options

Compile and link JCL (PDS or sequential):
  Data set name 'USER12.REL.CNTL(HTEST)'

Analyze features (Y/N):
  ASG-Insight: Y   ASG-SmartTest: Y   Extended Analysis: N
  ASG-SmartDoc: N   ASG-Encore: N

AKR data set name 'USER12.GENERAL.AKR'
AKR program name _____ (if overriding PROGRAM-ID)

Analyze options:
-----
-----

Compile? (Y/N) . . . . . Y   (Y if needed by features)
Link load module reusable? (Y/N) Y   (ASG-SmartTest)
  
```

Publication Conventions

ASG uses these conventions in technical publications:

Convention	Represents
ALL CAPITALS	Directory, path, file, dataset, member, database, program, command, and parameter names.
Initial Capitals on Each Word	Window, field, field group, check box, button, panel (or screen), option names, and names of keys. A plus sign (+) is inserted for key combinations (e.g., Alt+Tab).
<i>lowercase italic monospace</i>	Information that you provide according to your particular situation. For example, you would replace <i>filename</i> with the actual name of the file.
Monospace	Characters you must type exactly as they are shown. Code, JCL, file listings, or command/statement syntax. Also used for denoting brief examples in a paragraph.
Vertical Separator Bar () with underline	Options available with the default value underlined (e.g., Y <u>N</u>).

1

Introduction

This chapter describes Encore product installation, re-engineering analysis, code extraction analysis, target generation, and operating environments, and contains these sections:

Section	Page
ASG Service Pack	1
Encore Overview	2
Operating Environment	5

Note: _____

Center installation and customization must be performed before installing Encore. If Center has not been installed, see the *ASG-Center Installation Guide* at this time.

ASG Service Pack

Verify whether you received an ASG Service Pack for this product. If so, read the instructions for installing the Service Pack before proceeding with the product installation. The installation instructions are located in the text file on the distribution media included with the Service Pack. If you have any problems with the Service Pack, contact the ASG Service Desk.

Encore Overview

Encore is the ESW re-engineering product for COBOL applications and includes these features:

- Analysis facilities that allow you to extract code based on the most frequently used re-engineering criteria.
- Code generation facilities that allow you to use the results of the analysis extract to generate a standalone program, a callable module, and a complement module. Prior to code generation, you can view and modify the extracted logic segment using the COBOL editor.

Encore is designed to support these functions:

- Web enablement
- Open interface
- Platform and technology migration
- Re-implementation of business functions
- Component-based maintenance

Encore contains a Common User Access (CUA) interface with action bars, pull-down menus, and pop-up screens that make it easy to learn and use.

Encore Components

Encore is comprised of these four functional components:

- Re-engineering analysis
- Code extraction
- Migration
- Target generation

Re-engineering Analysis Overview

Encore's re-engineering analysis displays the source code in various levels. The re-engineering analysis component consists of these three displays:

View Methods	Contents of Views
Structure	Displays the hierarchical relationships within the program in a graphical mode. You can zoom in on the actual COBOL source represented by the graphical display.
Tree	Displays the program in logical execution order. This view allows straightforward analysis of how the program works.
Source	Displays the program in source code order and offers COBOL-intelligent browsing capabilities.

The main purpose of re-engineering analysis is to gather enough information about the program to make the best decision regarding code isolation and extraction.

Re-engineering analysis is available through the View facility.

Extract Overview

Encore's main focus is the Extract facility, which addresses the re-engineering aspects of code renewal, and provides the capability to isolate and extract logical code segments from existing programs. These are the eight methods you can use to isolate and extract code:

Objective Type	Purpose
Perform Range	Isolates and extracts a specific named PERFORMed range of code within the COBOL program.
Transaction	Isolates and extracts all code required for a specific transaction (i.e., all code between a start point and multiple end points that is, or could be, executed based on the values of a data variable).
Computation Variable	Isolates and extracts all code that calculates a specific data variable from the beginning of the program to a specific statement in the program.
Report	Isolates and extracts all code necessary to produce a set of identified output (WRITE) statements.
Statement	Isolates and extracts a user-selected set of COBOL source statements.
Complement	Extracts the original program, excluding statements from a previous extract.

Objective Type	Purpose
Common Code	Identifies code common to two or more previous extracts.
CICS Server	Isolates and extracts all code required to generate a CICS server program.

See the *ASG-Encore User's Guide* for more details about the Extract facility.

Migration

The most basic feature of Encore is to convert an existing MVS COBOL application to a component-based architecture. Encore provides these capabilities for migration:

- Clear separation of user interface and data access code from the business logic.
- Removal of redundant logic.

Target Generation Overview

Target Generation addresses the disposition of the segment created in the Code Extraction phase. For Target Generation, Encore provides these capabilities:

- Separate program generation.
- Callable module generation.
- Complement module generation consisting of the original program with the extracted Logic Segment removed.
- IO module generation containing all input/output functions for an FD (File Description).
- Logical COBOL segment access with an editor, such as ISPF Edit or SmartEdit for modification and customization.
- Knowledge retention that helps with everyday maintenance tasks or major system rewrites.
- Logic Segment extraction used for input to CASE tools.

The Target Generation environment is a powerful lever against change. When business needs call for enhancements to a specific function, only one Logic Segment is impacted, dramatically reducing the time required for analysis and change.

Operating Environment

This section describes the platforms on which Encore runs and the hardware and software requirements for running the application.

Environment

Encore products have these requirements:

Platform	Software
Host	<ul style="list-style-type: none"> MVS/XA, MVS/ESA, or OS/390 MVS ISPF Version 3.5 thru 4.8 (3.1 is the minimum supported release level, Estimate requires ISPF 3.3 (minimum supported release level)) BATCH region size of 4096 KB or larger TSO logon region size of 2048 KB or larger below 16 MB Direct Access storage 3270 type terminals; Models 2, 3, 4, or 5 VSAM (required if any AKR is allocated as VSAM)
Clients	Windows 95/98/2000 or Windows NT with access to MVS
Servers	OS/390

COBOL Compilers

ESW products support these COBOL compilers:

Product	COBOL Compiler
Compiler	<ul style="list-style-type: none"> COBOL D, E, and F COBOL II COBOL/370 COBOL for MVS and VM COBOL for OS/390 and VM

Note:

Object oriented extensions added to COBOL for MVS and VM, and to COBOL for OS/390 and VM are not supported.

Preprocessor Support

Encore supports these preprocessor languages directly:

Product	Language
Processor	<ul style="list-style-type: none">• Command-level CICS• Command-level DL/I• IDMS• SQL

Note: _____

Other preprocessed languages are not recognized, but can be supported from the generated COBOL code.

2

Customization

This chapter describes customizing specific Encore components, invoking Encore, and validating Encore installation, and contains these sections:

Section	Page
Prerequisite	7
Step 1 - Modifying CNTL Library Members	8
Step 2 - Adding Encore Modules to MLPA/PLPA	10
Step 3 - Customizing Panvalet or Librarian	11
Step 4 - Customizing the Module Header Box	13
Step 5 - Setting up the Encore/COBOL/SF Interface	13
Step 6 - Invoking Encore	14
Step 7 - Validating Encore	15

Prerequisite

Center installation and customization must be performed before customizing Encore.

Note: _____

If Center has not been installed, see the *ASG-Center Installation Guide* before attempting to install or customize Encore.

Step 1 - Modifying CNTL Library Members

Use these CNTL library members to compile and analyze Encore demonstration programs. Modify these members to specify a valid JOB card and correct parameter values, but do not execute them at this time:

Member	Description
VIAACMP1	JCL to compile the ASG VIARDEMO demonstration program.
VIAACMP2	JCL to compile the ASG VIARDRIV and VIARSUBR demonstration programs.
VIAACMP3	JCL to compile the ASG VIARANOM demonstration program.
VIAACMP4	JCL to compile the ASG VIARBRWS demonstration program.

Specify the correct values for these parameters:

Parameter	Description
VIASOFT	Specify the high-level node where the ESW products are installed.
CENTER	Specify the second-level node where ESW products are installed. If the ESW dataset names contain more than three nodes, then specify all nodes except the first and the last as CENTER. For example, the dataset name SYS3.CENxx.NEW.LOADLIB should have VIASOFT=SYS3 and CENTER=CENxx.NEW.
SYSOUT	Specify the correct SYSOUT character.
SYSDA	Specify the appropriate UNIT for temporary datasets.
COMPILR	Specify the COBOL compiler load module name. It currently defaults to the COBOL II compiler.
COBCOMP	Specify the COBOL compiler load library name. It currently defaults to the COBOL II library name.

Specify the correct values for these VIAACMP3 and VIAACMP4 CNTL library member parameters:

Parameter	Description
CICS	Specify the high-level node(s) where the CICS Command Translator/Preprocessor has been installed.
CICSCOB	Specify the low-level node where the CICS Command Translator/Preprocessor copybooks have been installed.
CICSLIB	Specify the low-level node where the CICS Command Translator/Preprocessor load library has been installed.
CICSTRAN	Specify the CICS Command Translator/Preprocessor load module name.
TRNPARAM	Specify the CICS Command Translator/Preprocessor parameters that are required by your facility.

To override the default Encore installation options

Perform this action only to override Encore default installation options.

- Edit the ASG.VIACEN_{xx}.CNTL member VIA\$PRMA and modify these options.
 - COBOLSF
 - Gen-COBOL-Last-Node=COBOL
 - NETRON
 - Export

See the Installation Option appendix in the *ASG-Center Installation Guide* for information about changing the installation option values.

Step 2 - Adding Encore Modules to MLPA/PLPA

These are the Encore load modules that can be added to the Modified Link Pack Area (MLPA) or Pageable Link Pack Area (PLPA), which are re-entrant and linked AMODE(31), RMODE(ANY).

ASG Module Additions to MPLA/PLPA			
VIAENC	VIARCMMP	VIARCSFI	VIARCSFM
VIARISPP	VIARICSF	VIARCSOM	VIARWKSP
VIAZBTCH	VIAZCMDP	VIAZGENP	VIAZPRIP
VIAZTBLP			

For the names of all the modules that are eligible for location in the MLPA or PLPA, see the installation guide for the other ESW products you have installed.

These are some advantages to locating modules in the MLPA or PLPA:

- A reduction of the memory requirement per user
- An overall decrease in required swap space
- Performance improvements

Moving these modules to MLPA/PLPA is optional. It is recommended that the original ESW load library (from the installation tape) be kept as a staging library so that you can apply any required PTFs. You can copy the re-entrant modules to LPA and the non-re-entrant modules to a separate user library. These steps also require changes to the user's logon or product allocations. The CNTL library contains two members, VIASLPAJ and VIASLPXJ, which can be used to perform these steps.

Note: _____

Do not use the ISPF 3.3 copy feature to copy these modules because some of them have aliases.

Step 3 - Customizing Panvalet or Librarian

Panvalet Customization

To customize Panvalet

Note: _____

Do not perform [step 1](#) if you are using Panvalet R12 and the module PAM is available in LINKLST or LPA.

- 1 Edit CNTL member VIASPAMJ and specify a valid JOB card and the correct values for these parameters:
 - ASMBLR
 - PANLIB
 - SYSOUT
 - SYSDA
 - VIASOFT
 - CENTER

- 2 Submit this job to assemble and link edit the Panvalet VIASPAM module.

If Panvalet requires changes to its customized edit panel, these changes may need to be included in the ESW panel VSPEDPAN. If this situation occurs, contact the ASG Service Desk for instructions.

If the Panvalet load library is allocated under ISPLLIB and the LIBDEF facility is used, WTP MEMBER NOT FOUND warning messages display each time you select a Panvalet member. Try these alternatives to correct the problem:

- Adding Encore allocations to LOGON and bypassing LIBDEF processing.
- Adding the Panvalet load library allocation to VIALLIB in LIBDEF processing.
- Setting WTPMSG OFF (not recommended).

Librarian Customization

To customize Librarian

- 1 Edit CNTL member VIASFAIJ and specify a valid JOB card and the correct values for these parameters:

- ASMBLR
- LIBRLIB
- LIBRMAC
- SYSOUT
- SYSDA
- VIASOFT
- CENTER

If you are running on Librarian R3.9, edit CNTL member VIASFAIR and comment out the four lines containing these delete statements:

```
DELETE EPLOC=XFAIROPN
DELETE EPLOC=XFAIRMOD
DELETE EPLOC=XFAIRREC
DELETE EPLOC=XFAIRCLS
```

- 2 Submit this job to assemble and link edit the Librarian VIASFAIR module.

If Librarian requires changes to its customized edit panel, these changes may need to be included in the ESW panel VSPEDLIB. If this situation occurs, contact the ASG Service Desk for instructions.

If the Librarian load library is allocated under ISPLLIB and the LIBDEF facility is not used, WTP MEMBER NOT FOUND warning messages display each time you select a Librarian member. Try these alternatives to correct the problem:

- Adding Encore allocations to LOGON and bypassing LIBDEF processing.
- Adding the Librarian load library allocation to VIALLIB in LIBDEF processing.
- Setting WTPMSG OFF (not recommended).

Step 4 - Customizing the Module Header Box

When a module is generated, a header box is placed at the beginning of the source code. You can modify the template for the header box to conform to site standards.

To modify the header box template

- ▶ Edit CNTL member VIARCBSK. All text in the member can be edited except for the symbolic parameters (i.e., parameters that begin with &).

Step 5 - Setting up the Encore/COBOL/SF Interface

To interface between Encore and COBOL/SF

- 1 Edit CNTL member VIA\$PRMA and add COBOLSF=YES.
- 2 Edit CNTL member VIARCSFP. This member specifies the dataset name allocations to be performed when COBOL/SF is invoked during the ANALYZE process.
 - a Verify, and change if necessary, the high-level node for the COBOL/SF libraries. The default is ECF.V2R2M1.
 - b Specify a VOLSER for the SSQTEXT VSAM dataset. The default is question marks.
- 3 Edit CLIST member VIAACSF. Verify, and change if necessary, the high-level node for the COBOL/SF libraries. The default is ECF.V2R2M1.
- 4 Ensure that ECF.V2R2M1.SECFEXC0 is included in SYSPROC allocation. This is described in the Host Installation section of the Program Directory for COBOL/SF.

Step 6 - Invoking Encore

To invoke Encore from an ISPF panel

Use these panel definition cards to add an Encore option to the ISPF primary menu or another dialog menu.

- This entry describes the Encore option to the user:

