

# FOCUS<sup>®</sup> for S/390<sup>®</sup>

MVS/TSO Installation Guide  
Version 7.0

# Contents

---

---

1.	FEATURES AND COMPONENTS.....	1-1
1.1	Mainframes and Operating Systems .....	1-1
1.2	Installed Features.....	1-1
1.2.1	The Basic FOCUS System .....	1-2
1.2.2	High Resolution Graphics on IBM 3270 Type Terminals .....	1-3
1.2.3	The FOCUS/ICU Interface.....	1-3
1.2.4	Link Edited User Exits .....	1-3
1.2.5	Changes to FOCUS Defaults.....	1-4
1.2.6	The IBI Subsystem vs. the FOCUS SVC .....	1-4
	Installation Requirements .....	1-4
	Features and Functionality .....	1-5
	Capacity and Performance.....	1-5
	Troubleshooting and Maintenance .....	1-6
1.2.7	Introduction to the IBI Subsystem.....	1-6
	Overview .....	1-6
	Benefits.....	1-7
1.2.8	MVS Subsystems .....	1-7
	Subsystems .....	1-7
	The Subsystem Interface .....	1-7
	Cross-Memory Services .....	1-8
1.2.9	The IBI Subsystem .....	1-8
	Information Builders Subsystem-Based Products .....	1-8
	Benefits of Using the Subsystem.....	1-9
1.2.10	Technical Overview.....	1-9
	Communication Datasets.....	1-10
1.2.11	The FOCUS SVC .....	1-10
1.2.12	Simultaneous Usage (SU).....	1-11
1.3	The FOCUS Libraries .....	1-11
1.3.1	FOCLIB.LOAD.....	1-12
1.3.2	FUSELIB.LOAD.....	1-13
1.3.3	FUSELIB.DATA.....	1-13
1.3.4	ERRORS.DATA .....	1-13
1.3.5	FOCCTL.DATA.....	1-14
1.3.6	MASTER.DATA and FOCEXEC.DATA.....	1-14
1.3.7	WINFORMS.DATA .....	1-14
1.4	Dataset Naming Conventions.....	1-14
1.5	Virtual Storage Requirements .....	1-15
1.5.1	FOCUS Storage Requirements.....	1-16
1.5.2	Storage Requirements for the Simultaneous Usage (SU) facility.....	1-16

2.	BASIC INSTALLATION STEPS .....	2-1
2.1	Allocate Disk Space for FOCUS Libraries .....	2-1
	Example 1: CLIST to Allocate Disk Space .....	2-4
	Example 2: JCL to Allocate Disk Space .....	2-5
2.2	Unload the FOCUS Distribution Tape .....	2-6
2.3	Install the FOCUS Site Code .....	2-10
2.4	Select FOCUS Default Startup Options .....	2-11
2.5	Test the Basic FOCUS System.....	2-16
2.5.1	Logon into a TSO region .....	2-16
2.5.2	Allocate space for a test FOCUS database .....	2-16
2.5.3	Execute the TSOALLOC CLIST .....	2-16
2.5.4	Execute the FOCUS test procedure CARTEST .....	2-16
	Sample JCL for Executing CARTEST as a Batch Job .....	2-17
2.5.5	Online Release Information.....	2-17
2.6	Install Link-Edited User Exits and Change FOCUS Defaults .....	2-18
2.7	Install the Reentrant Modules .....	2-18
2.8	Reblocking FOCUS Modules .....	2-19
2.9	Install High-Resolution Graphics for 3270 Type Terminals.....	2-21
2.10	Installing the IBI Subsystem .....	2-22
2.10.1	Installation Planning.....	2-22
	Installation .....	2-22
	Capacity and Performance.....	2-22
	Training .....	2-22
2.10.2	Installation .....	2-23
	Installation Steps .....	2-23
	Product-Related Installation Issues .....	2-23
	Maintenance Issues.....	2-25
2.10.3	IBI Subsystem Usage .....	2-26
	The SUBSYSI Program.....	2-26
	Operator Controls .....	2-27
2.10.4	Troubleshooting.....	2-28
	Diagnostic information.....	2-29
2.11	Install the FOCUS SVC .....	2-30
2.11.1	Select a user SVC number in the allowed user range.....	2-30
2.11.2	Modify the FOCUS SVC global variables. ....	2-30
2.11.3	Assemble and link edit the modified source code. ....	2-31
2.11.4	Post the FOCUS SVC number into zap control statements.....	2-33
2.11.5	Zap the FOCUS SVC number into the basic FOCUS system.....	2-33
2.11.6	Test use of the FOCUS SVC by the zapped modules. ....	2-33

2.12	Install the Simultaneous Usage (SU) Facility .....	2-35
2.12.1	Allocate the communication dataset FOCUSU .....	2-35
2.12.2	Create the auxiliary FOCUS database FOCUSU. ....	2-36
2.12.3	Installing the FOCUS/SU Security Interface.....	2-36
	Using the Interface .....	2-37
2.13	TSO Logon Procedures and Allocation CLISTS.....	2-38
3.	Customizing FOCUS .....	3-1
3.1	Changes to FOCUS Defaults.....	3-1
3.1.1	Changing the Number of Active FOCUS Databases Allowed.....	3-1
3.1.2	Increased Field Buffer Size .....	3-1
	Installation .....	3-1
3.1.3	Setting the Maximum Number of Cache Pages.....	3-3
	Installation .....	3-4
3.1.4	Default Space Allocation Table for Work Files: TABLA.....	3-6
	TABLA Installation:.....	3-7
3.1.5	Default System File Naming Conventions .....	3-7
	Modifying Naming Conventions for Automatic Allocation: DSNTAB.....	3-8
	DSNTAB Installation: .....	3-11
3.1.6	Modifying the Efficiency List of TSO Commands in FOCUS: TSOTAB.....	3-12
	TSOTAB Installation.....	3-12
3.1.7	Resetting the Maximum Allowed Data Exceptions .....	3-12
3.2	Installing HiperFOCUS for MVS/ESA.....	3-14
3.2.1	Pre-Installation Requirements .....	3-14
3.2.2	HiperFOCUS Installation .....	3-14
3.2.3	HiperBUDGET and the IBI SUBSYSTEM .....	3-16
3.3	Installing the FOCUS Menu.....	3-18
3.4	Installing the FOCUS ToolKit .....	3-20
3.5	Enabling National Language Support .....	3-27
4.	User Exits .....	4-1
4.1	Usage Accounting Exit: UACCT.....	4-1
4.1.1	UACCT Calling Sequence .....	4-1
4.1.2	Usage Statistics Calls (CALLTYPE=0) .....	4-3
4.1.3	DBA Violation Calls (CALLTYPE=1).....	4-5
4.1.4	Start of Processing Calls (CALLTYPE=2) .....	4-5
4.1.5	External Security Violations.....	4-6
4.1.6	UACCT Installation.....	4-6
4.2	Dataset Naming Enforcement (DSNCHK) .....	4-7
	DSNCHK Installation:.....	4-8
4.3	User-Coded Password Security Interface.....	4-9
4.3.1	FOCUSID Calling Sequence.....	4-9
4.3.2	Installation .....	4-10

5.	Interface Installation.....	5-1
5.1	Installing the ICU Interface.....	5-1
5.2	Installing and Using an External Password Security System.....	5-3
5.2.1	The FOCUS-Supplied ACF2 Interface .....	5-3
5.2.2	Usage.....	5-3
	Default FOCUS Userids.....	5-4
	Imposed FOCUS Userids .....	5-4
	Unidentified FOCUS Users.....	5-5
5.2.3	Installation .....	5-5
5.3	Installing the FOCCALC Graphics Facility.....	5-6
5.3.1	Installation .....	5-6
6.	MAINTENANCE .....	6-1
6.1	FOCUS Release Numbering .....	6-1
6.2	The Release Signature.....	6-1
6.3	Test and Production Releases.....	6-1
6.4	Corrections to Reported Problems .....	6-2
6.5	The Release Memos .....	6-2
6.6	Release Tapes.....	6-2
6.7	Load Libraries and Maintenance Libraries .....	6-3
6.8	How to Apply an Individual PTF.....	6-4
6.9	The FOCMAP Interface.....	6-5
A.	Common Mistakes During FOCUS Installation .....	A-1
B.	Sample JCL for the Central Database Job.....	B-1
C.	Subsystem Error Messages.....	C-1
D.	Subsystem Sample JCL and Zaps .....	D-1
D.1	SUBSYSZP: Zap to Activate the IBI Subsystem.....	D-2
D.2	SUBSYSNM: Zap to Change the Subsystem Name.....	D-3
D.3	SUBSYSCP: JCL to Copy SUBSYSI Into a Separate Library .....	D-4
D.4	SUBSYSIJ: JCL to run SUBSYSI .....	D-5
D.5	SUBSYSIH: JCL to Run SUBSYSI and Initialize HiperBUDGET .....	D-6
D.6	SUBSYSP: Sample Cataloged Procedure.....	D-7

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. In most, if not all cases, these designations are claimed as trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

FOCUS® and TableTalk® are registered trademarks of Information Builders, Inc.

SmartMode™ is a trademark of Information Builders, Inc.

Information Builders™ is a trademark of Information Builders, Inc.

IBM®, IMS/ESA®, VTAM®, and DB2® are registered trademarks of International Business Machines Corporation.

RACF™, MVS/ESA™, SQL/DS™, CICS™, DRDA™ and QMF™ are trademarks of International Business Machines Corporation.

CA-IDMS/DB®, CA-ACF2®, CA-TOP SECRET®, and CA-DATACOM/DB® are registered trademarks of Computer Associates, Inc.

MODEL 204® is a registered trademark of Praxis International, Inc.

SUPRA® and TOTAL® are registered trademarks of Cincom Systems, Inc.

Oracle® is a registered trademark of the Oracle Corporation.

Teradata® is a registered trademark of Teradata Corporation.

Copyright © 1997, by Information Builders, Inc. All rights reserved. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

## Preface

---

This is the installation guide for Release 7.0 of the MVS/TSO version of FOCUS. It defines the MVS/TSO configurations required for the operation and maintenance of FOCUS, and takes you through the installation process step by step. It is written for the MVS/TSO user support group, not for the community of FOCUS end users. You will find that FOCUS is very easy to install because it is an application-level product, and every effort has been made to keep the installation procedures simple. A working copy of FOCUS can be installed with only a user-level knowledge of MVS/TSO and these instructions, without any prior knowledge of FOCUS.

## How This Manual Is Organized

---

The following table describes the chapters and appendixes included in this manual.

<b>Chapters/ Appendixes</b>	<b>Title</b>	<b>Description</b>
<b>1</b>	<i>Features and Components</i>	A brief description of the components which can be installed and the environment required for their execution.
<b>2</b>	<i>Basic Installation</i>	Contains the step-by-step installation procedures.
<b>3</b>	<i>Customizing FOCUS</i>	Describes various customization issues.
<b>4</b>	<i>User Exits</i>	Describes the installation of User Exits.
<b>5</b>	<i>Interface Installation</i>	Describes the installation of various interfaces.
<b>6</b>	<i>Maintenance</i>	Deals with various maintenance topics.
<b>A</b>	<i>Common Mistakes During FOCUS installation</i>	Describes some common mistakes made during FOCUS installation.
<b>B</b>	<i>Sample JCL for the Central Database Job</i>	Shows sample JCL for the central database job.
<b>C</b>	<i>Subsystem Error Messages</i>	Shows Subsystem error messages.
<b>D</b>	<i>Subsystem Sample JCL and Zaps</i>	Shows Subsystem Sample JCL and zaps.

Each FOCUS 7.0 release tape comes with a specific version of the *FOCUS MVS/TSO Installation Guide*, identified by the date on the cover. A new version of the installation guide is published only when additions or corrections are needed, and it may not change with each release or version.

## Documentation Conventions

---

The following conventions are used throughout this manual:

UPPERCASE	Denotes a term that you must enter exactly as shown.
<i>lowercase italics</i>	Denotes a term that you must substitute with an appropriate value.
{ }	Indicates a group of required parameters, one of which you must select.
[ ]	Indicates a group of optional parameters, one of which you can select.
<u>underscore</u>	Indicates a default setting.
...	Indicates that you can enter a parameter multiple times.
. . . .	Indicates that there are (or could be) intervening or additional commands.
Master File Description	Used throughout this manual instead of "Master File," which is the comparable term used in EDA documentation.

## Related Publications

---

See the *FOCUS for IBM Mainframe User's Manual* (DN1001005.0495) and the *Summary of New Features for 7.0.\* CD* (DN1001050.0298).

Refer to the *Information Builders Publications Catalog* (DN9400022.0498) for the most up-to-date listing and price of publications, plus ordering information. To obtain a catalog, contact the Publications Order Department at 1-800-969-4636 or visit the bookstore on our World Wide Web site (<http://www.ibi.com/bookstore>).

## User Feedback

---

In an effort to produce effective documentation, the Documentation Services staff at Information Builders welcomes any opinion you can offer regarding this manual. Please use the Reader Comments form at the end of this manual to relay suggestions for improving the publication or to alert us to corrections.

Thank you, in advance, for your comments.



## Information Builders Customer Support Services

---

Call Information Builders Customer Support Service (CSS) at (800) 736-6130 or (212) 736-6130 or your local Information Builders representative. Customer Support Consultants are available between 8:00 a.m. and 8:00 p.m. EST to address all your FOCUS for IBM Mainframe Installations questions.

You can also upload your questions to the FOCWIZARD forum on CompuServe. We do not accept questions via FAX.

For general guidance on IBI product capabilities and documentation, you can speak with a Customer Support Consultant, or visit our World Wide Web site (<http://www.ibi.com>).

## Information You Should Have

---

To help our consultants answer your questions most effectively, be ready to provide the following information when you call:

- Your six-digit site code number (xxxx.xx).
- The release of FOCUS that you are using.
- The release of your operating system.
- The exact nature of the problem:
  - Are the results or is the format incorrect; is the text or calculations missing or misplaced?
  - The error message and return code, if applicable.
  - Is this related to any other problem?
- Has the procedure ever worked in its present form? Has it been changed recently? How often does the problem occur?
- Is this problem reproducible? If so, how?
- Have you tried to reproduce your problem in the simplest form possible?
- How is the problem affecting your business? Is it halting development? Do you just have questions about functionality or documentation?

## **Information Builders Consulting and Training**

---

Interested in training? Information Builders Education Department offers a wide variety of training courses for this and other Information Builders products. For more information, course descriptions, locations, and dates, or to register for classes, visit the Information Builders Education Emporium at our World Wide Web site (<http://www.ibi.com/pshome.htm>), or call 1-800-969-INFO to speak to an Education Representative.

# 1 FEATURES AND COMPONENTS

---

---

This chapter describes the installation options and the MVS configurations required for the installation and execution of FOCUS. The actual installation steps are detailed in Chapter 2.

## 1.1 Mainframes and Operating Systems

---

The MVS/TSO version of FOCUS Release 7.0 runs on any IBM or plug compatible computer that supports MVS/XA or later. Please contact the IBI Installation Services Department in New York for details.

FOCUS is configured to take maximum advantage of IBM's OS/390, using the Extended Link Pack Area (XLPA). Installing the reentrant part of FOCUS in XLPA allows more FOCUS code to be shared among users and requires a smaller logon region size.

This guide details the installation of FOCUS under MVS/XA and MVS/ESA.

## 1.2 Installed Features

---

In order to make the installation as easy as possible, most commonly used FOCUS features supported under MVS are delivered fully enabled and are installed as part of the basic system. The installation of a few optional features requires additional steps, also covered in this manual, but the installation of major interfaces is covered in separate installation guides.

This manual covers the installation of the following components:

- The basic FOCUS system.
- High resolution graphics on IBM 3270 type terminals, along with the ICU Interface.
- Link edited user exits.
- Changes to FOCUS defaults.
- Reentrant modules.
- The IBI Subsystem.
- The FOCUS SVC.
- Simultaneous Usage (SU) for simultaneous update of FOCUS databases.

The installation of major database interfaces and the following FOCUS products is covered in separate manuals:

ADABAS	IDMS/SQL	MSO*
DATACOM	IMS/DB	ORACLE
DB2	IMS/DC	SYSTEM 2000 (S2K)
FOCCALC	MILLENIUM	TERADATA
IDMS	MODEL 204	TOTAL

- \* The FOCUS Multi-Session Option (MSO) is a FOCUS operating environment that runs under either MVS/XA, MVS/ESA, or OS/390 providing complete FOCUS functionality and enabling multiple users to share access services in a single MVS address space. Refer to the *FOCUS for IBM Mainframe Multi-Session Option Installation and Technical Reference Guide Release 7.0* for installation instructions.

The basic FOCUS system contains all the FOCUS features needed for the majority of all applications and its installation requires only a user level knowledge of common MVS utilities. All other installation steps are optional and they can be deferred until you have installed the basic system and have become familiar with it.

## 1.2.1 The Basic FOCUS System

---

From an installation standpoint, the basic FOCUS system consisting of many FOCUS features is installed in one step. The basic system contains everything except graphics support for IBM 3270 type terminals, interfaces to complex databases created by other DBMS systems, and Simultaneous Usage (SU), the facility that allows several concurrent FOCUS users to update the same FOCUS database at the same time. It contains all the FOCUS data management functions, including FIDEL (the FOCUS full screen data entry facility), and the entire FOCUS query language and report writer. The report writer works with FOCUS databases, and also with sequential, ISAM, and VSAM files. The basic system also includes high resolution graphics for a limited set of ASCII graphics terminals, but not for IBM 3270 type terminals.

The basic system is a prerequisite for the use of all other FOCUS features. FOCUS is sensitive to its environment and the basic system runs under TSO, or as an MVS batch job. It can also run under CICS, MSO, or IMS/DC, if you install separate front-end interfaces.

To install the basic system, you need only unload the FOCUS distribution tape and create a TSO logon procedure or batch JCL. It is that simple. Then you will be ready to test almost all FOCUS features that you will use.