```
% EN +ASG Encore - COBOL re-engineering Environment
```

- This entry invokes Encore when the letters EN are entered:

```
EN, 'CMD(%ENCORE) NEWAPPL(VIAR)'
```

Note: _____

After updating the ISPF environment you may need to re-enter ISPF before the facilities become available.

To invoke Encore using a CLIST

Use either of these methods to invoke Encore using a CLIST:

- Type TSO VIAENC on the command line in the ESW Primary screen. The VIAENC CLIST invokes the ENCORE CLIST while specifying the NEWAPPL(VIAR) parameter to set the correct application ID.
- Type EN on the command line in the ESW Primary screen. The EN CLIST automatically invokes a pre-defined ENCORE CLIST.

Step 7 - Validating Encore

Note: _____

The dataset names used in the following validation steps are the default installation names. If you have changed them, use the changed names in place of the default names.

To verify logon library allocations

- 1 Enter Encore by selecting the correct option from the appropriate menu.

Note: _____

If you installed Encore as indicated in ["Step 6 - Invoking Encore" on page 14](#) use the appropriate selection on the ISPF menu or the CLIST.

- 2 Select Help ► About to display the Help - About pop-up, as shown in [Figure 1](#).

This pop-up displays the product release and level of Encore that has been installed. It also contains the product name, product release number, product maintenance level, and the operation system.

Figure 1 • Help - About Pop-up

```

File View Extract Generate Search List Options Help
-----
C |                                     Help - About                                | ogram: VIARBRWS
  |-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
  | The following is release information for this ASG                          | *****
  | product.                                                                    | *****
  |                                                                              |
  | Product name . . . . . : ASG-Encore                                         | *****
  | Release number . . . . . : x.x                                             | *****
  | Maintenance level . . . . . : xxx                                          | *****
  |                                                                              |
  | ASG-CENTER release number : x.x                                           | *****
  | Maintenance level . . . . . : xxx                                          | *****
  |                                                                              |
  | Operating system . . . . . : OS(390)                                       | *****
  |-----|-----|-----|-----|-----|-----|-----|-----|-----|
                                     Copyright Allen Systems Group, Inc., an unpublished work.
                                     A proprietary product of ASG, Inc. Use restricted to authorized licensees.
                                     Visit the ASG Support Web Site at www.asg.com

```

- 3 Press PF3/15 to exit the Help - About pop-up.
- 4 Review and/or modify Encore options by selecting the Options pull-down from the action bar.

- c Select Options ► Log/list/punch to display the Options - Log/List/Punch Definition pop-up, as shown in [Figure 4](#). Review and/or modify the log, list, and punch file defaults. Enter the Job statement information and press PF3/15 to exit.

Figure 4 • Options - Log/List/Punch Definition Pop-up

```

-----
Options - Log/List/Punch Definition
Command ==> _____

1 - Process log file  2 - Process list file  3 - Process punch file
                    4 - Customized data set name

Options              Log              List              Punch
-----
Process option      . . . . . PD          PK              K
Primary tracks      . . . . . 1           1              1
Secondary tracks    . . . . . 2           5              5
Lines per page      . . . . . 56          56             56
Sysout class        . . . . . *           *              *

Process options:  PK (print/keep), PD (print/delete), K, or D.

Job statement information:
//NAME JOB (ACCOUNT),'(NAME)',
//          MSGCLASS=A
//*      INSERT '/*ROUTE PRINT NODE.USER' HERE IF NEEDED.
//*
```

- d Select Options ► PF keys to display the Options - PF Key Definition pop-up, as shown in [Figure 5](#). Review and/or modify the PF key definitions (press Enter to switch between PF keys 1-12 and 13-24). Press PF3/15 to exit.

Figure 5 • Options - PF Key Definition Pop-up

```

File View Extract Generate Search List Options Help
-----
Options - PF Key (01-12) Definition
Command ==> _____
RWS

Press Enter to process changes and/or to display alternate keys.
Press PF3/15 (END) to exit.

Number of PF keys: 24      Terminal type: 3278

PF01 HELP
PF02 SPLIT
PF03 END
PF04 ZOOMIN
PF05 RFIND
PF06 FX %
PF07 UP
PF08 DOWN
PF09 SWAP
PF10 LEFT
PF11 RIGHT
PF12 RECALL
S.
```

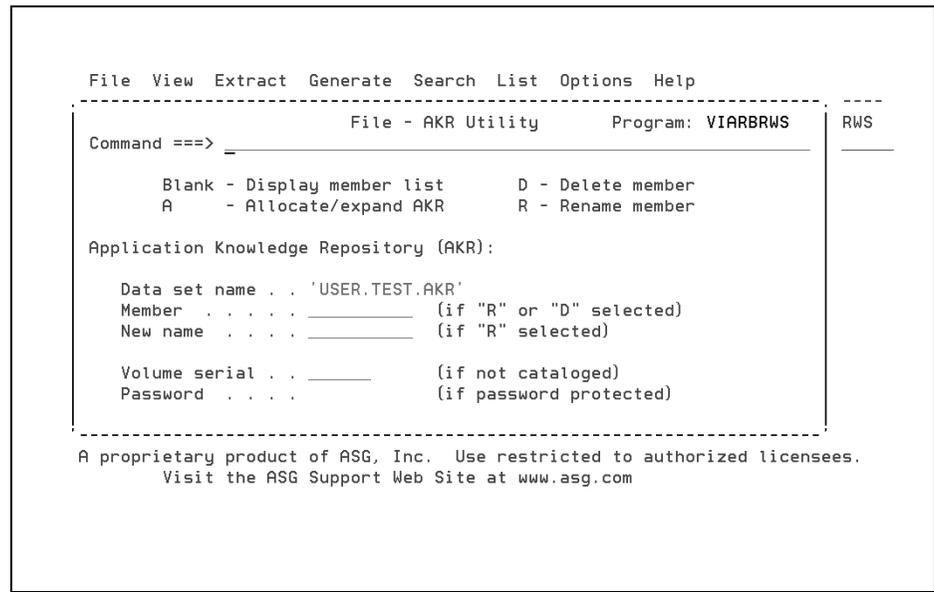
To allocate an AKR

- 1 Select File ▶ AKR Utility to display the File - AKR Utility pop-up, as shown in [Figure 6](#).

Note:

If you have already created an AKR when validating another ESW product, proceed to the section ["To analyze the demonstration program" on page 20](#).

Figure 6 • File - AKR Utility Pop-up



- 2 Complete the File - AKR Utility pop-up by typing the name of an AKR to be allocated. Type an A on the command line and press Enter to display the File - AKR Allocation/Expand pop-up (see [Figure 7 on page 19](#)).

Note:

The fields on the File - AKR Allocate/Expand pop-up may vary depending on the settings for the Center installation parameters, AKR-DSORG-VSAM and SMS. See the *ASG-Center Installation Guide* for more information.

- 3 Verify that the name is correct and enter the appropriate values for the Management Class, Storage Class, and Data Class fields if you are using SMS managed datasets.

[Figure 7](#) shows a sample of how the File - AKR Allocate/Expand pop-up looks for a VSAM AKR without SMS support installed.

Figure 7 • File - AKR Allocate/Expand Pop-up for a VSAM AKR without SMS

```

File View Extract Generate Search Logic List Options Help
-----
File - AKR Allocate/Expand
Command ==> _____
          S - Submit JCL      E - Edit JCL      C - Specify Catalog
Expand existing AKR . . . NO          (Yes or No)
AKR data set name . . . 'USER.TEST.AKR'
Volume . . . . . _____
Unit . . . . . SYSDA          (Generic unit name)
Space units . . . . . RECORDS  (Records, Tracks or Cylinders)
Primary space . . . . . 4000    (Primary amount in above units)
Secondary space . . . . . 0     (Secondary amount in above units)

Job statement information:
//USER JOB (ACCOUNT),NAME,
//      MSGCLASS=A
//*   INSERT '/*ROUTE PRINT NODE.USER' HERE IF NEEDED.
//*

```

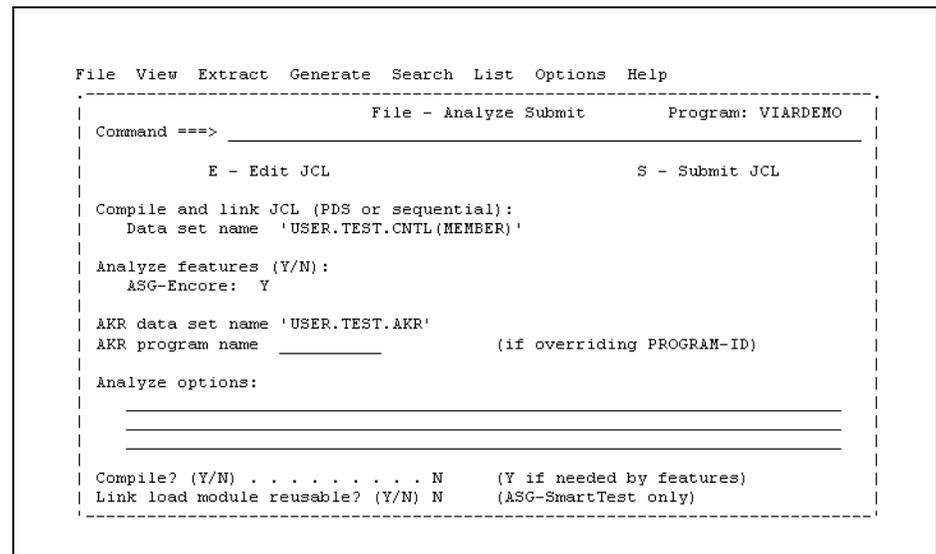
If you did not specify SMS managed datasets in the CENTER installation parameter SMS while installing Center, the fields pertaining to SMS are not displayed. In that case, enter these values as required:

- a For VOLUME, type the value where the permanent AKR is to be placed.
- b Review the Space units and Space amount values.
- c Type the appropriate UNIQUE parameter for the volume selected if you are allocating a VSAM AKR.
- d If this dataset needs to be cataloged under a user catalog, type C on the command line to display the AKR Catalog Information pop-up. Enter the Catalog DSN and password as required, and press PF3/15 to exit.
- e Type the JOB statement information for your site and type S on the command line to submit the job.
- f After the job finishes, verify that the AKR was successfully allocated and initialized.
- g Press PF3/15 until you return to the Encore Primary screen.

To analyze the demonstration program

- 1 Select File ► Analyze to display the File - Analyze Submit pop-up, as shown in [Figure 8](#).
 - a Type the AKR dataset name from [step 2 on page 18](#).
 - b Set the Compile flag to N.
 - c Type S on the command line to submit the job.
 - d Verify that the analyze job completed successfully. If the analyze job did not completed successfully, see the *ASG-Center Installation Guide* for additional information.

Figure 8 • File - Analyze Submit Pop-up



These are the compile and link JCL dataset names required for the Compile and link JCL field on the File - Analyze Submit pop-up shown in [Figure 8 on page 20](#):

Dataset/Member	Description
ASG.VIACENxx.CNTL(VIAACMP1)	For program VIARDEMO.
ASG.VIACENxx.CNTL(VIAACMP2)	For programs VIARDRIV and VIARSUBR.
ASG.VIACENxx.CNTL(VIAACMP3)	For program VIARANOM (*). You can use the Analyze options field shown in Figure 8 on page 20 to enter analysis report parameters or to use the batch-driven Anomaly Repair Facility.
ASG.VIACENxx.CNTL(VIAACMP4)	For programs VIARBRWS.

(*) For VIARANOM, you can use the Analyze options field on the File - Analyze Submit pop-up to enter these analysis report parameters. These parameters are used to produce analysis reports that help you understand and diagnose analysis problems and anomalies:

- VIARERPT - stops the analysis process after the analysis report is generated.
- VIARERPTAN - continues the analysis process after report generation and saves the information in the AKR.
- NUMPRMA(##) - specifies the number of parameters allowed before a CALL is listed in the Excessive Parameter List report.
- RPTCEN - centers the generated report (default is left justified).

You can also use these analyze parameters in the Analyze options field to use the batch-driven Anomaly Repair Facility to correct certain anomalies that may exist within your code:

- VIAANOMF - runs the Anomaly Repair Facility in batch mode. You can also use the online method by entering these additional operands on the Anomaly Facility - Batch Submit screen:
 - PARM(user parm) - any valid user parameter (e.g., DEBUG).
 - ANOMALY - required, used to specify these types of anomaly fixes:
 - OOPGT - for out of perform GOTO
 - GTD - for GOTO depending
 - IPR - for inconsistent performance ranges
 - COMMENT(YES or NO) - includes/omits original commented out code in new repaired code. YES is the default.

- DSN - name of the output dataset for the repaired code. If you do not specify a dataset name, output defaults to DD VIAANOUT. If DSN is a sequential file, do not code the MEM parameter.
- MEM - member name of the output dataset if you specified a DSN. Do not code if DSN is a sequential file.
- HEADER(YES or NO) - includes/omits an ASG header at the beginning of repaired code. YES if the default.

Example:

```
VIAANOMF , PARM (DEBUG1) , ANOMALY (OOPGT , GTF , IPR) , COMMENT (YES) ,  
DSN ( ' ASG . ANOMALY . REPAIRED . CODE ' ) , MEM (MEM1) , HEADER (YES)
```

See the online help, the *ASG-Encore User's Guide*, and the *ASG-Encore Reference Guide* for more information about these parameters.

- 2** Validate the Analyze Submit CLIST by entering either of these commands from the ISPF Primary Menu:

```
TSO VIASUBDS 'ASG.VIACENxx.CNTL(VIAACMP1)'
```

Or

```
TSO VIASUBDS 'ASG.VIACENxx.CNTL(VIAACMP2)'
```

Repeat [step a](#) through [step d](#) from "[To analyze the demonstration program](#)" on page [20](#).

See the online help for more information about the Analyze Submit pop-up.

To verify Structure View

- 1 Select File ▶ Open to display the File - Open Program pop-up.
- 2 Enter the AKR dataset name and the program name VIARDEMO. If applicable, enter the appropriate dataset password. Press Enter to open the program and display the Encore Primary screen.
- 3 Select View ▶ Structure to display the View - Structure View Request pop-up, as shown in [Figure 9](#).

Figure 9 • View - Structure View Request Pop-up

```

File View Extract Generate Search Logic List Options Help
-----
Comma |                                     View - Structure View Request
      |
      | Type request options. Then press Enter. For a name selection list
      | for a range (other than 1 or 2), type a pattern (e.g. ABC*) in the
      | name field.
      |
      | Levels 3   of   3   of depth
      |
      | Range
      | 1  1. Procedure division
      |   2. All entry points
      |   3. Perform range name
      |   4. Entry point name
      |   5. Nested program name
      |
      | Name _____
A pr  |
      | Options
      |  _  Repetition
      |
      |-----

```

- 4 Enter these values in the appropriate fields:
 - Levels: 3 (maximum)
 - Range: 1 (procedure division)

The Name field and Repetition option should be left blank. Press Enter to display the Structure View screen showing the message ASG1582I 3 OF 3 LEVELS DISPLAYED IN THE VIEW.

- 6** Leave the Options and (if present) Size change levels fields blank. Press Enter to redisplay the Structure View screen.

On the Structure View screen, the message ASG0443I 3 DATA REFS: 1 DEF, 1 USE, 1 MOD, FOUND FOR WK-INTEREST should display. The box labeled UPDATE-ACCT THRU UPDATE* should be highlighted.

Note: _____

If Insight is installed complete [step a](#) through [step d on page 26](#). If Insight is not installed, proceed to [step 7 on page 26](#).

If Insight is installed:

- a** Select Logic ► Data to display the Logic Order Search - Data Name pop-up and enter this information:
- Data Name: WK-INTEREST
 - Start line/label: PROCEDURE
 - References: 1
 - Direction: 1
 - Options: Blank
 - Action: 1
- b** Press Enter. The short message, 1 REF(S) REACHABLE displays. The box labeled UPDATE-ACCT THRU UPDATE* should be highlighted.

- c Select View ► Zoom In, or type ZOOMIN on the command line and move the cursor to the highlighted box (UPDATE-ACCT THRU UPDATE*). Press Enter to display the View - Source pop-up, as shown in [Figure 11](#). This pop-up displays with the code for the selected unit highlighted and the rest of the program excluded.

Figure 11 • View Source Pop-up

```

-----
|  View Search Logic List Options Help  |
|-----|
|                                     |
|                                     |
| Command ==> _____ SCREEN POSITIONED |
|                                     |
| 000273 UPDATE-ACCT. PERF RNG |
|                                     |
| 000275     PERFORM INIT-ACTION-RECORD. 1 LINE NOT DISPLAYED |
|                                     |
| 000277     IF ACCT-TYPE OF ACCT-RECORD = 'INT' PERF RNG |
| 000278     MOVE CHK-INTEREST-RATE TO INTEREST-RATE PERF RNG |
| 000279     ELSE IF ACCT-TYPE OF ACCT-RECORD = 'CHK' PERF RNG |
| 000280     MOVE ZERO TO INTEREST-RATE PERF RNG |
| 000281     ELSE IF ACCT-TYPE OF ACCT-RECORD = 'MMA' PERF RNG |
| 000282     MOVE MMA-INTEREST-RATE TO INTEREST-RATE PERF RNG |
| 000283     ELSE IF ACCT-TYPE OF ACCT-RECORD = 'SAV' PERF RNG |
| 000284     MOVE SAV-INTEREST-RATE TO INTEREST-RATE. PERF RNG |
|                                     |
| 000286     IF ACCT-MULTI-ACCT = 'YES' 1 LINE NOT DISPLAYED |
| 000287     ADD CHK-MULTI-ACCT-RATE TO INTEREST-RATE PERF RNG |
| 000288     MOVE ACTION-MULTI-ACCT TO WORK-ACTION-REASON PERF RNG |
| 000289     PERFORM INITIATE-ACTION. PERF RNG |
|-----|

```

- d Press PF3 twice to return to the Encore Primary screen.

If Insight is not installed:

- 7 Select View ► Tree to display the View - Tree View Request pop-up and enter this information:
 - Levels: 4 (maximum)
 - Contents: 1
 - Range: 1
- 8 Leave the Name and Repetition option fields blank. Press Enter to display the Tree View screen ([Figure 12 on page 27](#)).

- 9 The message ASG1582I 4 OF 4 LEVELS DISPLAYED IN THE VIEW displays, as shown in [Figure 12](#). Type LEVELS 3 on the command line and press Enter.

Figure 12 • Tree View Screen

```

File View Extract Generate Search List Options Help
-----
Tree View                                     LINE 1 OF 21
Command ==> _____ Scroll ==> CSR

***** ***** TOP OF DATA *****
000000 PROCEDURE DIVISION USING CONTROL-PARAMETERS, BRANCH-BALANCE,
000002     INITIALIZE-PGM.
000003     OPEN-FILES.
000003     GET-NUM-OF-DAYS.
000004     SUM-TOTAL-DAYS.
000002     ACCT-MAINTENANCE.
000003     ACCT-MAINTENANCE-EXIT.
000003     INITIATE-ACTION.
000003     ACCT-MAINTENANCE-EXIT. REPEATED
000003     UPDATE-ACCT.
000004     INIT-ACTION-RECORD.
000004     INITIATE-ACTION. REPEATED
000004     INITIATE-ACTION. REPEATED
000004     INITIATE-ACTION. REPEATED
000003     ASG1582I 4 OF 4 LEVELS DISPLAYED IN THE VIEW.
000002     ACCT-MAINTENANCE-EXIT.

```

The message ASG1582I 3 OF 4 LEVELS DISPLAYED IN THE VIEW displays as the number of levels displayed is reduced to 3, as shown in [Figure 13](#).

Figure 13 • Tree View Screen with Levels Displayed Message

```

File View Extract Generate Search List Options Help
-----
Tree View                                     LINE 1 OF 15
Command ==> _____ Scroll ==> CSR

***** ***** TOP OF DATA *****
000000 PROCEDURE DIVISION USING CONTROL-PARAMETERS, BRANCH-BALANCE,
000002     INITIALIZE-PGM.
000003     OPEN-FILES.
000003 +   GET-NUM-OF-DAYS.
000002     ACCT-MAINTENANCE.
000003     ACCT-MAINTENANCE-EXIT.
000003     INITIATE-ACTION.
000003     ACCT-MAINTENANCE-EXIT. REPEATED
000003 +   UPDATE-ACCT.
000003     UPDATE-ACCT-EXIT.
000003     EXCEPTION-REPORT-1.
000002     ACCT-MAINTENANCE-EXIT.
000002     REPORT-FINAL-CNTRS.
000002     CLOSE-PGM.
000003     ASG1582I 3 OF 4 LEVELS DISPLAYED IN THE VIEW. *****

```

- 10 Type JUMP on the command line, as shown in [Figure 14](#), position the cursor on any of the source lines, and press Enter.

Figure 14 • Tree View Screen

```

File View Extract Generate Search List Options Help
-----
Command ==> JUMP_----- Tree View LINE 1 OF 15
                                Scroll ==> CSR

***** ***** TOP OF DATA *****
000000 PROCEDURE DIVISION USING CONTROL-PARAMETERS, BRANCH-BALANCE,
000002     INITIALIZE-PGM.
000003     OPEN-FILES.
000003 +     GET-NUM-OF-DAYS.
000002     ACCT-MAINTENANCE.
000003     ACCT-MAINTENANCE-EXIT.
000003     INITIATE-ACTION.
000003     ACCT-MAINTENANCE-EXIT. REPEATED
000003 +     UPDATE-ACCT.
000003     UPDATE-ACCT-EXIT.
000003     EXCEPTION-REPORT-1.
000002     ACCT-MAINTENANCE-EXIT.
000002     REPORT-FINAL-CNTRS.
000002     CLOSE-PGM.
000003
***** ***** ASG15821 3 OF 4 LEVELS DISPLAYED IN THE VIEW. *****

```

The View - Source pop-up displays with the cursor positioned on the selected line, as shown in [Figure 15](#).

Figure 15 • View - Source Pop-up Showing JUMP Results

```

View Search List Options Help
-----
Command ==> ----- View - Source Program: VIARDEMO
                                Scroll ==> CSR

000243 ACCT-MAINTENANCE.
000244
000245     READ TRAN-FILE
000246     AT END
000247     MOVE EOF-LITERAL TO TRAN-FILE-FLAG
000248     GO TO ACCT-MAINTENANCE-EXIT.
000249
000250     ADD +1 TO TRAN-CNT.
000251
000252     MOVE TRAN-ACCT-KEY TO ACCT-KEY.
000253
000254     READ ACCT-FILE KEY IS ACCT-KEY
000255     INVALID KEY
000256     MOVE ACTION-MISSING-ACCT TO WORK-ACTION-REASON
000257     MOVE SPACES TO ACCT-RECORD
000258     PERFORM INITIATE-ACTION
000259     MOVE +4 TO WORK-RETURN-CODE

```

- 11 Press PF3 twice to return to the Encore Primary screen.

To verify Perform Range extract processing

- 1 Select Extract ► Perform Range and press Enter to display the Extract - Name Logic Segment pop-up.
- 2 Accept the default name or type PERFORM-RANGE-0001 in the Name field, and press Enter to display the Extract - Perform Name List pop-up.
- 3 Press PF5 (Filter) to display the Extract - Perform Range Filter pop-up, as shown in [Figure 16](#).
- 4 In the COBOL Subset field, type the subset name OUTPUT and press Enter to redisplay the Extract - Perform Name List pop-up, which shows the PERFORM Ranges associated with the COBOL subset OUTPUT (see [Figure 17 on page 30](#)).

Figure 16 • Extract - Perform Range Filter Pop-up

```

-----
C |                                     Extract - Perform Name List
S |                                     Extract - Perform Range Filter
E | Type information for filtering PERFORM Range Name list.
|
| PERFORM Name _____
|
| Data Name . . _____
|
| Use Context                               Options
| 1  1. References                          /  Aliasing
| 2  2. Uses
| 3  3. Modifications
|
| Line Range . . _____
| COBOL Subset  OUTPUT
| Text Pattern  _____
|
| REPORT-FINAL-CNTRS          5  2.38    1    1    0    0
|
| PF4=View  PF5=Filter  PF17=Sort
-----

```

- Type S on the INITIATE-ACTION line and press Enter to highlight the line, as shown in [Figure 17](#).

Figure 17 • Extract - Perform Name List Pop-up

```

-----
Command ==> _____ Scroll ==> CSR
Select PERFORM Range name(s) for extract. Then press Enter. Press
Enter when complete. Use SORT command to order column by value.
S - Select      V - View Source      U - Unselect

PERFORM Range Name          Size      Times PERF Direct Subord
                             Lines %      PERF   by  PERFs  PERFs
-----
_ ACCT-MAINTENANCE THRU ACCT-MAI  27 12.86    1    1    3    4
_ EXCEPTION-REPORT-1             16  7.62    1    1    0    0
s INITIATE-ACTION                 13  6.19    5    2    0    0
_ REPORT-FINAL-CNTRS              5  2.38    1    1    0    0
_ UPDATE-ACCT THRU UPDATE-ACCT-E  52 24.76    1    1    2    2
***** BOTTOM OF DATA *****

PF4=View PF5=Filter PF17=Sort
-----

```

- Press Enter again to return to the Encore Primary screen.

- 7 Select View ► Logic Segment to display the View - Select Logic Segment pop-up, as shown in [Figure 18](#). Select PERFORM-RANGE-0001 and press Enter.

Figure 18 • View - Select Logic Segment Pop-up

```

View - Select Logic Segment          1 LOGIC SEGMENTS
Command ==> _____ Scroll ==> CSR
S - Select
-----
Program   Logic Segment Name      Description
-----
S VIARDEMO PERFORM-RANGE-0001      Perform Range
***** BOTTOM OF DATA *****

```

The Source View screen displays with the logic segment lines highlighted and the non-segment lines excluded, as shown in [Figure 19](#).

Figure 19 • Source View Screen

```

File View Extract Generate Search List Options Help
-----
Command ==> _____ Source View          Program: VIARDEMO
                                          Scroll ==> CSR
-----
000243 ACCT-MAINTENANCE.                19 LINES NOT DISPLAYED
                                           LOGICSEG
000245 READ TRAN-FILE                    1 LINE NOT DISPLAYED
                                           LOGICSEG
000246 AT END                            LOGICSEG
000247 MOVE EOF-LITERAL TO TRAN-FILE-FLAG LOGICSEG
000248 GO TO ACCT-MAINTENANCE-EXIT.     LOGICSEG
-----
000250 ADD +1 TO TRAN-CNT.                1 LINE NOT DISPLAYED
                                           LOGICSEG
000252 MOVE TRAN-ACCT-KEY TO ACCT-KEY.   1 LINE NOT DISPLAYED
                                           LOGICSEG
-----
000254 READ ACCT-FILE KEY IS ACCT-KEY    1 LINE NOT DISPLAYED
                                           LOGICSEG
000255 INVALID KEY                       LOGICSEG
000256 MOVE ACTION-MISSING-ACCT TO WORK-ACTION-REASON LOGICSEG
000257 MOVE SPACES TO ACCT-RECORD       LOGICSEG
000258 PERFORM INITIATE-ACTION           LOGICSEG
000259 MOVE +4 TO WORK-RETURN-CODE       LOGICSEG
000260 GO TO ACCT-MAINTENANCE-EXIT.     LOGICSEG