## 1.2.2 High Resolution Graphics on IBM 3270 Type Terminals

---

FOCUS high resolution graphics on IBM 3278, 3279 and compatible terminals require the assistance of the IBM GDDM graphics support program. From the standpoint of GDDM, FOCUS is just another application level program and the success of its 3270 type graphics depends largely on the GDDM support at your site. GDDM has complex hardware and software requirements, and if you intend to use it with FOCUS, you should make sure that it is properly installed. You can test it by writing a COBOL program that creates 3270 type graphics. Consult the appropriate manuals for the hardware and software configurations required by IBM.

To include this feature into FOCUS you simply link edit a GDDM stub from the GDDM library into the basic FOCUS system. This is a simple step and it can be done any time after the installation of the basic system.

## 1.2.3 The FOCUS/ICU Interface

---

FOCUS can be used to generate graphs in conjunction with IBM's Interactive Chart Utility (ICU). The normal FOCUS GRAPH syntax is used, and all of its capabilities are available. The ICU Interface can either place the user directly in the ICU environment or can save the graph format and data for subsequent ICU processing. To use this Interface, FOCUS must be installed with the "High Resolution Graphics for 3270 Type Terminals" option described in Section 1.2.2 .

See Chapter 5 for more information.

## 1.2.4 Link Edited User Exits

---

Several user exits can be link edited into FOCUS:

- The Usage Accounting Exit (UACCT).

The Usage Accounting Exit can be used to capture FOCUS usage statistics, such as the number of records retrieved and lines printed, aggregated at the FOCUS command level rather than at the job step or TSO session levels. The Usage Accounting Exit will be alerted to log attempted violations of FOCUS security, and it can also be used to limit access to FOCUS procedures and files.

- An External Password Security System (FOCUSID).

The user exit FOCUSID enables an external security system to set passwords automatically. With this exit, users need not know the password for an application.

- The Dataset Naming Enforcement Exit (DSNCHK).

DSNCHK can be used to enforce strict dataset naming conventions for FOCUS MASTER files, FOCEXEC files, ERROR files, and FOCUS data files.

The distributed version of FOCUS contains dummy routines that resolve the exit calls. If you are not interested in these exits simply do nothing. They are independent of one another and can always be added later on, through a simple link edit.

Detailed information on these exits is available in Chapter 4.

## 1.2.5 Changes to FOCUS Defaults

---

Installation options are available to increase the field buffer size and to change the default allocation table, default naming convention and the maximum allowable data exceptions. These changes to FOCUS defaults must be made at installation time. See Chapter 3 for detailed information.

## 1.2.6 The IBI Subsystem vs. the FOCUS SVC

---

Following is a comparison of the FOCUS SVC and the IBI Subsystem. This comparison will help you decide which to install for use with FOCUS.

### Installation Requirements

---

The FOCUS SVC installation requires:

- A SVC must be installed on MVS.
- An IPL of MVS is required to activate the new SVC.
- A zap job must be run to insert the proper SVC number in the FOCUS libraries.

The IBI Subsystem installation requires:

- A zap job must be run to activate the IBI Subsystem and disable the use of the FOCUS SVC.
- If the subsystem will be run on MVS 5.2 or later, an entry in the MVS System Name table must be reserved for use by the IBI subsystem. An IPL will be required if a new entry is added. An IPL is not required if an existing entry is used or if the MVS system is release 5.1 or earlier.
- An APF-authorized library is required for the subsystem modules.
- SYS1.PARMLIB must be changed. An IPL of MVS is required to activate the changes.

- The subsystem must be activated by running the SUBSYSI utility program.

## Features and Functionality

---

The FOCUS SVC provides the following feature:

- Basic inter-address space communications, for SU, MSO, and the IMS BMP.

The IBI subsystem provides the following features:

- MSO Load Balancing for release 7.0.5 or later.
- Basic inter-address space communications, for SU, MSO, and the IMS BMP.
- System-wide regulation of hiperspace utilization, through the HiperBUDGET feature of HiperFOCUS.
- Use of cross-memory services for inter-address space communications.
- Operator controls, through system commands and through the SUBSYSI utility program.
- Better cleanup of CSA storage at task and address space failure.

## Capacity and Performance

---

With the FOCUS SVC:

- An 8K storage area, or “mailbox”, is allocated in XCSA for each server communication path. There is one mailbox per SU or IMS BMP address space, and there are two mailboxes per MSO address space.
- The SVC uses ENQ/DEQ for serialization.

With the IBI Subsystem:

- 20K of XCSA is used to hold the subsystem modules and data. If the modules and data are deleted (by using the REMOVE option of SUBSYSI), 2K of XCSA is still used to hold items which may not be deleted. This storage is reclaimed if the subsystem is reactivated.
- If a server is running and is swappable, then a mailbox must be allocated in XCSA, as with the SVC.
- The subsystem uses Compare and Swap for serialization, which is faster than the SVC’s use of ENQ/DEQ.
- The subsystem uses a PC routine and cross-memory services for moving data, as long as the server address space is non-swappable. This is somewhat faster than the SVC’s technique for moving data.

- The subsystem enables the HiperBUDGET feature of HiperFOCUS, which allows for system-wide regulation of expanded storage usage by IBI products.

## Troubleshooting and Maintenance

---

With the FOCUS SVC:

- The SVCTEST program is used to verify correct installation of the SVC.
- The SVC is generally compatible across Release levels and FOCUS releases. If using the Simultaneous Usage facility of FOCUS, sink and source releases are compatible starting with FOCUS Release 6.5.7 Put Level 9301 up through the current 7.0 Release.
- An IPL of MVS is required to install a new version of the SVC or to make any changes to such parameters as the SVC number or storage keys.

With the IBI Subsystem:

- The SUBSYSI utility enables the installation of a new version of the subsystem without requiring an IPL of the system.
- Operator commands allow basic troubleshooting of the subsystem.
- The subsystem must be updated whenever specified when installing a new release of FOCUS.

## 1.2.7 Introduction to the IBI Subsystem

### Overview

---

The IBI Subsystem is a state-of-the-art facility which provides communications between and coordination among address spaces running Information Builders products on MVS systems.

The IBI Subsystem can replace the IBI SVC, providing additional functionality, easier installation and maintenance, and enhanced ease of use.

The IBI Subsystem is often referred to as “the subsystem” in this document.



## Benefits

---

Benefits of the IBI Subsystem include:

- Faster, simplified installation of FOCUS and MSO
- Expanded storage control
- Enhanced error recovery capability
- New control facilities
- Improved performance and capacity

## 1.2.8 MVS Subsystems

---

This section provides a brief overview of MVS Subsystems and related technology. For further information, refer to the appropriate IBM documentation, including *MVS/ESA System Modifications*.

### Subsystems

---

A subsystem is any program which receives requests through the subsystem interface. Many MVS-based products, both from IBM and other vendors, use the subsystem interface.

Products which run as subsystems include JES2 and JES3 (which run as the primary, as opposed to secondary, subsystem), SMS, DB2, IMS, NetView, Oracle, Omegamon, and Model 204.

Up to 32,767 subsystems may be defined to MVS at one time. It is common to have many subsystems running simultaneously on an MVS system.

A subsystem may run in its own address space, or it may run in the address space of the program which is requesting its services. In the latter case, the subsystem's modules typically reside in common storage.

The impact of a subsystem on an MVS system depends on the subsystem; the overhead of using the subsystem facility is negligible.

### The Subsystem Interface

---

The subsystem interface, or SSI, is used by MVS and programs to make requests of subsystems or to pass information to subsystems. MVS uses the SSI to inform subsystems of particular events, such as the issuing of an operator command or the starting or ending of a task. Programs may use the SSI to pass information to subsystems or to request services from them.

FOCUS uses the SSI to communicate with the IBI Subsystem.

## Cross-Memory Services

---

Cross-memory services is an MVS facility which is typically used in conjunction with a subsystem. Cross-memory services allows a program to:

- Access data in another address space.
- Run in another address space, but under the same TCB.
- Communicate easily between address spaces.

## 1.2.9 The IBI Subsystem

### Information Builders Subsystem-Based Products

---

The following Information Builders products use the IBI Subsystem:

- HiperFOCUS

The HiperBUDGET feature of HiperFOCUS uses the IBI Subsystem to regulate use of expanded storage on a system-wide basis.

- Multi-Session Option (MSO)

MSO uses the IBI Subsystem to communicate with CICS and TSO, and to improve its error recovery capability.

MSO Load Balancing is implemented using the IBI Subsystem.

- Simultaneous Usage (SU)

SU uses the IBI Subsystem to communicate between FOCUS users and sink machines.

- SmartMode

SmartMode, which is an SU-based service, uses the IBI Subsystem by virtue of SU using it.

- IMS BMP

The IMS BMP uses the IBI Subsystem to communicate between FOCUS users and the BMP address space.

## Benefits of Using the Subsystem

---

The IBI Subsystem provides the following benefits:

- Faster, simplified installation of FOCUS and MSO:

The FOCUS SVC no longer needs to be installed, reducing the complexity of the installation procedure.