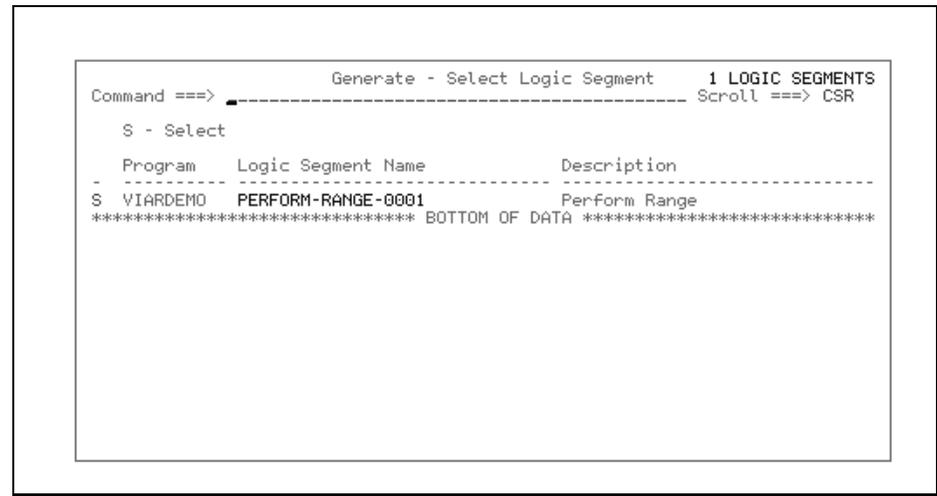
```

- 8 Press PF3 to return to the Encore Primary screen.

To verify Logic Segment generation processing

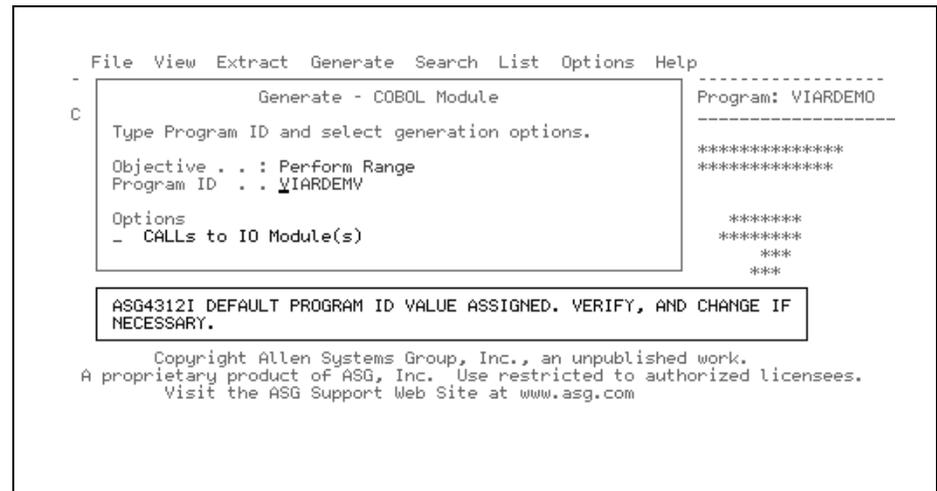
- 1 Select Generate ► COBOL Module to display the Generate - Select Logic Segment pop-up, as shown in [Figure 21](#). Select PERFORM-RANGE-0001 and press Enter to display the Generate - COBOL Module pop-up (see [Figure 22](#)).

Figure 21 • Generate - Select Logic Segment Pop-up



VIARDEMV displays in the Program ID field on the Generate - COBOL Module pop-up. Leave the Options field blank.

Figure 22 • Generate - COBOL Module Pop-up



- 2 Press Enter to display the Generate - Specify Perform Range ENTRY Names pop-up, as shown in [Figure 23](#).

Figure 23 • Generate - Specify Perform Range ENTRY Names Pop-up

```
-----  
Generate - Specify Perform Range ENTRY Names  
Command ==> _____ Scroll ==> CSR  
  
Specify the ProgramID and ENTRY names of the modules to be generated.  
S - Split H - Move I - Into + - Expand - - Collapse  
  
Program ENTRY PERFORM Range Name  
-----  
***** TOP OF DATA *****  
VIARDEMY *INITIATE-ACTION  
***** BOTTOM OF DATA *****  
  
-----
```

- 3 Press Enter to display the Generate - Multi-module PERFORM Ranges pop-up (see [Figure 24 on page 35](#)).

- 4 Select VIARDEMV and press Enter to display the Edit screen. A COBOL subprogram corresponding to the logic segment is generated. The Edit screen appears displaying the subprogram. The message ASG1608I ENTER THE CREATE COMMAND TO SAVE THE COBOL PROGRAM also displays.

Figure 24 • Generate - Multi-module PERFORM Ranges Pop-up

```

-----
Generate - Multi-module PERFORM Ranges
Command ==> _____ Scroll ==> CSR

Select the module to generate, then press Enter.

Program ID  ENTRY Name  PERFORM Range Name
-----
***** TOP OF DATA *****
S VIARDEMV          INITIATE-ACTION
***** BOTTOM OF DATA *****
-----

```

- 5 Press PF3 four times to return to the Encore Primary screen.

- Type DISPLAY on the command line and press Enter, as shown in [Figure 28](#). The lines of the current logic segment are displayed and the non-segment lines are excluded.

Figure 28 • Editor Screen

```

ASG-Encore EDIT - VIARN60.PROD.CNTL(VIARDEMO) - 60.00      COLUMNS 00001 00072
Command ==> DISPLAY                                     Scroll ==> CSR
***** ***** Top of Data *****
000100 000100 IDENTIFICATION DIVISION.
000200 000200 PROGRAM-ID. VIARDEMO.
000300 000300 AUTHOR. ASG DEMONSTRATION.
000400 000400
000500 000500*****
000600 000600*   THIS PROGRAM DOES THE MONTH END CALCULATIONS FOR EACH TYPE *
000700 000700*   OF CHECKING ACCOUNT CARRIED BY THE UNIVERSAL BANK. IT *
000800 000800*   HANDLES THE FOLLOWING TYPES OF ACCOUNTS. *
000900 000900*   CHECKING WITH INTEREST *
001000 001000*   CHECKING WITH NO INTEREST *
001100 001100*   MONEY MARKET ACCOUNT (MMA) *
001200 001200*   SAVINGS ACCOUNT *
001300 001300*   IT ALSO UPDATES THE ACCOUNT BALANCE, BRANCH BALANCE AND *
001400 001400*   BANK BALANCE. *
001500 001500*****
001600 001600
001700 001700 ENVIRONMENT DIVISION.
001
001 ASG1603I ENTER THE DISPLAY COMMAND FOR LOGIC SEGMENT INFORMATION.
002
002100 002100 INPUT-OUTPUT SECTION.
    
```

- Type FX WK- INTEREST on the command line, as shown in [Figure 29](#), and press Enter.

Figure 29 • Editor Screen - FX WK-INTEREST Command

```

ASG-Encore EDIT - VIARN60.PROD.CNTL(VIARDEMO) - 60.00      COLUMNS 00001 00072
Command ==> FX WK-INTEREST                             Scroll ==> CSR
***** ***** Top of Data *****
HI==> 003800 01 ACCT-RECORD. - - - - - 37 Line(s) not Displayed
- - - - -
HI==> 004000 05 ACCT-KEY. - - - - - 1 Line(s) not Displayed
- - - - -
HI==> 004200 10 ACCT-TYPE - - - - - PIC X(3).
- - - - - 2 Line(s) not Displayed
HI==> 004500 05 ACCT-MIN-BALANCE - - - - - PIC S9(6)V99.
HI==> 004600 05 ACCT-AVG-BALANCE - - - - - PIC S9(6)V99.
HI==> 004700 05 ACCT-MULTI-ACCT - - - - - PIC X(3).
- - - - - 1 Line(s) not Displayed
HI==> 004900 05 ACCT-INTEREST - - - - - PIC S9(6)V99.
- - - - - 15 Line(s) not Displayed
HI==> 005600 01 TRAN-RECORD. - - - - - 1 Line(s) not Displayed
- - - - -
HI==> 005800 05 TRAN-ACCT-KEY. - - - - - 25 Line(s) not Displayed
- - - - -
HI==> 0084 - - - - - E ' '
HI==> 0085 ASG1604I ALL LOGIC SEGMENT LINES WERE DISPLAYED.
- - - - - not Displayed
HI==> 008700 77 CHK-INTEREST-RATE - - - - - PIC S9(4)V9(4) COMP-3
    
```


- 5 Type S on the command line and press Enter to submit the analyze job.
- 6 Verify the analyze job ran successfully by checking for a return code of 0 or 4.

Note: _____

Return codes higher than 0 (zero) usually indicate a compiler message.

To validate DB2 support by analyzing a DB2 program

Note: _____

If you have DB2 installed, verify that ASG.VIACEN_{xx}.CNTL(VIASBIND) has been run (see the *ASG-Center Installation Guide* for more information). This installation step is required for sites using DB2. Failure to complete the DB2 installation step results in erroneous behavior by ESW products.

- 1 Select File ► Analyze to display the File - Analyze Submit pop-up.
- 2 Enter the dataset name for the compile and link JCL for the DB2 program to be analyzed.
- 3 Enter the AKR dataset name.
- 4 Set the Encore analyze flag to Y.
- 5 Type S in the command input area and press Enter to submit the analyze job.
- 6 Verify the analyze job ran successfully by checking for a return code of 0 or 4.

Note: _____

Return codes higher than 0 (zero) usually indicate a compiler message.

- 7 Select File ► Close and press Enter to close the VIARDEMO program. Save any logic segments, if desired.

To verify parameter-driven anomaly repair processing

Complete these steps if you entered any parameter-driven anomaly repair parameters in the Analyze options field on the File - Analyze Submit pop-up (see [Figure 8 on page 20](#)). This section verifies the VIARANOM anomaly repair parameter processing discussed in ["To analyze the demonstration program" on page 20](#).

- 1 Select File ► Open to display the File - Open Program pop-up and select VIARANOM.

- 2 Type VIARERPT in the Analyze options field and submit the VIARANOM analyze job. When the job finishes review the DD VIARERPT output, which shows the anomalies that have been found. Because of the VIARERPT option used in this example, the analyze job will not finish and will not add VIARANOM to the AKR.
- 3 Review the output from DD VIARERPT, noting that VIARANOM contains three anomalies that can be repaired. Perform a second analysis using the these parameters in the Analyze options field:

```
VIAANOMF , ANOMALY ( 00PGT , GTF , IPR )
```

Submit the job and verify that it completes with no errors. Review the DD VIAANOUT output to verify that it contains the repaired code. Verify that an ASG header has been generated at the top of the generated code and that the repaired code is commented out and new code has been added.

- 4 Using the generated repaired anomaly code, perform a normal Encore analyze with no anomaly options or parameters and verify that it analyzed correctly.
- 5 Perform the VIARANOM analyze again, but this time use the DSN parameter to send the output results to an allocated PDS and member. Set the COMMENT and HEADER parameters to NO. Submit the job and verify that the repaired code has been written to the specified PDS and member. Verify that the output does not have an ASG header added to it and that the original fixed code is no longer commented out.
- 6 Perform the VIARANOM analyze one more time, sending the anomaly output to an allocated sequential file. Submit the analyze job and verify that the repaired code was written to the sequential file.
- 7 Select File ► Close and press Enter to close the VIARANOM program. Save any logic segments, if desired.

To verify online panel-driven anomaly repair processing

Complete these steps if you entered any online panel-driven anomaly repair parameters in the Analyze options field on the File - Analyze Submit pop-up (see [Figure 8 on page 20](#)). This section verifies the VIARANOM anomaly repair processing discussed in ["To analyze the demonstration program" on page 20](#).

- 1 Select File ► Open to display the File - Open Program pop-up and select VIARANOM.
- 2 Type VIARERPTAN in the Analyze options field and submit the VIARANOM analyze job. When the job finishes review DD VIARERPT, which shows the anomalies that have been found. The analyze job finishes and adds VIARANOM to the AKR.

- 3** Select File ► Anomaly facility to display the File - Anomaly Facility selection screen and select Batch Anomaly Repair to display the Anomaly Facility - Batch Submit pop-up. Enter the AKR dataset name and AKR program name fields (VIARANOM for the program name). Select the type of anomalies to repair from the Anomalies to be repaired section. Verify that the job information is correct and submit the job. When the job finishes, verify that there are no errors and review DD VIAANOUT to verify that it contains the repaired code. Verify that an ASG header has been generated at the top of the generated code and that the repaired code is commented out and new code has been added.
- 4** Using the generated repaired anomaly code, perform a normal Encore analyze with no anomaly options or parameters and verify that it analyzed correctly.
- 5** Perform the VIARANOM analyze again, but this time use the Output dataset name and Output member name fields to send the output results to an allocated PDS and member. Press PF4 and deselect the Comment out changed lines? and Generate ASG header? fields. Submit the job and verify that the repaired code was written to the specified PDS and member, the output does not contain an ASG header, and the fixed code is no longer commented out.
- 6** Perform the VIARANOM analyze one more time, sending the anomaly output to an allocated sequential file by filling in the Output dataset name field. Submit the analyze job and verify that the repaired code was written to the specified sequential file.
- 7** Select File ► Close and press Enter to close the VIARANOM program. Save any logic segments, if desired.

To verify anomaly report processing

Complete this section if you entered any anomaly report generation parameters in the Analyze options field on the File - Analyze Submit pop-up (see [Figure 8 on page 20](#)). This section verifies the VIARANOM anomaly report generation parameter processing discussed in ["To analyze the demonstration program" on page 20](#).

- ▶ Validate the anomaly report(s) using the JES Master output generated from the program analyze.

Note:

The VIARERPTAN parameter generates analysis anomaly reports. Sample generated reports are shown in [Figure 32](#) and [Figure 33](#). See the *ASG-Encore User's Guide* and *ASG-Encore Reference Guide* for more information about generating the anomaly reports.

Figure 32 • Dead Data (In Copy) Report

```

*****
*                               *
*          DEAD DATA (IN COPY) REPORT          *
*                               *
* This report lists the number of dead data (unreferenced data names) *
* found in each copy member. *
*****
NUMBER DEAD NAMES   FOUND IN COPY MEMBER
-----
          2         USER.HOLD.CNTL(MEMBER)
          35         CICS330.LIBRARY(MEMBER)

DD-MMM-YYYY  HH:MM:SS                                PROGRAM: VIARANOM

```

Figure 33 • Excessive Parameter List Report

```

*****
*                               *
*          EXCESSIVE PARAMETER LIST REPORT          *
*                               *
* This report lists all call statements using an excessive *
* number of parameters. The default number of parameters is 8. *
*****
LINE NUMBER   CALLS PROGRAM          NUMBER OF PARAMETERS
-----
          330   'DETRPT'                9
          363   'DETRPT'                9

DD-MMM-YYYY  HH:MM:SS                                PROGRAM: VIARANOM

```


- To ensure that you are selecting the code you want to extract, type V next to the CICS EXEC you want to view, as shown in [Figure 36](#). Press Enter to display the View Source pop-up (see [Figure 37](#)).

Figure 36 • CICS Server Endpoints Pop-up - View Option

```

-----
                                Extract - CICS Server Endpoints
Command ==>                                Scroll ==> CSR

Select the statement for each CICS Server Endpoint and press Enter to view
the corresponding Startpoints. This screen will be redisplayed to allow
additional Endpoints to be selected. Press Enter when all selections are
complete.

  S - Select      V - View Source      U - Unselect
-----
000481  EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000482          ERASE END-EXEC.
V 000615  EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000616          ERASE END-EXEC.
000652  EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW') END-EXEC.
000672  EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000673          ERASE END-EXEC.
_ 000694  EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000695          ERASE END-EXEC.
_ 000795  EXEC CICS SEND MAP('MENU') MAPSET('VIARMNU') ERASE END-EXEC.
***** BOTTOM OF DATA *****

```

- Verify that the CICS EXEC server code selected represents the type of code you want to extract. After verifying the code press PF3 to return to the CICS Server Endpoints pop-up (see [Figure 38 on page 47](#)). If necessary, repeat this process until you find the code you want to extract.

Figure 37 • CICS Server Extract View - Source Pop-up

```

-----
View Search Logic List Options Help
-----
                                View - Source      SCREEN POSITIONED
Command ==>                                Scroll ==> CSR
-----
000273  UPDATE-ACCT.                                PERF RNG
----- 1 LINE NOT DISPLAYED
000275  PERFORM INIT-ACTION-RECORD.                PERF RNG
----- 1 LINE NOT DISPLAYED
000277  IF ACCT-TYPE OF ACCT-RECORD = 'INT'        PERF RNG
000278      MOVE CHK-INTEREST-RATE TO INTEREST-RATE  PERF RNG
000279  ELSE IF ACCT-TYPE OF ACCT-RECORD = 'CHK'    PERF RNG
000280      MOVE ZERO TO INTEREST-RATE                PERF RNG
000281  ELSE IF ACCT-TYPE OF ACCT-RECORD = 'MMA'    PERF RNG
000282      MOVE MMA-INTEREST-RATE TO INTEREST-RATE  PERF RNG
000283  ELSE IF ACCT-TYPE OF ACCT-RECORD = 'SAV'    PERF RNG
000284      MOVE SAV-INTEREST-RATE TO INTEREST-RATE. PERF RNG
----- 1 LINE NOT DISPLAYED
000286  IF ACCT-MULTI-ACCT = 'YES'                    PERF RNG
000287      ADD CHK-MULTI-ACCT TO INTEREST-RATE        PERF RNG
000288      MOVE ACTION-MULTI-ACCT TO WORK-ACTION-REASON PERF RNG
000289      PERFORM INITIATE-ACTION.                    PERF RNG
-----

```

- 7 Type S next to the line you want to select, as shown in [Figure 38](#), and press Enter to display the CICS Server Startpoints pop-up (see [Figure 39](#)).

Figure 38 • CICS Server Endpoints Pop-up - Select Option

```

-----
                                Extract - CICS Server Endpoints
Command ==>                               Scroll ==> CSR

Select the statement for each CICS Server Endpoint and press Enter to view
the corresponding Startpoints. This screen will be redisplayed to allow
additional Endpoints to be selected. Press Enter when all selections are
complete.

  S - Select      V - View Source      U - Unselect
-----
000481    EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000482          ERASE END-EXEC.
S 000615    EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000616          ERASE END-EXEC.
- 000652    EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW') END-EXEC.
- 000672    EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000673          ERASE END-EXEC.
- 000694    EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000695          ERASE END-EXEC.
- 000735    EXEC CICS SEND MAP('MENU') MAPSET('VIARHNU') ERASE END-EXEC.
***** BOTTOM OF DATA *****
-----

```

- 8 Type S in the indicated command area or press Enter to accept the selection (all highlighted text is preselected). Press Enter to return to the CICS Server Endpoints pop-up (see [Figure 40 on page 48](#)).

Figure 39 • CICS Server Startpoints Pop-up

```

-----
                                CICS Server Startpoints
Command ==> _                               Scroll ==> CSR

Select statements for CICS Server Startpoints and press Enter. Press Enter
when selections are complete to return to the Endpoints screen.

Endpoint

  S - Select      V - View Source      U - Unselect
-----
000616    EXEC CICS RECEIVE MAP('BROWSE') MAPSET('VIARBRW')
000617          RESP(RESPDNSE) END-EXEC.
***** BOTTOM OF DATA *****
-----

```

- 9 The message displayed in the long message area confirms that the CICS Statements for the EXEC CICS SEND code were successfully selected, as shown in [Figure 40](#). Press Enter to display the CICS Server Paths pop-up (see [Figure 41 on page 49](#)).

Figure 40 • CICS Server Endpoints Pop-up - Selection Confirmation Message

```
Command ==> Extract - CICS Server Endpoints Scroll ==> CSR
Select the statement for each CICS Server Endpoint and press Enter to view
the corresponding Startpoints. This screen will be redisplayed to allow
additional Endpoints to be selected. Press Enter when all selections are
complete.

S - Select      V - View Source      U - Unselect
-----
- 000481      EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000482      ERASE END-EXEC.
- 000615      EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000616      ERASE END-EXEC.
- 000652      EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW') END-EXEC.
- 000672      EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000673      ERASE END-EXEC.
- 000694      EXEC CICS SEND MAP('BROWSE') MAPSET('VIARBRW')
000695      ERASE END-EXEC.
- 000735      EXEC CICS SEND MAP('MENU') MAPSET('VIARMNU') ERASE END-EXEC.
*****

ASG4102I CICS STATEMENTS SUCCESSFULLY SELECTED.
```

- 10 Default T (TRUE) or F (FALSE) selections have been made based on the selection criteria, as shown in [Figure 41](#). An X indicates that the code does not apply to this extract. You can change the settings, if desired.

Figure 41 • CICS Server Paths Pop-up

```

Extract - CICS Server Paths
Command ==> _ Scroll ==> CSR

Identify conditionals which are definitely TRUE for the Server, those which
are definitely FALSE, and leave blank the conditionals which may be either
TRUE or FALSE. Press Enter when all selections are complete.

  T - TRUE      F - FALSE      blank - TRUE/FALSE      V - View Source
-----
T 000298      IF EIBCALEN NOT = 0 THEN
X 000377      IF EIBCALEN = 0 THEN GO TO NEXT-LINE.
000508      IF EIBCALEN = 0 THEN GO TO TEST-STATS.
F 000634      IF EIBRID = DFHCLEAR THEN
F 000640      IF EIBRID = DFHPP1 OR DIRI = 'F' THEN GO TO PAGE-FORWARD.
T 000641      IF EIBRID = DFHPP2 OR DIRI = 'B' THEN GO TO PAGE-BACKWARD.
***** BOTTOM OF DATA *****

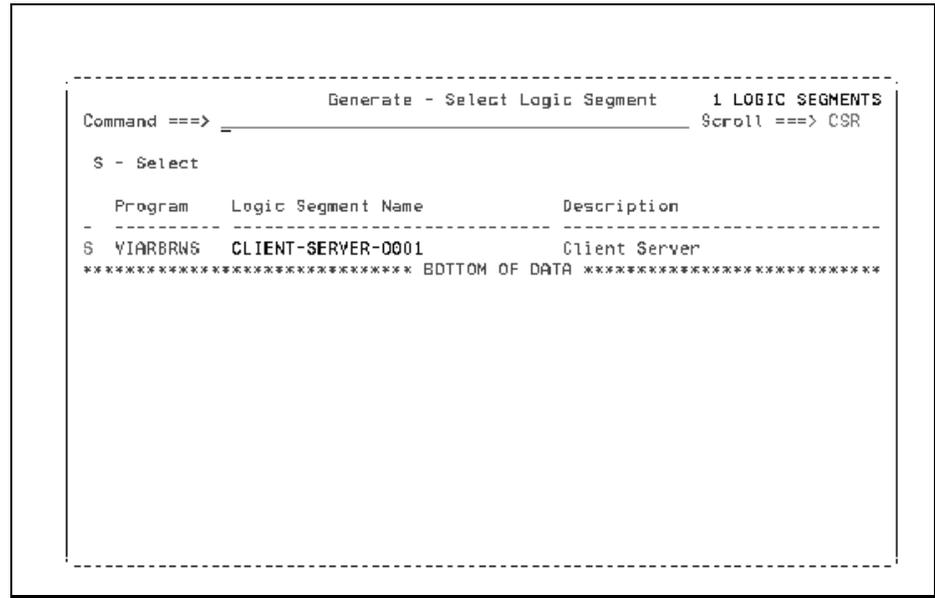
```

- 11 Press Enter to return to the Encore Primary screen, which displays a message indicating that the criteria for the statement extract was successfully selected.

To verify CICS server generation

- 1 Select Generate ► Server and press Enter to display the Generate - Select Logic Segment pop-up, as shown in [Figure 42](#).

Figure 42 • Generate - Select Logic Segment Pop-up



- 2 Type S next to the Logic Segment name you want to select, or accept the default selection, and press Enter to display the first Generate - HOST Copy Member pop-ups (see [Figure 43 on page 51](#)).

Note:

The number of Generate - HOST Copy Member pop-ups displayed during this procedure depends on the ending and starting statements selected on the corresponding CICS Server Endpoint (see [Figure 38 on page 47](#)) and CICS Server Startpoint (see [Figure 39 on page 47](#)) pop-ups. A minimum of two Generate - HOST Copy Member pop-ups always display for a generated server.

- 3 Type the new name you want to assign to the DFHCOMMAREA copy member in the Copy Member field (CPYCOMAR in this example, as shown in [Figure 43](#)). Assign the Starting Level Number and Level Number Increment you want or accept the defaults (CICS defaults to a record level of 5). Press Enter to display the second Generate - HOST Copy Member pop-up (see [Figure 44](#)).

Figure 43 • Generate - HOST Copy Member Pop-up Number 1

```

Generate - Select Logic Segment
D .----- ll ==> CSR
      Generate - HOST Copy Member      1 OF 2
Enter new COPY member information. Press Enter when
complete.
-
S Existing Copybook Information
* Copy Member . . . . . : CPYCOMAR
  Data Name . . . . . : DFHCOMMAREA

New Copybook Information
  Copy Member . . . . . : CPYCOMAR
  Starting Level Number 5
  Level Number Increment 5

```

- 4 Accept the new copy member defaults, or change the name and level numbers, and press Enter to display the Generate - HOST Server Module pop-up (see [Figure 45 on page 52](#)).

Figure 44 • Generate - HOST Copy Member Pop-up - Number 2

```

Generate - Select Logic Segment
D .----- ll ==> CSR
      Generate - HOST Copy Member      2 OF 2
Enter new COPY member information. Press Enter when
complete.
-
S Existing Copybook Information
* Copy Member . . . . . : VIABRWC
  Data Name . . . . . : BROWSEO

New Copybook Information
  Copy Member . . . . . : VIABRCS
  Starting Level Number 5
  Level Number Increment 5

```


- 9 Type CREATE on the command line and C 9999 on the first line of code, as shown in [Figure 49](#). Press Enter to display the Edit/View - Create pop-up to save the module in your source library (see [Figure 50](#)).

Figure 49 • Edit Screen

```

ASG-Encore EDIT - VIASHG.ENC00036.COBOL                COLUMNS 00001 00072
Command ==> CREATE                                     Scroll ==> CSR
***** ***** Top of Data *****
C 9999 -      05 RESPONSE PIC S9(8) COMP.
000002      05 ASG-DFHCOMM.
000003      *
000004      *
000005      10 STATS PIC X(1).
000006      10 RIDB PIC 9(6).
000007      10 RIDF PIC 9(6).
000008      05 ASG-DFHEIBVARS.
000009      10 ASG-EIBAID PIC X(1).
***** ***** Bottom of Data *****

ASG1608I ENTER THE CREATE COMMAND TO SAVE THE COBOL PROGRAM.
    
```

- 10 Type the name you want to save the member under in your source library, clear any data in the Data Set Name and Volume Serial fields, and press Enter to return to the Edit screen (see [Figure 51 on page 55](#)).

Figure 50 • Edit/View - Create Pop-up

```

                                Edit/View - Create
Command ==>

"Current" Data Set: USERID.ENC00033.COBOL

To ISPF Library:
Project . . . . ASG
Group . . . . CE60XXXX
Type . . . . CNTL
Member . . . . CPYCOMAR

To Other Partitioned Data Set Member:
Data Set Name . . .
Volume Serial . . . (If not cataloged)

Data Set Password . . (If password protected)

Enter "/" to select option
Specify pack option for "CREATE" Data Set

Press ENTER key to create. Enter END command to cancel create.
    