An MVS IPL may not be required to install the subsystem. It is not required to update the subsystem.

- Expanded storage control:

The HiperBudget facility of HiperFOCUS allows coordination of usage of MVS hiperspaces across all address spaces running IBI products, thus allowing comprehensive enforcement of installation policies governing hiperspace usage.

- Enhanced error recovery capability:

By intercepting task-end and memory termination events, error recovery is enhanced. The subsystem can always perform cleanup when any IBI product terminates.

- New control facilities:

Display status from the MVS system console.

Control via the MVS system console or batch jobs.

- Improved performance and capacity:

Cross-memory services can be used to transfer data across address spaces, eliminating the need for the CSA “mailbox” used by the FOCUS SVC.

Use of ENQ and DEQ has been replaced by use of the Compare and Swap instruction.

## 1.2.10 Technical Overview

---

The IBI Subsystem runs in the address space of the IBI product which is using its services. All of the modules used by the subsystem are reentrant and pageable, and reside in the Extended Common Storage Area (XCSA). In addition, all control blocks created by the subsystem reside in the XCSA and are also pageable.

The subsystem uses 20K of XCSA for programs and control blocks, almost all of which is taken by programs.

The subsystem is loaded and initialized by the SUBSYSI utility, which must be run from an APF-authorized library. SUBSYSI can also be used to stop, replace, and/or unload the subsystem modules.

When used for communications between address spaces, the subsystem can use cross-memory services only when the server address space (MSO server or SU sink machine) is non-swappable. If cross-memory services can not be used, then an 8K “mailbox” is allocated in XCSA for each communication path used by the server. MSO address spaces use two mailboxes, for a total of 16K; SU address spaces use one mailbox. The subsystem code is run whenever any of the following events occur:

- An IBI product requests subsystem services.
- An operator command is issued with the subsystem name as the first four characters of the command.
- A task-end or memterm event occurs in an IBI address space.
- The SUBSYSI utility program is run.

## Communication Datasets

---

The communication datasets, (FOCSU, FOCUSU) which were used with the FOCUS SVC are still used with the IBI Subsystem.

Since the datasets are never actually opened, you might need to take action to prevent HSM or similar products from migrating them off-line.

### 1.2.11 The FOCUS SVC

---

The FOCUS SVC supports the following features:

- Simultaneous Usage for simultaneous database update of FOCUS databases.
- The FOCUS Multi Session Option (MSO) communications.
- Access from TSO to on-line IMS databases controlled by IMS/DC (the BMP extension of the FOCUS/IMS DB interface).
- SmartMode

The installation of an SVC requires the assistance of the MVS support group. The FOCUS SVC is not required for ordinary TSO or batch usage.

The FOCUS SVC is not required to access IMS databases if the users operate via the FOCUS IMS/DC interface. It is required only if TSO and IMS/DC run on the same CPU, the IMS databases are allocated as OLD to IMS/DC, and users want to access these on-line databases from TSO terminals. The FOCUS SVC is not required when FOCUS runs in IMS batch mode or in BMP mode.

The FOCUS SVC is designed only for the MVS operating system and is not supported under MVT, VS1 or SVS. The same SVC can be used under MVS/XA or MVS/ESA.

NOTE: If the FOCUS SVC was installed prior to Release 6.0, that SVC cannot be used with Release 7.0 of FOCUS. In this case, you must replace the existing FOCUS SVC with the SVC from the FOCUS Release 7.0 tape (unless the SVC from a Release 6.0 tape was previously installed). This replacement SVC will work with all FOCUS releases, under MVS/XA and MVS/ESA.

If you are not sure that you need this SVC, call your local IBI representative or the Installation Services Department in New York for further clarification. The New York office can be reached at 212-736-4433.

### 1.2.12 Simultaneous Usage (SU)

---

Simultaneous Usage allows many concurrent TSO users or batch jobs to update the same FOCUS database simultaneously. Database I/O for all users is performed by one central database job, having exclusive control of the database. It serializes I/O at the logical record level, rather than at the database level. With this feature, users can update a FOCUS database at any time, without being concerned about whether another user is also updating it. Without this feature, all users must allocate FOCUS databases in their own address space, and their updates are serialized by means of enqueues on the entire database. If several users attempt to do updates, one may be forced to wait or may have to try the update later.

You should be aware that even without this feature, it is safe to allocate all FOCUS databases in SHR mode because the FOCUS serialization will prevent destructive updates by concurrent on-line users. However, DISP=SHR will not protect against destructive updates by a concurrent Host Language Interface (HLI) program. If you must have simultaneous updates by concurrent HLI programs, or even by one HLI program concurrent with on-line TSO users, you must use the central database facility.

The central database job that performs the FOCUS database I/O can service all users on the same CPU.

Simultaneous Usage requires the FOCUS SVC or IBI Subsystem.

## 1.3 The FOCUS Libraries

---

All the components whose installation is described in this manual are distributed in seven partitioned datasets:

FOCLIB.LOAD	The FOCUS load library.
FUSELIB.LOAD	Load library of user written routines.
FUSELIB.DATA	The FUSELIB maintenance library
ERRORS.DATA	Text of FOCUS error messages.
FOCTL.DATA	The FOCUS maintenance library.
MASTER.DATA	Library of sample file descriptions.
FOCEXEC.DATA	Library of sample FOCUS procedures.
WINFORMS.DATA	Library sample MAINTAIN procedures

Only the FOCLIB.LOAD and the ERRORS.DATA libraries are required for execution. The use of FUSELIB.LOAD is optional, and the other four libraries are used only for testing and maintenance. The space requirements and DCB attributes are shown in Chapter 2, which describes the installation.

### 1.3.1 FOCLIB.LOAD

---

The FOCLIB.LOAD library contains all the load modules needed to execute the basic FOCUS system. It contains both reentrant and nonreentrant modules, and the main load module, named FOCUS. All modules are fully resolved and ready for execution without further action on your part.

Some of the modules have aliases, and it is critical that all names and aliases be retained if you move them from one load library to another. All aliases have the same entry point as the membername of the corresponding module. Should you lose an alias during a copy or a move operation, you can always reassign it with linkage editor. Sample JCL is:

```
//LINK      EXEC PGM=IEWL,PARM='NCAL,LIST,LET,RENT,SIZE=1024K'
//SYSLMOD  DD   DISP=SHR,DSN=prefix.FOCLIB.LOAD
//MAINTAIN DD   DISP=SHR,DSN=prefix.FOCCTL.DATA
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSPRINT DD   SYSOUT=*
//SYSLIN   DD   *

INCLUDE SYSLMOD(F$ROOT) < Reentrant module
INCLUDE MAINTAIN(F$ROOT) < Cntl for F$ROOT will assign alias FSROOT
NAME     F$ROOT(R)      < Output module

/*
```

### 1.3.2 FUSELIB.LOAD

---

The FUSELIB.LOAD library contains a number of small load modules that implement user written functions that can be invoked from FOCUS computational expressions. They are additions to the FOCUS computational facilities, contributed by the community of FOCUS users. We recommend that you make this library available to all of your users.

The functions contained in the FUSELIB.LOAD library are documented in the *User Written Subroutines Library* manual available from Information Builders.

### 1.3.3 FUSELIB.DATA

---

The FUSELIB.DATA library contains members which have the linkage editor control statements for the FUSELIB subroutines.

This library is only required when applying FOCUS maintenance to the user written subroutines.

### 1.3.4 ERRORS.DATA

---

The ERRORS.DATA library contains the text of all error messages generated by the basic FOCUS system. This library is required for FOCUS Release 7.0. If the ERRORS PDS is not allocated the following error is displayed:

```
(FOC000) UNABLE TO LOCATED ERRORS FILE, FOCUS PROCESSING IS UNPREDICTABLE.
```

There are several members, corresponding to major functional areas, and additional members will be created if you also install one of the major interfaces, such as the FOCUS/IMS DB interface. An ERRORS.DATA library must be allocated to the ddname ERRORS whenever FOCUS is started.

The members of this partitioned dataset are not numbered sequentially, and if you copy members from one PDS to another, be sure to specify NONUM. Do not add or delete records from the error messages files; FOCUS error messages are accessed by relative record number, not by the displayed error number.

### 1.3.5 FOCCTL.DATA

---

The FOCCTL.DATA dataset is called the maintenance library because, among other things, it contains the linkage editor control statements needed to maintain the load modules in FOCLIB.LOAD. It also contains all the components installed using this manual, but that are not installed as part of the basic FOCUS system, and examples of batch JCL and TSO CLISTS. FOCCTL.DATA is an essential dataset even though it is not needed for executing FOCUS. It contains the linkage editor control statements for each member of FOCLIB.LOAD, under the same membername.

NOTE: Each version of the FOCCTL.DATA dataset differs significantly from prior versions, and you cannot use the maintenance library of one release to modify the load library of another release.

The load modules in FOCLIB.LOAD and their linkage editor control statements in FOCCTL.DATA are associated via a common membername. For example, the ORDER statements for the reentrant module F\$ROOT are contained in member F\$ROOT of FOCCTL.DATA. The linkage editor control statements always specify the entry point name, which is not necessarily the same as the load module membername.