```

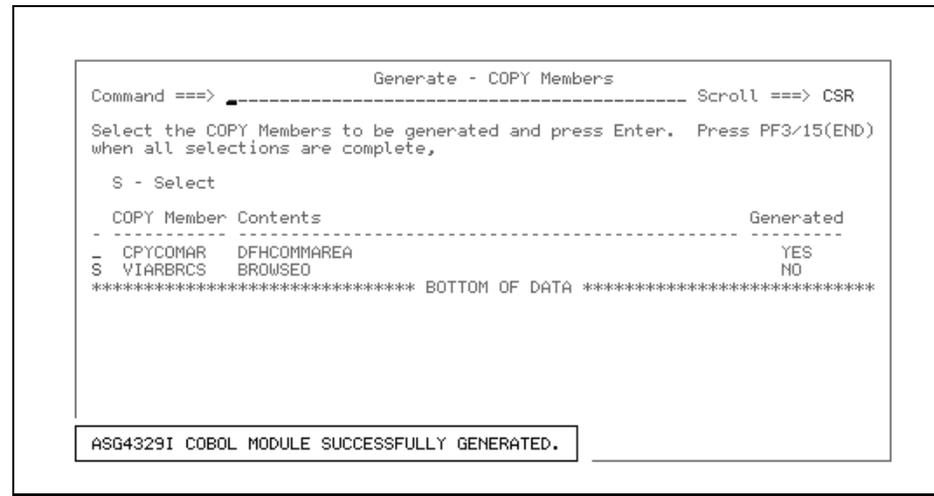
- 11 The Edit screen redisplay with a message in the upper right corner of the screen confirming that the member has been created, as shown in [Figure 51](#). Press PF3 to return to the Generate - Copy Members pop-up (see [Figure 52 on page 56](#)).

Figure 51 • Edit Screen

```
ASG-Encore EDIT - VIASHG.ENC00038.COBOL          Member CPYCOMAR created
Command ==>                                     Scroll ==> CSR
***** ***** Top of Data *****
000001      05 RESPONSE PIC S9(8) COMP.
000002      05 ASG-DFHCOMM.
000003      *                                     ASG-ENCORE - DFHCOMMAREA
000004      *
000005      10 STATS PIC X(1).
000006      10 RIDB PIC 9(6).
000007      10 RIDF PIC 9(6).
000008      05 ASG-DFHEIBVARS.
000009      10 ASG-EIBAID PIC X(1).
***** ***** Bottom of Data *****
```

A confirmation message displays in the long message area indicating that the COBOL module generated successfully, as shown in [Figure 52](#). The status also changes from NO to YES under the Generated column. Select the next Copy member and repeat [step 8 on page 53](#) through [step 11 on page 55](#) to edit and save the member.

Figure 52 • Generate - COPY Members Pop-up



- 12 Once saved, press Enter and then PF3 to return to the Generate - COPY Member pop-up.

After the member has been saved and you return to the Generate - COPY Members pop-up, a message displays in the long message area indicating that the COBOL module generated successfully, as shown in [Figure 53](#). The status also changes from NO to YES under the Generated column.

Figure 53 • Generate - COPY Members Pop-up

```

Command ==> _____ Generate - COPY Members _____ Scroll ==> CSR
Select the COPY Members to be generated and press Enter. Press PF3/15(END)
when all selections are complete,

S - Select

COPY Member Contents _____ Generated
- CPYCOMAR DFHCOMMAREA YES
- VIARBRCS BROWSEO YES
***** BOTTOM OF DATA *****

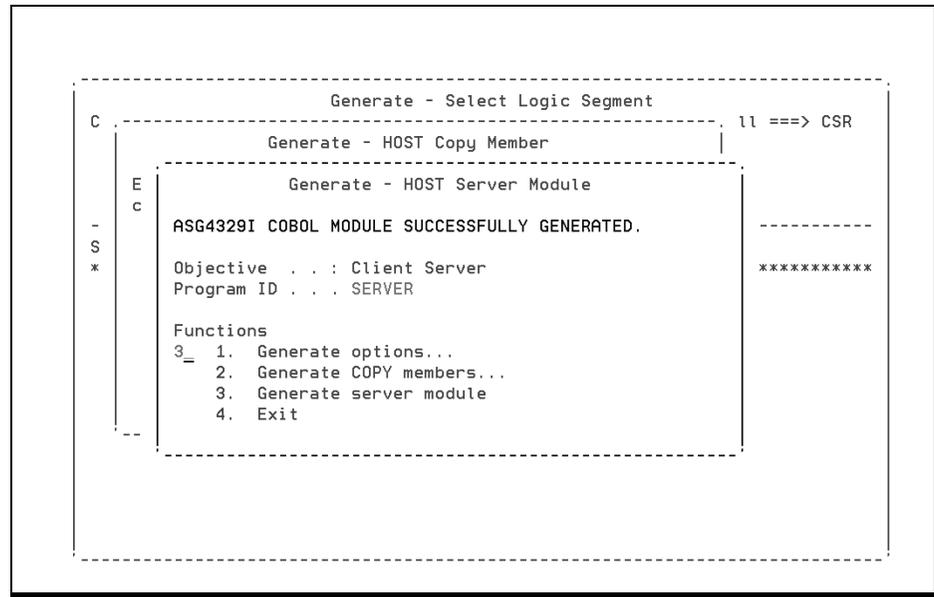
ASG4329I COBOL MODULE SUCCESSFULLY GENERATED.

```

- 13** Press PF3 to return to the Generate - HOST Server Module pop-up (see [Figure 54 on page 58](#)).

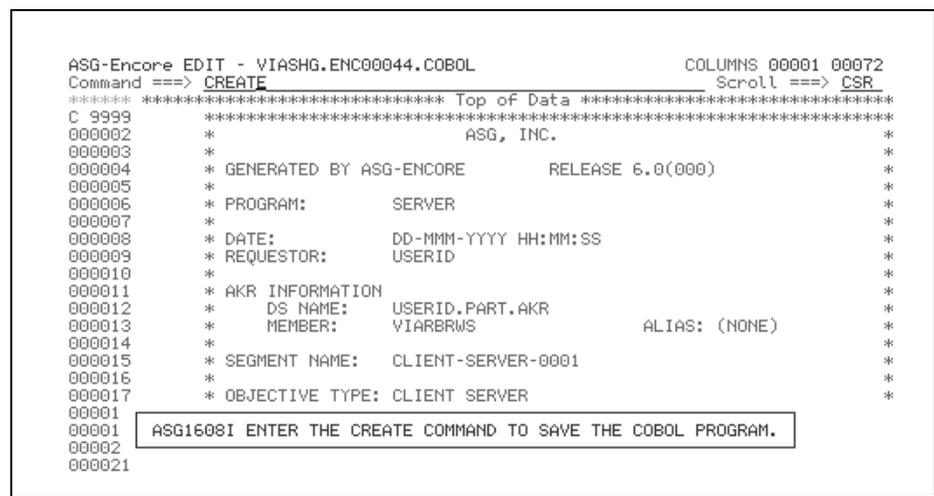
- 14 Type 3 to select Generate server module, as shown in [Figure 54](#) and press Enter to display the Edit screen (see [Figure 55](#)).

Figure 54 • Generate - HOST Server Module Pop-up



- 15 Type CREATE on the command line and C 9999 on the first line of code. Press Enter to display the Edit/View - Create pop-up to save the module in your source library (see [Figure 56 on page 59](#)).

Figure 55 • Edit Screen



- 16 Type the name you want to save the member under in your source library, as shown in [Figure 56](#), clear any data in the Data Set Name and Volume Serial fields, and press Enter to return to the Edit screen (see [Figure 57](#)).

Figure 56 • Edit/View - Create Pop-up

```

  Menu  RefList  Utilities  Help
  _____
  Edit/View - Create
  Command ==> _____
  "Current" Data Set: VIASHG.ENC00044.COBOL

  To ISPF Library:
  Project . . . USER
  Group . . . TEST
  Type . . . CNTL
  Member . . . SERVER

  To Other Partitioned Data Set Member:
  Data Set Name . . . _____
  Volume Serial . . . _____ (If not cataloged)

  Data Set Password . . . _____ (If password protected)

  Enter "/" to select option
  _ Specify pack option for "CREATE" Data Set

  Press ENTER key to create. Enter END command to cancel create.

```

- 17 The Edit screen redisplay with a message confirming that the member has been created. Press PF3 to return to the Generate - HOST Server Module pop-up (see [Figure 58 on page 60](#)).

Figure 57 • Edit Screen

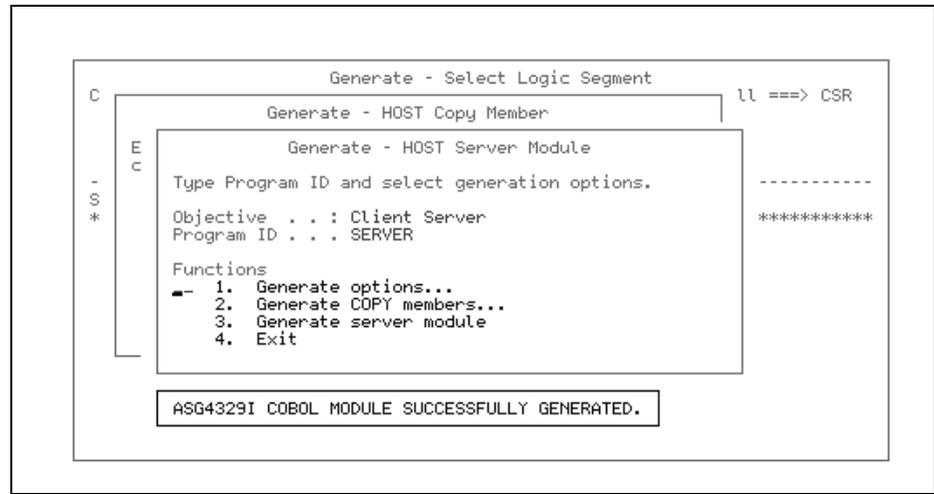
```