### 1.3.6 MASTER.DATA and FOCEXEC.DATA

---

These two datasets contain file descriptions (MASTER) and procedures (FOCEXEC) used to test FOCUS after it has been installed, and all the components of the FOCUS TOOLKIT and FOCUS MENU. They are not required by FOCUS for normal execution.

### 1.3.7 WINFORMS.DATA

---

This dataset contains all of the sample MAINTAIN application procedures and is not required.

## 1.4 Dataset Naming Conventions

---

The library dataset names shown here are the names we use on the distribution tape. We strongly urge you to use the standard names as the lower level qualifiers for the datasets and add high level qualifiers appropriate for your installation. You can add high level prefixes to differentiate between test and production libraries, or between different FOCUS releases.



The standard names are used in all the documentation relating to the MVS/TSO version of FOCUS, and also in the JCL distributed with FOCUS maintenance tapes. You will find that maintaining FOCUS is very easy if you follow our suggested naming conventions.

The production versions of the execution time library FOCLIB.LOAD should be named

```
prefix.PROD.standardname
```

These names will be embedded in many JCL procedures and TSO CLISTs. You can switch releases without disturbing your users by renaming these datasets. Using this scheme, the production version of the FOCUS load library would be named

```
prefix.PROD.FOCLIB.LOAD
```

All other libraries, including test versions of the three execution time libraries, could be named

```
prefix.Vmm.Rnn.standardname
```

where:

mm           Version number

Nn           Release number

For example, the maintenance library for FOCUS Version 7.0 Release 01 would be named

```
prefix.V70R01.FOCCTL.DATA
```

## 1.5 Virtual Storage Requirements

---

This section describes the virtual storage requirements for the basic FOCUS system and for the Simultaneous Usage Facility.

### 1.5.1 FOCUS Storage Requirements

FOCUS storage requirements include space for the SASC library, the runtime environment for compiled SASC programs. Beginning with FOCUS Release 7.0.8, you have a choice of two options for installing this library, ALLRES and TRANSIENT:

- Running FOCUS with the ALLRES option ensures that other versions of SASC do not conflict with ours. The cost is 470K of below the line storage.
- Running FOCUS using the TRANSIENT option reduces the below-the-line storage requirements because the SASC transient module, SASLIBTR, is loaded in above the line memory. To install this version of the SASC runtime module see FOCCTL.DATA(TRLIB).

The following table summarizes the FOCUS storage requirements:

			<b>FOCUS</b>	<b>FOCUS with MAINTAIN</b>
Minimum	above the 16 meg storage line		4700K	7500K
	below the 16 meg storage line	ALLRES TRANSIENT	486K 310K	486K 310K
Recommended	above the 16 meg storage line		5000K	8300K
	below the 16 meg storage line	ALLRES TRANSIENT	486K 310K	486K 310K

Maintain application developers that use the Painter to create WINFORMs should increase storage by approximately 1M.

Larger region sizes will be needed under the following conditions:

- If you use high-resolution graphics that require GDDM.
- If you use FOCUS to access databases other than its own e.g.: DB2, IDMS, ADABAS, IMS.

### 1.5.2 Storage Requirements for the Simultaneous Usage (SU) facility

The following discussion pertains to the job that performs all the FOCUS database I/O for the optional Simultaneous Usage (SU) facility.

The virtual storage requirements of this central database job consist of a fixed component and a variable component. The fixed component is affected by the programs to be run under SU, and the variable component depends on the number of centrally controlled databases and on the number of TSO IDs or batch jobs that will be serviced.

The fixed component is about 512K bytes. This accounts for all the programs that will be loaded, and for about 60 database buffers of 4096 bytes each. This is close to the default maximum of 64 database I/O buffers FOCUS can manage.

The variable component needs 72 bytes per field in each centrally controlled database, plus a maximum of 16000 bytes for each TSO ID or batch job that communicates with the central database job. This is a worst case figure. A more likely average is about 6000 bytes per active user. The maximum would be needed, for example, in an installation with 20 concurrent users who will access 10 FOCUS databases, each of which has 100 fields. Taking the number of bytes per user at its largest (16000 bytes), we have:

fixed component	512,000 bytes
10 databases, 100 fields each	72,000 bytes
20 users, 16000 bytes each	320,000 bytes
<b>TOTAL</b>	<b>904,000 bytes</b>

In the absence of detailed information, the minimum virtual region for this job is 960K. The recommended region is 50% greater, or 1440K. This increase is based on efficiency considerations and activity on the central database job.

## 2 BASIC INSTALLATION STEPS

---

This chapter describes the installation steps for the MVS version of FOCUS. The steps for installing a new release of FOCUS are:

1. Allocate disk space for FOCUS libraries.
2. Unload the FOCUS distribution tape.
3. Install the FOCUS site code.
4. Select FOCUS default startup options.
5. Test the basic FOCUS system.
6. Install link-edited user exits and change FOCUS defaults.
7. Install the reentrant modules.
8. Reblocking FOCUS Modules
9. Install high-resolution 3270 graphics.
10. Install the IBI Subsystem or
11. Install the FOCUS SVC.
12. Install the Simultaneous Usage (SU) facility.
13. Create TSO logon procedures and CLISTs.

Appendix A lists several common mistakes that may occur during FOCUS installation.

If you are installing the FOCUS Menu, see Chapter 3 for installation instructions.

If you are installing the FOCUS ToolKit, see Chapter 3 for installation instructions.

If you are installing translated FOCUS error messages, see Chapter 3 for installation instructions.

NOTE: The installation of FOCUS Release 7.0 under MVS/SP, MVS/370, VS2, VS1, or earlier versions of OS is not supported.

### 2.1 Allocate Disk Space for FOCUS Libraries

---

The table on the next page shows the disk space requirements of the seven FOCUS libraries required to install the basic system. They must be catalogued with prefixed dataset names on a permanently mounted pack.

When you choose the local dataset names, remember that FOCLIB.LOAD, FUSELIB.LOAD, and ERRORS.DATA must be named so that they can be accessed in READ-only mode by all users. If you have the RACF or ACF2 security systems, you will have to establish a rule for each dataset to be unloaded.

The space requirements are stated for several disk types but any CKD disk device with a track capacity of at least 13030 bytes can be used.

The load libraries on the FOCUS distribution tape are blocked at 13030 bytes so that they can be unloaded onto a 3330 disk. The DCB attributes of the other libraries are as shown in the table on the following page. You can vary their blocksize if you like, but not their RECFM or LRECL attributes. Do not alter the blocksize of FOCCTL.DATA, since it contains linkage editor control cards. If the blocksize of the control cards is too big, the linkage editor will not accept them. The following chart shows the disk space requirements for FOCUS libraries:

		Disk Type 3390	
		cylinder	tracks
FOCLIB.LOAD	Primary Allocation	54	810
	Secondary Allocation	1	10
	Directory Blocks (RECFM=U, BLKSIZE=13030)	65	65
FUSELIB.LOAD	Primary Allocation	2	30
	Secondary Allocation	1	5
	Directory Blocks (RECFM=U, BLKSIZE=13030)	45	45
ERRORS.DATA	Primary Allocation	11	165
	Secondary Allocation	1	5
	Directory Blocks (LRECL=80, BLKSIZE=1600, RECFM=FB)	15	15

		Disk Type 3390		
		cylinder	tracks	blocks
FOCCTL.DATA	Primary Allocation	5	75	
	Secondary Allocation	1	5	
	Directory Blocks (LRECL=80, BLKSIZE=1600, RECFM=FB)	25	25	
FOCEXEC.DATA	Primary Allocation	9	135	
	Secondary Allocation	1	5	
	Directory Blocks (LRECL=80, BLKSIZE=1600, RECFM=FB)	30	30	
MASTER.DATA	Primary Allocation	3	15	
	Secondary Allocation	1	5	
	Directory Blocks (LRECL=80, BLKSIZE=1600, RECFM=FB)	10	10	
WINFORMS.DATA	Primary Allocation	3	15	
	Secondary Allocation	1	5	
	Directory Blocks (LRECL=80, BLKSIZE=1600, RECFM=FB)	50	50	
IBISASC.C550.LINKLIB	Primary Allocation			696
	Secondary Allocation			13
	Directory Blocks (BLKSIZE=6144, RECFM=U)			45
FUSELIB.DATA	Primary Allocation	1	3	
	Secondary Allocation	1	5	
	Directory Blocks (LRECL=80, BLKSIZE=1600, RECFM=FB)	4	4	

**Example 1: CLIST to Allocate Disk Space**

---

```

/* THIS IS AN EXAMPLE OF A CLIST THAT WILL ALLOCATE ALL THE FILES      */
/* NEEDED TO STORE FOCUS ON DISK                                       */
FREE ATTRLIST(FOCLOAD FUSLOAD ERRDATA CTLDATA EXCDATA MTRDATA MNTDAT FUSDAT
  SASDAT)
/*                                                                    */
ATTRIB FOCLOAD RECFM(U) BLKSIZE(13030) DSORG(PO)
ALLOC DA(FOCLIB.LOAD) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(FOCLOAD)
/*                                                                    */
ATTRIB FUSLOAD RECFM(U) BLKSIZE(13030) DSORG(PO)
ALLOC DA(FUSELIB.LOAD) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(FUSLOAD)
/*                                                                    */
ATTRIB ERRDATA RECFM(F B) BLKSIZE(1600) LRECL(80) DSORG(PO)
ALLOC DA(ERRORS.DATA) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(ERRDATA)
/*                                                                    */
ATTRIB CTLDATA RECFM(F B) BLKSIZE(1600) LRECL(80) DSORG(PO)
ALLOC DA(FOCCTL.DATA) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(CTLDATA)
/*                                                                    */
ATTRIB EXCDATA RECFM(F B) BLKSIZE(1600) LRECL(80) DSORG(PO)
ALLOC DA(FOCEXEC.DATA) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(EXCDATA)
/*                                                                    */
ATTRIB MTRDATA RECFM(F B) BLKSIZE(1600) LRECL(80) DSORG(PO)
ALLOC DA(MASTER.DATA) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(MTRDATA)
/*                                                                    */
ATTRIB MNTDAT RECFM(F B) BLKSIZE(1600) LRECL(80) DSORG(PO)
ALLOC DA(WINFORMS.DATA) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(MNTDAT)
/*                                                                    */
ATTRIB FUSDAT RECFM(F B) BLKSIZE(1600) LRECL(80) DSORG(PO)
ALLOC DA(FUSLIB.DATA) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid) USING(FUSDAT)
/*                                                                    */
ATTRIB SASDAT RECFM(U) BLKSIZE(6144) DSORG(PO)
ALLOC DA(SASC.C550.LINKLIB) CATALOG SPACE(n,n) DIR(m) xxx VOLUME(volid)
  USING(SASDAT)
/*                                                                    */
FREE ATTRLIST(FOCLOAD FUSLOAD ERRDATA CTLDATA EXCDATA MTRDATA MNTDAT FUSDAT
  SASDAT)

```

In the sample CLIST above, you will need to supply the following values:

xxx	Is either TRACKS or CYLINDERS.
n,n	Is the appropriate primary and secondary space allocation.
m	Is the number of directory blocks.
volid	Is a valid volume identifier.

**Example 2: JCL to Allocate Disk Space**

---

```
//JOBNAME JOB (ACCT), 'FOCUS ALLOC',
//          CLASS=T,MSGCLASS=X
//*****
//STEP1 EXEC PGM=IEFBR14
//*****
//FOCLOAD DD DSN=prefix.FOCLIB.LOAD,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=U,BLKSIZE=13030)
//FUSELIB DD DSN=prefix.FUSELIB.LOAD,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=U,BLKSIZE=13030)
//FUSELIBD DD DSN=prefix.FUSELIB.DATA,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//ERRORS DD DSN=prefix.ERRORS.DATA,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//FOCCTL DD DSN=prefix.FOCCTL.DATA,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//FOCEXEC DD DSN=prefix.FOCEXEC.DATA,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//MASTER DD DSN=prefix.MASTER.DATA,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//WINFORMS DD DSN=prefix.WINFORMS.DATA,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600)
//IBISASC DD DSN=prefix.IBISASC.C550.LINKLIB,DISP=(,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//          DCB=(RECFM=U,BLKSIZE=6144)
//SYSPRINT DD SYSOUT=*
```

In the sample JCL above, you will need to supply the following values:

- xxx            Is either TRK or CYL.
- n,n           Is the appropriate primary and secondary space allocation.
- M             Is the number of directory blocks.
- Volid         Is a valid volume identifier.



## 2.2 Unload the FOCUS Distribution Tape

---

This step provides sample JCL to unload the MVS version of FOCUS from the tape.

FOCUS is distributed on a 9-track 6250 BPI standard label tape or on a 3480 cartridge (The 9 track 6250 BPI tape is the standard). The volume serial number is clearly shown on the external tape label.

Insert the actual VOLSER into DD statements I1 through I60 in the IEBCOPY JCL shown below, and modify DD statements O1 through O60 to reflect the actual dataset names selected in Section 2.1.

If your system has TMS (Tape Management System), change the LABEL keywords to LABEL=(n,SL,EXPDT=98000).

```
//FOCLOAD EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//I1 DD DSN=FOCLIB.LOAD,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(1,SL)
//I2 DD DSN=FUSELIB.LOAD,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(2,SL)
//I3 DD DSN=ERRORS.DATA,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(3,SL)
//I4 DD DSN=FOCCTL.DATA,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(4,SL)
//I5 DD DSN=FOCEXEC.DATA,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(5,SL)
//I6 DD DSN=MASTER.DATA,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(6,SL)
//I55 DD DSN=WINFORMS.DATA,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(55,SL)
//I59 DD DSN=IBISASC.C550.LINKLIB,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(59,SL)
//I60 DD DSN=FUSELIB.DATA,DISP=(OLD,PASS),UNIT=TAPE,
// VOL=SER=volid,LABEL=(60,SL)
//O1 DD DSN=prefix.FOCLIB.LOAD,DISP=OLD
//O2 DD DSN=prefix.FUSELIB.LOAD,DISP=OLD
```

```
//03      DD      DSN=prefix.ERRORS.DATA,DISP=OLD
//04      DD      DSN=prefix.FOCCTL.DATA,DISP=OLD
//05      DD      DSN=prefix.FOCEXEC.DATA,DISP=OLD
//06      DD      DSN=prefix.MASTER.DATA,DISP=OLD
//055     DD      DSN=prefix.WINFORMS.DATA,DISP=OLD
//059     DD      DSN=prefix.IBISASC.C550.LINKLIB,DISP=OLD
//060     DD      DSN=prefix.FUSELIB.DATA,DISP=OLD
//SYSIN   DD      *
COPY      INDD=I1,OUTDD=O1
COPY      INDD=I2,OUTDD=O2
COPY      INDD=I3,OUTDD=O3
COPY      INDD=I4,OUTDD=O4
COPY      INDD=I5,OUTDD=O5
COPY      INDD=I6,OUTDD=O6
COPY      INDD=I55,OUTDD=O55
COPY      INDD=I59,OUTDD=O59
COPY      INDD=I60,OUTDD=O60
```

## Sample JCL for Complete Installation as a Batch Job

```
//JOBNAME JOB (ACCOUNTING INFOR), 'FOCUS BASE LOAD',MSGCLASS=A,
//          REGION=1024K,MSGLEVEL=(1,1)
//*****
//STEP1 EXEC PGM=IEBCOPY
//*****
//I1 DD DSN=FOCLIB.LOAD,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(1,SL)
//I2 DD DSN=FUSELIB.LOAD,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(2,SL)
//I3 DD DSN=ERRORS.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(3,SL)
//I4 DD DSN=FOCCTL.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(4,SL)
//I5 DD DSN=FOCEXEC.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(5,SL)
//I6 DD DSN=MASTER.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(6,SL)
//I55 DD DSN=WINFORMS.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(55,SL)
//I59 DD DSN=IBISASC.C550.LINKLIB,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(59,SL)
//I60 DD DSN=FUSELIB.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//     VOL=SER=volid,LABEL=(60,SL)
//*****
//O1 DD DSN=prefix.FOCLIB.LOAD,DISP=(,CATLG,DELETE),
//     UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//     DCB=(BLKSIZE=13030,RECFM=U)
//O2 DD DSN=prefix.FUSELIB.LOAD,DISP=(,CATLG,DELETE),
//     UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//     DCB=(BLKSIZE=13030,RECFM=U)
//O3 DD DSN=prefix.ERRORS.DATA,DISP=(,CATLG,DELETE),
//     UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//     DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
//O4 DD DSN=prefix.FOCCTL.DATA,DISP=(,CATLG,DELETE),
//     UNIT=SYSDA,VOL=SER=volid,SPACE=(xxx,(n,n,m)),
//     DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
```

```

//O5 DD DSN=prefix.FOCEXEC.DATA,DISP=( ,CATLG,DELETE) ,
// UNIT=SYSDA,VOL=SER=valid,SPACE=(xxx,(n,n,m)) ,
// DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
//O6 DD DSN=prefix.MASTER.DATA,DISP=( ,CATLG,DELETE) ,
// UNIT=SYSDA,VOL=SER=valid,SPACE=(xxx,(n,n,m)) ,
// DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
//O55 DD DSN=prefix.WINFORMS.DATA,DISP=( ,CATLG,DELETE) ,
// UNIT=SYSDA,VOL=SER=valid,SPACE=(xxx,(n,n,m)) ,
// DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
//O59 DD DSN=prefix.IBISASC.C550.LINKLIB,DISP=( ,CATLG,DELETE) ,
// UNIT=SYSDA,VOL=SER=valid,SPACE=(xxx,(n,n,m)) ,
// DCB=(BLKSIZE=6144,RECFM=U)
//O60 DD DSN=prefix.FUSELIB.DATA,DISP=( ,CATLG,DELETE) ,
// UNIT=SYSDA,VOL=SER=valid,SPACE=(xxx,(n,n,m)) ,
// DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
//SYSIN DD *
COPY INDD=I1,OUTDD=O1
COPY INDD=I2,OUTDD=O2
COPY INDD=I3,OUTDD=O3
COPY INDD=I4,OUTDD=O4
COPY INDD=I5,OUTDD=O5
COPY INDD=I6,OUTDD=O6
COPY INDD=I55,OUTDD=O55
COPY INDD=I59,OUTDD=O59
COPY INDD=I60,OUTDD=O60
/*
//SYSPRINT DD SYSOUT=*

```

In the sample JCL above, you will need to supply the following values:

xxx	Is either TRK or CYL.
n,n	Is the appropriate primary and secondary space allocation.
m	Is the number of directory blocks.
valid	Is a valid volume identifier.