ASG-Encore EDIT - VIASHG.ENC00044.COBOL
Command ==> _____ Member: SERVER created
                               Scroll ==> CSR
***** ***** Top of Data *****
000001 *****
000002 *                               ASG, INC.                               *
000003 *                               *                                       *
000004 *   GENERATED BY ASG-ENCORE   RELEASE 6.0(000)   *
000005 *                               *                                       *
000006 *   PROGRAM:           SERVER                               *
000007 *                               *                                       *
000008 *   DATE:             DD-MMM-YYYY HH:MM:SS           *
000009 *   REQUESTOR:      USERID                             *
000010 *                               *                                       *
000011 *   AKR INFORMATION                               *
000012 *     DS NAME:     USERID.PART.AKR                   *
000013 *     MEMBER:     VIARBRWS                             ALIAS: (NONE) *
000014 *                               *                                       *
000015 *   SEGMENT NAME:  CLIENT-SERVER-0001                 *
000016 *                               *                                       *
000017 *   OBJECTIVE TYPE: CLIENT SERVER                       *
000018 *   IDENTIFICATION DIVISION.                            *
000019 *   PROGRAM-ID. SERVER.                                *
000020 *   ENVIRONMENT DIVISION.                               *
000021

```

- 18 Type 4 and press Enter to exit the Generate - HOST Server Module pop-up, as shown in [Figure 58](#), and return to the Encore Primary screen.

Figure 58 • Generate - HOST Server Module Pop-up



The customization and verification of Encore is complete.

Abend

Abnormal end of task.

action bar

The line of keywords at the top of a screen. Each keyword represents a category of actions that may be performed on that screen. An action is selected by moving the cursor to the desired keyword and pressing Enter.

AKR (Application Knowledge Repository)

A BDAM or VSAM file that contains all analysis information produced by the analyze function of the Application Analytical Engine.

AKR utility log

File that provides a summary of the commands issued to the Batch AKR Utility.

Alias Of

A field on a pop-up listing entries in the AKR. If the analyzed program contains an Entry point, Alias Of is the name of the program that contains the Entry point. If the name in the PROGRAM-ID statement was overridden at the time the analyze job was submitted, Alias Of is the name that was entered in the AKR program name field on the File - Analyze Submit pop-up.

analyze

The process by which Encore gathers information about the program, including program relationships, logic, data and execution paths, and stores this information in the AKR.

analyze options

Run-time options that control the Analyze processing. Many of these options are similar to the COBOL compiler options. Default values are established at installation time and can be overridden by editing the Analyzer JCL or by using the analyze screens.

analyze summary report

A summary of the run-time statistics and diagnostic messages that are produced when an analyze job completes.

anomalies

A deviation or departure from the normal Encore analysis process.

Anomaly Repair facility

An Encore facility that allows you to repair programs already analyzed in an AKR that contain anomalies. This is accomplished by allowing you to reanalyze the member that generated the anomaly condition using the File ► Anomaly facility pull-down option.

anomaly reports

Reports that contain information beneficial in understanding and diagnosing analysis problems and anomalies.

anchor

An HTML element that defines a link between Internet resources.

API

Application programming interface. A set of calling conventions defining how a service is invoked through a software package.

APPC

Advanced program-to-program communication. An implementation of the SNA LU 6.2 protocol that allows interconnected systems to communicate and share the processing of programs.

application

Any group of programs that a user wants to analyze/view as a whole. These programs would all be defined within the same application.

asynchronous

Without regular time relationship, unexpected or unpredictable with respect to the execution of program instructions. See ["synchronous" on page 72](#).

Batch AKR command

Control cards input in the VIASYSIN DD dataset of the Batch AKR Utility.

Batch AKR utility

The facility used to maintain the AKR without using ISPF.

browser

An application that displays World Wide Web documents.

business group

A logical grouping of applications by business function.

business rule

A logical business function that is performed by a COBOL program.

CERN

The Couseil Europeen pour la Recherche Nucleaire (European Particle Physics Laboratory), which developed hypertext technologies.

CICS server extract

A CICS server extract locates the necessary statements, files, and data elements required to create a CICS Server Extract program. This program is then used to create a COMMAREA-based server program from a CICS pseudo-conversational program.

client

As in client/server computing, the application that makes requests to the server and, often, handles the interaction necessary with the user.

client/server computing

A form of distributed processing, in which the task required to be processed is accomplished by a client portion that requests services and a server portion that fulfills those requests. The client and server remain transparent to each other in terms of location and platform. See [client](#) above and ["server" on page 71](#).

COBOL subset

COBOL verbs of a similar nature that have been grouped together. For example, READ, WRITE, OPEN, and CLOSE are grouped into the IO subset.

command

An option in a menu. A command is selected to perform an action. There are the three types of commands

- Primary commands
- Line commands
- Batch AKR utility commands

command input area

The field on Encore screens where primary commands are entered, indicated by ==> on the fourth line of a screen, or the second line of a pop-up.

commit

An action that an application takes to make permanent the changes it has made to CICS resources.

common code extract

The process of identifying the code that is common to two or more previously-extracted Logic Segments. This includes all statements that appear in a selected Logic Segment, as well as appearing in all other selected Logic Segments.

Common Gateway Interface (CGI)

The defined standard for the communications between HTTP servers and external executable programs.

Common User Access (CUA)

A style of graphical interface that features screens, actions bars, pull-downs, and pop-ups that are designed to provide easy access to all product features.

complement extract

The process of identifying a set of COBOL statements that do not include a previous extract. This includes all statements from the original program, excluding statements of the previously-extracted Logic Segment, but retaining any statements needed by other functional paths.

complement module

A complement module contains all statements from the original program, excluding statements of the extracted Logic Segment, but retaining any statements needed by other functional paths.

computation variable extract

The process of isolating the minimum set of statements that determine the value of a particular data variable (known as the computation variable) at a particular statement in the program.

cursor substitution character

A token substitution character that may be used in some primary commands. The cursor substitution character is set on the Options - Product Parameters pop-up.

conversational

A communication model where two distributed applications exchange information by way of a conversation. Typically, one application starts (or allocates) the conversation, sends some data, and allows the other application to send some data. Both applications continue in turn until one decides to finish (or deallocate). The conversational model is a synchronous form of communication.

criteria

The information entered that was used to isolate a Logic Segment.

cursor position

The current location of the cursor on the screen, used as a starting point in certain commands.

database

- A collection of interrelated data stored together with controlled redundancy according to a scheme to serve one or more applications.
- All data files stored in the system.
- A set of data stored together and managed by a database management system.

dataname

A standard COBOL term for fields defined in the DATA DIVISION of a COBOL program. Variable names, files, groups, array elements, and fully qualified datanames.

data usage

Defines how a data item is used.

- DEF indicates the statements in the DATA DIVISION where the data item is defined.
- USE indicates the statements where the value is used or tested.
- MOD indicates the statements where the value is set or modified.
- REF indicates any of the above conditions.

dead assignment

A statement that assigns a value to a variable, where the value is not referenced anywhere in the source code. The assignment is not necessary and may be removed from the code without affecting program execution.

delimiter

A character or sequence of characters used as a separator in text or data files.

Distributed Computing Environment (DCE)

Adopted by the computer industry as a de facto standard for distributed computing. DCE allows computers from a variety of vendors to communicate transparently and share resources such as computing power, files, printers, and other objects in the network.

distributed processing

An application or systems model in which function and data can be distributed across multiple computing resources connected on a LAN or WAN. See "[client/server computing](#)" on page 63.

Distributed Program Link (DPL)

Enables an application program executing in one CICS system to link (pass control) to a program in a different CICS system. The linked-to program executes and returns a result to the linking program. This process is equivalent to remote procedure calls (RPCs). You can write applications that issue RPCs that can be received by members of the CICS family.

Distributed Transaction Processing (DTP)

Enables a transaction running in one CICS system to communicate synchronously with transactions running in other systems. The transactions are designed and coded specifically to communicate with each other. This method is typically used by banks, for example in just-in-time stock replacement.

Double Byte Character Set (DBCS)

A character set that uses two bytes to represent each character. Various Double Byte Character Sets are used with languages such as Chinese and Japanese that cannot be represented with single byte codes.

environment

The collective hardware and software configuration of a system.

equate

A substitution name for a character string.

External Call Interface (ECI)

An application programming interface (API) that enables a non-CICS client application to call a CICS program as a subroutine. The client application communicates with the server CICS program using a data area called COMMAREA.

External Presentation Interface (EPI)

An application programming interface (API) that allows a non-CICS application program to appear to the CICS system as one or more standard 3270 terminals. The non-CICS application can start CICS transactions and send and receive standard 3270 data streams to those transactions.

File Transfer Protocol (FTP)

A protocol that defines how to transfer files from one computer to another.

forms

Parts of HTML documents that allow users to enter data.

function shipping

Enables an application program running in one CICS system to access resources owned by another CICS system. In the resource-owning system, a transaction is initiated to perform the necessary operation. For example, to access CICS files or temporary storage, and to reply to the requester. The user is unaware of these behind-the-scenes activities, and need not know where the resource actually exists.

gateway

Software that transfers data between normally incompatible applications, or between networks.

gopher

Menu-based software for exploring Internet resources.

Graphic Interchange Format (GFI)

256-color graphic format.

help

Encore provides these three levels of help:

- Long messages
- Notes
- Tutorial screens

Specific command information is available by entering a command, then pressing PF01/13. The Help facility can also be accessed from the Help pull-down or any Encore screen.

home page

The default page shown at the first connection to an HTTP server.

host

- In a computer network a computer providing services such as computation, database access, and network control functions.
- In a multiple computer installation, the primary or controlling computer.

hypertext

Text that activates connection to other documents when selected.

Hypertext Markup Language (HTML)

Standard language used to create hypertext documents.

Hypertext Transmission Protocol (HTTP)

Standard WWW client/server communications protocol.

Internet Keyed Payment Protocol (iKP)

Proposed protocol for conducting secure commercial financial transactions on the Internet.

intercommunication

Communication between separate systems by means of Systems Network Architecture (SNA), Transmission Control Protocol/Internet Protocol (TCP/IP), and Network Basic Input/Output System (NetBIOS) networking facilities.

internet

A collection of networks.

label name

Any PROCEDURE DIVISION paragraph or section name and the PROCEDURE and PROC literals. Label name specifies all transfers of control to a paragraph or section.

line command

An instruction entered in the line command area on certain screens.

list box

A dialog box option containing a list of items the user can select.

list file

The file that is allocated when a request to print is issued.

live exit

An abnormality in program control caused by out of perform range GO TOs and overlapping perform ranges.

log file

The file that is allocated by Encore and used for error messages and log commands. There is a separate log file created by the Batch AKR utility.

logic segment

A set of source statements that is the result of a perform range extract, report extract, computation variable extract, transaction extract, statement extract, complement extract, server extract, or common code extract. The criteria used to isolate a Logic Segment may be saved in the AKR.

logic segment complement

See ["complement module" on page 64](#).

logical program unit

A PERFORMed range of code including GO TO code, or a CALLEd program.

Logical Unit of Work (LUW)

An update that durably transforms a resource from one consistent state to another consistent state. A sequence of processing actions (for example, database changes) that must be completed before any of the individual actions can be regarded as committed. When changes are committed (by successful completion of the LUW and recording of the synch point on the system log), they do not need to be backed out after a subsequent error within the task or region. The end of an LUW is marked in a transaction by a synch point that is issued by either the user program or the CICS server, at the end of task. If there are no user synch points, the entire task is an LUW.

long message

A diagnostic or error message that is displayed on line five of a screen or line three of a pop-up. Long messages are sometimes preceded by short messages that are displayed in the upper right corner of the screen. Pressing PF01/PF13 (HELP) after receiving a short message displays the corresponding long message.

LU type 6.2 (LU 6.2)

A type of logical unit used for CICS intersystem communication (ISC). LU 6.2 architecture supports CICS host-to-system-level products and CICS host-to-device-level products. APPC is the protocol boundary of the LU 6.2 architecture.

markup tag

Special character sequences put in text used to pass information to a tool, such as a document formatter.

member

A member in a PDS or source manager such as Panvalet or Librarian. This can be the alias name found in the AKR.

menu

A list of available commands in an application window. Menu names appear in the menu bar of the application window.

menu bar

A bar that contains the menus available for your use.

message box

A box that displays a message (error or otherwise) to inform the user of a particular condition.

Multipurpose Internet Mail Extension (MIME)

The Internet standard for mail that supports text, images, audio, and video.

NCSA Mosaic

A web browser available on multiple platforms.

online help

See ["help" on page 67](#).

Online Transaction Processing (OLTP)

A style of computing that supports interactive applications in which requests submitted by terminal users are processed as soon as they are received. Results are returned to the requester in a relatively short period of time. An online transaction processing system supervises the sharing of resources to allow efficient processing of multiple transactions at the same time.

paragraph name

Any PROCEDURE DIVISION paragraph or section name, and the PROCEDURE and PROC literals. Paragraph name includes the entire paragraph or section.

partial statement