NOTE: If you are installing from a 3480 cartridge, you may have to change TAPE to your site-defined name for 3480 drives in DD statements I1 through I60 above.

## 2.3 Install the FOCUS Site Code

---

A unique identifying site code is assigned to each site where FOCUS is installed. This code, of the form nnnn.nn, is displayed in the FOCUS release signature whenever FOCUS is first entered, and also by the ? RELEASE command. You should know your site code whenever you call IBI with any question regarding FOCUS service or maintenance. It is the key to all our service and maintenance records.

The site code is embedded into FOCUS itself at installation time, or at any time after the installation. FOCUS will run properly without an embedded site code, but the release signature will show a blank where the site code would be.

The site code of your installation is shown on the packing list that accompanied your installation package. If you do not know your site code, you can always obtain it by calling the Installation Services Department in New York, at 1-800-736-6130.

To install your FOCUS site code, zap your site code into the FOCUS module. The JCL and all necessary instructions for this step are contained in member FOCSITE of the maintenance dataset FOCCTL.DATA.

Usage Notes:

- The distributed version of FOCUS does not contain a site code. If you do not zap your site code into the FOCUS module it will show as a blank.
- Your site code will have to be zapped into FOCUS with each new release you install.
- You do not have to rezap the site code into FOCUS following the application of an individual PTF, unless otherwise stated.

## 2.4 Select FOCUS Default Startup Options

---

Some sites may wish to change their startup options for FOCUS. FOCUS provides two means for implementing site-wide configuration parameters.

Member FOCPARM in ERRORS.DATA is executed before each user's FOCEXEC (PROFILE), and may only include SET commands. The sample member FOCPARM in ERRORS.DATA provided on the FOCUS tape is shown below:

```

***** TOP OF DATA *****
- *-----*
- * THIS SECTION CONSISTS SOLELY OF SET COMMANDS, USED TO CUSTOMIZE *
- * THE BEHAVIOR OF FOCUS AT YOUR SITE. PLEASE NOTE THAT ONLY SET *
- * COMMANDS ARE SUPPORTED, AND THAT ANY OTHER FOCUS COMMANDS ARE *
- * NOT PERMITTED AND WILL FORCE YOU OUT OF FOCUS. THIS MEMBER MAY *
- * NOT BE USED AS A PROFILE EXCEPT FOR SET COMMANDS. *
- *-----*

SET EMPTYREPORT=OFF

SET BLKCALC=NEW

SET FIELDNAME=NEW

SET QUALCH= .

SET QUALTITL=OFF

SET HOLDSTAT=ON

SET HOTMENU=OFF

SET IMMEDIATE=OFF

SET AUTOPATH=ON

SET AUTOINDEX=ON

***** BOTTOM OF DATA *****

```

Note that only SET commands may be included. SET commands are discussed in Chapter 21 of the *FOCUS for IBM Mainframe Users Manual Release 7.0*.

There are two other FOCPARM files: FOCPARMA and FOCPARMC, where FOCPARMA is for aggressive and FOCPARMC is for conservative settings. The aggressive settings are those that exploit new features, such as BLKCALC, AUTOPATH and AUTOINDEX. FOCPARMC does not exploit new features and behavior, and is generally more conservative than both FOCPARM and FOCPARMA. These files are members of the ERRORS.DATA PDS in MVS. You can use either member by changing FOCPARM to a different name, then renaming FOCPARMA or FOCPARMC to FOCPARM.

FOCPARMA contains:

```

-*-----*
-*  FOCPARM MAY BE USED TO CUSTOMIZE ATHE BEHAVIOR OF FOCUS AT YOUR  *
-*  INSTALLATION.  PLEASE NOTE THAT ONLY SET COMMANDS ARE SUPPORTED.  *
-*  ANY OTHER FOCUS COMMANDS YIELD AN ERROR WHICH TERMINATED THE    *
-*  FOCUS SESSION                                                    *
-*-----*

SET EMPTYREPORT=OFF
SET BLKCALC=NEW
SET FIELDNAME=NEW
SET QUALCH=.
SET QUALTITL=ON
SET HOLDSTAT=ON
SET HOTMENU=ON
SET IMMEDTYPE=ON
SET CACHE=256
SET AUTOPATH=ON
SET AUTOINDEX=ON
SET PRINTPLUS=ON

```

FOCPARMC contains:

```

-*-----*
-* THIS SECTION CONSISTS SOLELY OF SET COMMANDS, USED TO CUSTOMIZE *
-* THE BEHAVIOR OF FOCUS AT YOUR SITE. PLEASE NOTE THAT ONLY SET *
-* COMMANDS ARE SUPPORTED, AND THAT ANY OTHER FOCUS COMMANDS ARE *
-* NOT PERMITTED AND WILL FORCE YOU OUT OF FOCUS. THIS MEMBER MAY *
-* NOT BE USED AS A PROFILE EXCEPT FOR SET COMMANDS. *
-*-----*

SET EMPTYREPORT=ON

SET BLKCALC=OLD

SET FIELDNAME=OLD

SET HOLDSTAT=OFF

SET HOTMENU=OFF

SET IMMEDIATE=OFF

SET AUTOPATH=OFF

SET AUTOINDEX=OFF

```

An assembly language program called FOCPARM located in the dataset FOCCTL.DATA has also been provided. It enables sites to change startup default options, including some which may not be specified in SET commands. Instructions have been provided for changing the following FOCUS startup options:

- Displaying the FOCUS banner (BANNER).
- Clearing the screen (CLRSCRN).
- National Language Support Defaults (LANG).
- Selecting an alternative PROFILE or no PROFILE (PROFOPT).
- Continental Decimal Notation (CDN).

The options may be changed by replacing old options with new ones, and then assembling and linking the options into FOCINI. The default values in the sample FOCPARM provided with FOCUS are shown below. A value of '1' enables an option; a value of '0' disables an option.



```

* This file is used to change startup options
*
FOCPARM  CSECT
BANNER   DC      F'1'           1. Indicates display banner
CLRSCRN  DC      F'1'           2. Indicates clear screen at startup
*
* NATIONAL LANGUAGE SUPPORT DEFAULTS
*
LANG      DC      F'1'           3. Indicates Language Number (INTLCM)
JTERM    DC      CL8'IBM3270' 4,5.Indicates J-Terminal      (INTLCM)
          DC      F'0'           6. Available
          DC      F'0'           7. Available
          DC      F'0'           8. Available
*
PROFOPT  DC      F'0'           9. NOPROF/PROFILE 0=not allowed,1=allow
CDN      DC      F'0'           10. CDN SETTING 0=OFF 1=ON
XDBSC    DC      CL4'@   '      11. CHARACTER FOR EXT. DB SECUR. (7975)
PCHOLD   DC      F'1'           12. PCHOLD FLAG (11108)
TTFEX    DC      CL4'PDS '      13. DSORG OF TABLTALK FOCEXEC (11516)
DATTIM   DC      CL4'EDA '      14. DATE/TIME FORMAT (13619)
*-----
* A value of '1' enables an option, a value of '0' disables an option
* Other values are defaults for the option.
*
* DO NOT ADD, DELETE, OR CHANGE ORDER OF THE OPTIONS ABOVE!
*
* DO NOT ALTER THE CODE BELOW IN ANY WAY!
*****
.
.
.etc

```

NOTE: Since each installation of FOCUS generates its own version of the FOCPARM options, the procedures outlined below must be performed each time a new version of FOCUS is installed.

- Edit member FOCPARM of FOCCTL.DATA to revise the options as needed, and save the member back to dataset FOCCTL.DATA.
- Assemble the member FOCPARM from FOCCTL.DATA. The following is sample JCL for this:

```
//ASSEM      EXEC  PGM=IEV90 , PARM= 'LIST,DECK,NOOBJ,NOALIGN'
//SYSLIB     DD    DISP=SHR,DSN=SYS1.MACLIB
//SYSUT1     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT2     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT3     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSPUNCH  DD    DISP=SHR,DSN=prefix.OBJ(FOCPARM)
//SYSPRINT  DD    SYSOUT=*
//SYSLIN     DD    DUMMY
//SYSIN      DD    DISP=SHR,DSN=prefix.FOCCTL.DATA(FOCPARM)
```

- Link this new program into FOCUS. The following is sample JCL for this:

```
//LINKFOC   EXEC  PGM=IEWL, PARM= 'LET,NCAL,LIST,SIZE=1024K'
//SYSPRINT  DD    SYSOUT=*
//SYSUT1    DD    UNIT=SYSDA,SPACE=(CYL,(10,1))
//SYSLMOD   DD    DISP=OLD,DSN=prefix.FOCLIB.LOAD    <- FOCUS
//MAINTAIN  DD    DISP=SHR,DSN=prefix.FOCCTL.DATA    <- FOCCTL
//OBJECT    DD    DISP=SHR,DSN=prefix.OBJ           <- Assembled code
//SYSLIN    DD    *
            INCLUDE OBJECT(FOCPARM) <----- Assembled FOCPARM
            INCLUDE SYSLMOD(FOCINI)  <----- Module to be Changed
            INCLUDE MAINTAIN(FOCINI) <----- Linkedit Control Statements
            NAME   FOCINI(R)          <----- New Module
/*
```

## 2.5 Test the Basic FOCUS System

---

This step is a test under TSO of the basic system installed in Step 2.