In a generated COBOL module, a statement that has been modified to omit certain datanames or file names that are not required for that statement in the generated module. The partial statement contains only that portion of the statement required for the generated module.

perform range

A perform range consists of the source code contained in a PERFORM statement, and includes all code that is or could be executed as a result of GO TOs, PERFORMs, etc. within that PERFORM.

perform range extract

The process of isolating all executable statements of a perform range. This includes not only the statements within the range, but also all code that is reachable using GO TO or PERFORM from within the range.

pop-up

A window that appears as the result of selecting an item on a pull-down or pop-up, or as the result of entering certain commands. It is superimposed on the screen to allow entry of information for the requested action.

postscript

The standard for presenting text and graphics in a device-independent format.

primary command

An instruction entered in the command input area of the screen.

program

Program source member name, the name specified in the IDENTIFICATION DIVISION of a COBOL program, or the CSECT name of a program that is not COBOL.

protocol

- A formal set of conventions governing the format and control of data.
- A set of procedures or rules for establishing and controlling transmissions from a source device or process to a target device or process.

proxy

A gateway that allows Web browsers to pass on a network request (a URL) to an outside agent.

pseudo conversational

A type of CICS application design that appears to the user as a continuous conversation, but consists internally of multiple tasks.

pull-down

The list that appears when an action is selected on the action bar. On a pull-down, an action followed by ellipses (...) displays a pop-up when selected and an action not followed by ellipsis (...) immediately activates internal commands.

punch file

The file that is allocated when a request to punch is issued.

recovery

The use of archived copies to reconstruct files, databases, or complete disk images after they are lost or destroyed.

recoverable resources

Items whose integrity CICS maintains in the event of a system error. These include individual files and queues.

recursion

A perform range or paragraph that performs itself.

re-engineering

The process of renovating an existing program to isolate and extract distinct functions. Encore can isolate and extract code based on a perform range, report, computation variable, transaction, or server.

report extract

The process of isolating all WRITE/GENERATE statements for a specific File Description (FD) or Report Description (RD). Individual statements may be selected. The extracted unit consists of the statements required to produce the WRITE/GENERATE statement(s).

SBCS

See "[Single Byte Character Set - \(SBCS\)](#)" on page 72.

screen

A full-width display of information containing an action bar as the first line. Encore screens are modeled after TSO/ISPF screens.

screen subset

The result of an interactive command.

script

An executable program invoked by HTTP servers.

script file

A dataset containing Encore commands, created when SET SCRIPT is ON, and executed with the EXECUTE primary command or the File ► Execute script pull-down option.

server

Any computing resource dedicated to responding to client requests. Servers can be linked to clients through LANs and WANs to perform services (such as printing, database access, fax, and image processing) on behalf of multiple clients at the same time.

server extract

Process used to create a COMMAREA-based server program from a 3270 CICS pseudo-conversational program.

shortcut key

A keyboard key or combination of keys that invokes a particular command, such as CNTL + N.

short message

A diagnostic or error message that is displayed in the upper right corner of Encore screens. Pressing PF1/PF13 (HELP) after receiving a short message displays the corresponding long message.

Single Byte Character Set - (SBCS)

A character set that uses one byte to represent each character. Single Byte Character Sets are used with languages, such as English, where the characters can be represented with a one-byte code.

Socket Secure (SOCKS)

The gateway that allows compliant client code (client code made socket secure) to establish a session with a remote host.

Standard Generalized Markup Language (SGML)

The standard that defines several markup languages, HTML included.

statement extract

The process of isolating all user-identified statements from a program.

status bar

The area at the bottom of a main window that lists the status of an action and gives other information, such as the meaning of a command.

Storage Management Subsystem (SMS)

An operating environment that automates and centralizes the management of storage. To manage storage, SMS provides the storage administrator with control over data class, storage class, management class, storage group, and ACS routine definitions.

subset

A grouping of source lines in a program. See ["COBOL subset" on page 63](#), ["screen subset" on page 71](#), and ["tagged line subset" on page 73](#).

synchronous

- Pertaining to two or more processes that depend on the occurrence of a specific event such as a common timing signal.
- Occurring with a regular or predictable time relationship.

synchpoint

A logical point in execution of an application program where the changes made to the databases by the program are consistent and complete and can be committed to the database. The output, which has been held up to that point, is sent to its destination. The input is removed from the message queues and the database updates are made available to other applications. When a program terminates abnormally, CICS recovery and restart facilities do not back out updates prior to the last completed synchpoint.

tagged line subset

Command results displayed in columns 73 through 80 of the Source View screen.

target

The object of a primary command.

transaction

A unit of processing (consisting of one or more application programs) initiated by a single request. A transaction can require the initiation of one or more tasks for its execution.

transaction extract

The process of isolating all conditional statements for a specific transaction control variable. Certain program paths may then be selected or blocked, based on the transaction code value.

transaction processing

A style of computing that supports interactive applications in which requests submitted by users are processed as soon as they are received. Results are returned to the requester in a relatively short period of time. A transaction processing system supervises the sharing of resources for processing multiple transactions at the same time.

transaction routing

Enables a terminal connected to one CICS system to run a transaction in another CICS system. It is common for CICS/ESA, CICS/VSE, and CICS/MVS users to have a terminal-owning region (TOR) that owns end-user network resources.

VIASUB

An edit macro included with Encore that is used to submit an Analyze job.

VIASUBDS

A CLIST included with Encore that is used to submit an Analyze job.

view method

The manner in which a program is examined in the View facility. These are the three view methods based on source code:

- Structure view
- Tree view
- Source view

All three views are available concurrently when a program is examined.

work file

A temporary file allocated upon entry to Encore.

Appendix A

Encore CNTL and CLIST Members

Encore CNTL Members

This table contains Encore CNTL members.

Member	Description
VIA\$PRMA	The Encore specific default installation options.
VIA\$RLSA	Backup of VIA\$PRMA.
VIARANOM	COBOL demonstration program source for anomaly reporting and repair.
VIARCBSK	Generated module header box template.
VIAACMP1	JCL to compile and analyze VIARDEMO demonstration program.
VIAACMP2	JCL to compile and analyze VIARDRIV and VIARSVBR demonstration programs.
VIAACMP3	JCL to compile and analyze VIARANOM demonstration programs.
VIAACMP4	JCL to compile and analyze VIARBRWS demonstration programs.
VIARBRWC	Copybook for CICS demonstration program source.
VIARBRWS	CICS demonstration program source.
VIARBRW	Map for CICS demonstration program Main menu panel.
VIARCPY1	COBOL demonstration program copy member.
VIARCPY2	COBOL demonstration program copy member.
VIARCSFP	Description of COBOL/SF library+ allocations.

Member	Description
VIARDEMO	COBOL demonstration program source.
VIARDRIV	COBOL demonstration program source.
VIARIOBK	PARM block used to communicate with an IO module.
VIARIOFC	IO module function code constants.
VIARIOSK	Generated COBOL program template.
VIARMNU	Map for CICS demonstration program Browse menu panel.
VIARMNUC	Copybook for CICS demonstration program Browse menu panel.
VIARNTSK	Generated COBOL program message template.
VIARSUBR	COBOL demonstration program source.

Encore CLIST Members

This table contains Encore CLIST members.

Member	Description
ENCORE	Invokes the Encore product from the ISPF panel.
VIAACSF	Invokes the COBOL/SF ECFSTART command.
VIAACSF1	CLIST for COBOL/SF to Encore interface.
VIAACSTM	CLIST for customizing Encore.
VIAENC	Invokes the Encore product.
VIAATEST	CLIST to establish a TSO test session with Encore.

A

AKR, allocation 18
Alliance
 accessing from ESW screen ix
 description vi
 linking ix
Anomaly Repair Facility
 online panel-driven example 41
 parameter-driven example 40
Anomaly Repair Facility parameters
 ANOMALY 21
 COMMENT 21
 DSN 22
 GTD 21
 HEADER 22
 IPR 21
 MEM 22
 OOPGT 21
 PARM 21
 VIAANOMF 21
anomaly report parameters
 NUMPRMA(##) 21
 RPTCEN 21
 VIARERPT 21
 VIARERPTAN 21
AutoChange
 accessing from ESW screen ix
 description vi

B

Bridge
 accessing from ESW screen ix
 description vi

C

CENTER parameter 8, 11–12
Center, description vi
CICS parameter 9
CICS server
 CICS EXEC 46
 CICS server endpoints 46
 CICS server paths 48

CICS server startpoints 47
 copy member field 51
 CREATE command 54
 endpoints 45
 level number increment 51
 self-directed server name 45
 starting level number 51
 statement extract 49
CICS server extract 63
CICS server extract - CICS statements 48
CICS server extract - EXEC CICS SEND 48
CICS server extracts 4
CICSCOB parameter 9
CICSLIB parameter 9
CICSTRAN parameter 9
CLIST member
 ENCORE 14, 76
 VIAACSF 76
 VIAACSF1 76
 VIAATEST 76
 VIACSTM 76
 VIAENC 76
CNTL member
 VIA\$PRMA 13, 75
 VIA\$RLSA 75
 VIAACMP1 21, 75
 VIAACMP2 21, 75
 VIAACMP3 21, 75
 VIAACMP4 21, 75
 VIAACSF 13
 VIARANOM 75
 VIARBRW 75
 VIARBRWC 75
 VIARBRWS 75
 VIARCBSK 13, 75
 VIARCPY1 75
 VIARCPY2 75
 VIARCSFP 13, 75
 VIARDEMO 76
 VIARDRIV 76
 VIARIOBK 76
 VIARIOFC 76

- VIARIOSK 76
- VIARNTSK 76
- VIARSUBR 76
- VIASLPJ 10
- VIASLPXJ 10
- COBCOMP parameter 8
- COBOL 5
- COBOL/SF, interfacing with Encore 13
- common code extract 63
- compile and link JCL
 - VIARANOM program 21
 - VIARBRWS program 21
 - VIARDEMO program 21
 - VIARDRIV program 21
 - VIARSUBR program 21
- compilers
 - COBOL D, E, and F 5
 - COBOL for MVS and VM 5
 - COBOL for OS/390 and VM 5
 - COBOL II 5
 - COBOL/370 5
 - Object Oriented conditions not supported 5
- COMPILR parameter 8
- complement extract 64
- computation variable extract 64
- conventions page xii
- customizing, module header template 13

D

- DB2, validation 40
- demonstration programs
 - VIARANOM 8
 - VIARBRWS 8
 - VIARDEMO 8
 - VIARDRIV 8
 - VIARSUBR 8

E

- Encore
 - accessing from ESW screen ix
 - description vii
 - functional components 2
 - invoking from ISPF 14
 - invoking with CLIST 14
 - invoking with TSO VIAENC CLIST 14
 - migration 4
- ENCORE CLIST member 14, 76
- Encore installation
 - CENTER parameter 8
 - CICS parameter 9
 - CICSCOB parameter 9

- CICSLIB parameter 9
- CICSTRAN parameter 9
- CNTL member VIAACMP1 8
- CNTL member VIAACMP2 8
- CNTL member VIAACMP3 8
- CNTL member VIAACMP4 8
- CNTL member VIARCSFP 13
- CNTL members to modify and execute 8
- COBCOMP parameter 8
- COBOL/SF interface set up 13
- COMPILR parameter 8
- NEWAPPL 14
- SYSDA parameter 8
- SYSOOT parameter 8
- TRNPARM parameter 9
- UNIQUE 19
- validating parameter definitions 16
- validation 15
- VIA\$PRMA COBOLSF option 9
- VIA\$PRMA export option 9
- VIA\$PRMA
 - Gen-COBOL-Last-Node= option 9
- VIA\$PRMA modification 9
- VIA\$PRMA NETRON option 9
- VIAACMP3 parameters 9
- VIAACMP4 parameters 9
- VIAACSF CLIST member 13
- VIASOFT parameter 8

environment

- client - Windows 95/98/2000 5
- client - Windows NT with access to MVS 5
- host - 3270 type terminals 5
- host - BATCH region size 5
- host - Direct Access storage 5
- host - MVS ISPF 5
- host - MVS/XA, MVS/ESA, or OS/390 5
- host - TSO logon region size 5
- host - VSAM 5
- platform and software requirements 5
- server - OS/390 5

Estimate

- accessing from ESW screen ix
- description vii

ESW

- description v
- invoking products viii
- product integration ix

I

- IDMS, validation 39

- Insight
 - accessing from ESW screen [ix](#)
 - description [vii](#)
 - using analysis functions [ix](#)
 - working with [25](#)
- installation prerequisites, Center [7](#)
- ISPF
 - adding Encore [14](#)
 - menu option [14](#)
- L**
- Librarian customization
 - ASMBLR parameter [12](#)
 - CENTER parameter [12](#)
 - customized edit panel [12](#)
 - ISPLLIB [12](#)
 - LIBDEF [12](#)
 - LIBDEF facility [12](#)
 - LIBRLIB parameter [12](#)
 - LIBRMAC parameter [12](#)
 - load library [12](#)
 - SYSDA parameter [12](#)
 - SYSOUT parameter [12](#)
 - VIALLIB [12](#)
 - VIASFAIJM module [12](#)
 - VIASFAIR member [12](#)
 - VIASFAIR module [12](#)
 - VIASOFT parameter [12](#)
 - VSPEDLIB panel [12](#)
 - WTP messages [12](#)
- M**
- menu option, ISPF [14](#)
- MLPA
 - adding Encore modules [10](#)
 - improving performance [10](#)
 - staging library [10](#)
 - VIAENC module addition [10](#)
 - VIARCMMP module addition [10](#)
 - VIARCSFI module addition [10](#)
 - VIARCSFM module addition [10](#)
 - VIARCSOM module addition [10](#)
 - VIARICSF module addition [10](#)
 - VIARISPP module addition [10](#)
 - VIARWKSP module addition [10](#)
 - VIASLPXJ CNTL member [10](#)
 - VIAZBTCH module addition [10](#)
 - VIAZCMDP module addition [10](#)
 - VIAZGENP module addition [10](#)
 - VIAZPRIP module addition [10](#)
 - VIAZTBLP module addition [10](#)
- MPLA, VIASLPAJ CNTL member [10](#)
- N**
- NEWAPPL parameter [14](#)
- O**
- overview, target generation [4](#)
- P**
- Panvalet customization
 - ASMBLR parameter [11](#)
 - CENTER parameter [11](#)
 - customized edit panel [11](#)
 - ISPLLIB [11](#)
 - LIBDEF [11](#)
 - LIBDEF facility [11](#)
 - load library [11](#)
 - PANLIB parameter [11](#)
 - SYSDA parameter [11](#)
 - SYSOUT parameter [11](#)
 - VIALLIB [11](#)
 - VIASOFT parameter [11](#)
 - VIASPAM module [11](#)
 - VIASPAMJ CNTL member [11](#)
 - VSPEDPAN panel [11](#)
 - WTP messages [11](#)
- parameter, UNIQUE [19](#)
- perform range extract [70](#)
- PLPA
 - adding Encore modules [10](#)
 - improving performance [10](#)
 - staging library [10](#)
 - VIAENC module addition [10](#)
 - VIARCMMP module addition [10](#)
 - VIARCSFI module addition [10](#)
 - VIARCSFM module addition [10](#)
 - VIARCSOM module addition [10](#)
 - VIARICSF module addition [10](#)
 - VIARISPP module addition [10](#)
 - VIARWKSP module addition [10](#)
 - VIASLPAJ CNTL member [10](#)
 - VIASLPXJ CNTL member [10](#)
 - VIAZBTCH module addition [10](#)
 - VIAZCMDP module addition [10](#)
 - VIAZGENP module addition [10](#)
 - VIAZPRIP module addition [10](#)
 - VIAZTBLP module addition [10](#)
- preprocessor support
 - command level CICS [6](#)
 - command level DL/I [6](#)
 - IDMS [6](#)
 - other preprocessed languages [6](#)
 - SQL [6](#)
- product integration [ix](#)
- PTFs, applying maintenance [10](#)

R

- Recap
 - accessing from ESW screen [ix](#)
 - description [vii](#)
- report extract
 - description [71](#)
 - extracting the objective [44](#)

S

- server extract [71](#)
- SmartDoc
 - accessing from ESW screen [ix](#)
 - description [vii](#)
- SmartEdit
 - accessing from ESW screen [ix](#)
 - description [viii](#)
- SmartTest
 - accessing from ESW screen [ix](#)
 - description [viii](#)
- statement extract [72](#)
- SYSDA parameter [8](#)
- SYSOUT parameter [8](#)

T

- target generation, overview [4](#)
- transaction extract [73](#)
- TRNPARAM parameter [9](#)

U

- UNIQUE parameter [19](#)

V

- VIA\$PRMA CNTL member [75](#)
- VIA\$RLSA CNTL member [75](#)
- VIAACMP1 CNTL member [8, 21, 75](#)
- VIAACMP2 CNTL member [8, 21, 75](#)
- VIAACMP3 CNTL member [8, 21, 75](#)
- VIAACMP4 CNTL member [8, 21, 75](#)
- VIAACSF CLIST member [76](#)
- VIAACSF CNTL member [13](#)
- VIAACSF1 CLIST member [76](#)
- VIAATEST CLIST member [76](#)
- VIAACSTM CLIST member [76](#)
- VIAENC CLIST member [14, 76](#)
- VIARANOM CNTL member [75](#)
- VIARANOM demonstration program [8](#)
- VIARBRW CNTL member [75](#)
- VIARBRWC CNTL member [75](#)
- VIARBRWS CICS demonstration program [8](#)
- VIARBRWS CNTL member [75](#)
- VIARCBSK CNTL member [75](#)
- VIARCPY1 CNTL member [75](#)

- VIARCPY2 CNTL member [75](#)
- VIARCSFP CNTL member [13, 75](#)
- VIARDEMO CNTL member [76](#)
- VIARDEMO demonstration program [8](#)
- VIARDRIV CNTL member [76](#)
- VIARDRIV demonstration program [8](#)
- VIARIOBK CNTL member [76](#)
- VIARIOFC CNTL member [76](#)
- VIARIOSK CNTL member [76](#)
- VIARNTSK CNTL member [76](#)
- VIARSUBR CNTL member [76](#)
- VIARSUBR demonstration program [8](#)
- VIASFAIR module [12](#)
- VIASLPAJ CNTL member [10](#)
- VIASLPXJ CNTL member [10](#)
- VIASOFT parameter [8](#)

ASG Worldwide Headquarters Naples Florida USA | asg.com