### 2.5.1 Logon into a TSO region

---

You can use any TSO ID from which the installed FOCUS libraries can be accessed. Any logon procedure will do.

### 2.5.2 Allocate space for a test FOCUS database

---

```
ALLOC DA(CAR.FOCUS) SP(5,5) TRACKS LRECL(4096) BLKSIZE(4096) RECFM(F) NEW
```

This database will be allocated as ddname CAR and loaded with data in the next step. You can use it for any additional tests and experiments with FOCUS after the completion of Section 2.5.4.

### 2.5.3 Execute the TSOALLOC CLIST

---

This TSO CLIST allocates the FOCUS libraries installed in Section 2, and invokes FOCUS via the CALL command. The CLIST is member TSOALLOC of FOCCTL.DATA, as allocated in Section 2.1. You will be prompted for the fully qualified dataset names of FOCLIB.LOAD, ERRORS.DATA, MASTER.DATA, FOCEXEC.DATA, and of the test FOCUS database allocated in Section 2.5.2. At the end of this step, you will see the FOCUS release banner (meaning that you are in FOCUS), and double caret (>>) that indicates FOCUS is ready to receive a command.

### 2.5.4 Execute the FOCUS test procedure CARTEST

---

When you have received the double caret from FOCUS, type:

```
>>EX CARTEST
```

press

```
ENTER KEY
```

The CARTEST procedure, which is a member of FOCEXEC.DATA, will load data into the CAR database, as defined by member CAR of MASTER.DATA, and test the major FOCUS areas. The FOCEXEC and MASTER PDSs must be allocated for this procedure to work. Messages are typed as each test is completed and the end of all tests is signaled by the message:

```
AT END OF TEST MAJOR AREAS OK
```

This completes the installation of the basic FOCUS system. You can return to the TSO command level by typing the FOCUS command FIN when you receive the double caret after the above message.

If you receive any error messages during the test step, call your local IBI representative or the IBI Installation Services Department in New York at 1-800-736-6130.

The most common reasons for errors are failed allocations for ddnames CAR, CCARS, MASTER, FOCEXEC, HOLD, HOLDMAST, FOCSTACK, or FOCSML, all of which are allocated by TSO ALLOCATE commands issued by the TSOALLOC CLIST.

### Sample JCL for Executing CARTEST as a Batch Job

---

```
//CARTEST EXEC PGM=FOCUS,REGION=2048K
//STEPLIB DD DSN=prefix.FOCLIB.LOAD,DISP=SHR
//MASTER DD DSN=prefix.MASTER.DATA,DISP=SHR
//FOCEXEC DD DSN=prefix.FOCEXEC.DATA,DISP=SHR
//ERRORS DD DSN=prefix.ERRORS.DATA,DISP=SHR
//CCARS DD DSN=prefix.MASTER.DATA(CCARS),DISP=SHR
//OFFLINE DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
EX CARTEST
FIN
/*
```

If you omit FIN as the last command to be executed, FOCUS will run properly but it will terminate with condition code 8.

### 2.5.5 Online Release Information

---

Release statistics, installation and operation changes, program temporary fix (PTF) information, and release notes are available online. To view the online release information, issue the following command from the FOCUS prompt:

```
>> HELP README
```

This will instruct you to execute the READMEF FOCEXEC. This FOCEXEC allows you to choose what release information you are interested in (PTFs, Installation information, New Features, etc.). These reports contain all of the information that was previously published in the Maintenance Log for each FOCUS Version, and replace the Maintenance Log. The ERRORS and MASTER PDSs must be allocated in order to run READMEF.

## 2.6 Install Link-Edited User Exits and Change FOCUS Defaults

---

This step is optional and independent of all other installation steps. It can be performed any time after the basic FOCUS system is installed.

User exits, such as the Usage Accounting and Dataset Naming Enforcement Exits, are installed by link editing your own code into the basic FOCUS module. Chapter 4 describes these exits and the required linkage editor instructions.

Certain FOCUS defaults can be changed, including:

- the number of active FOCUS databases allowed.
- field buffer size (DATREC and LINREC).
- the default space allocation table for work files (TABLA).
- file-naming conventions (DSNTAB).
- the list of commonly executed TSO commands (TSOTAB).

Chapter 3 contains detailed instructions for changing these defaults.

## 2.7 Install the Reentrant Modules

---

Improved performance gains can be achieved for each TSO/FOCUS user by placing FOCUS re-entrant and serially reusable modules in the Extended Link Pack Area. All programs that are part of PLPA have a corresponding entry in the LPA Directory, which is created at IPL time. This significantly improves the performance of FOCUS, and also reduces the total amount of virtual storage each FOCUS user requires.

It is strongly recommended that you install the reentrant code in the Extended Link Pack Area. If you do not, a copy of the reentrant module is loaded into each user's address space at run time.

Sample JCL to perform this procedure is found in members JFSALLPA, JFSCPLPA, JFSDELPA, and JFSCPBACK of the dataset FOCCTL.DATA.

## 2.8 Reblocking FOCUS Modules

---

All FOCUS load libraries are distributed with a blocksize of 13030 bytes so that they can be loaded onto any CKD disk device with a track capacity of at least 13030 bytes. Even if you allocated the load library on a 3380 pack, with a full track capacity of 47476 bytes, the largest single record in any load module is still only 13030 bytes. If you want to gain the benefit of the larger track capacity, you must relink the modules; it is not enough to copy them from a library with a small blocksize to one with a larger blocksize. The actual load module blocksizes have no logical significance, but they do have a small effect on efficiency. For 3380 and 3390 devices, we recommend half track blocking.

The following example shows how to relink members FOCUS and F\$ROOT:

```

/* REBLOCKING FOCUS MODULES.

//LINK1      EXEC  PGM=IEWL,PARM='NCAL,LET,LIST,SIZE=(2048K)'
//SYSPPRINT  DD    SYSOUT=*
//SYSUT1     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//OLDLIB     DD    DSN=prefix.FOCLIB.LOAD,DISP=OLD
//FOCCTL     DD    DSN=prefix.FOCCTL.DATA,DISP=SHR
//SYSLMOD    DD    DSN=prefix.TEST.FOCLIB.LOAD,DISP=OLD
//SYSLIN     DD    *

        INCLUDE OLDLIB(FOCUS)          ---membername
        INCLUDE FOCCTL(FOCUS)          ---same membername
        NAME FOCUS(R)                  ---same membername

/*

//LINK2      EXEC  PGM=IEWL,PARM='RENT,NCAL,LET,LIST,SIZE=(2048K)'
//SYSPPRINT  DD    SYSOUT=*
//SYSUT1     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//OLDLIB     DD    DSN=prefix.FOCLIB.LOAD,DISP=OLD
//FOCCTL     DD    DSN=prefix.FOCCTL.DATA,DISP=SHR
//SYSLMOD    DD    DSN=prefix.TEST.FOCLIB.LOAD,DISP=OLD
//SYSLIN     DD    *

        INCLUDE OLDLIB(F$ROOT)         ---membername
        INCLUDE FOCCTL(F$ROOT)         ---same membername
        NAME F$ROOT(R)                 ---same membername

/*

```

The other modules in FOCLIB.LOAD can be relinked similarly. Only the larger modules are worth reblocking: FOCUS, F\$ROOT, FOCSAM, HLISNK, and HLI. The others are so small that reblocking would not reduce the actual number of module records. If a module had an alias, be sure that it is not lost. (Aliases and membertnames always have the same entry point.)

Note: The HLI module requires the REUS option be specified in the PARM field when executing a link edit for this module. For example:

```
//LINKHLI EXEC PGM=IEWL,PARM='REUS,NCAL,LET,SIZE=(2048K)'
```

## 2.9 Install High-Res Graphics for 3270 Type Terminals

---

This step describes the installation of FOCUS high-resolution graphics for IBM 3270 type terminals, which require the use of the IBM GDDM graphics support program. Omit this step if you are not installing high-resolution graphics or the ICU Interface. High-resolution graphics on ASCII terminals are part of the basic FOCUS system and do not require a separate installation step.

The simple link edit through which this feature is installed can be done any time after the basic FOCUS system is installed. The JCL for this link edit is as follows:

```
//LINKGDDM EXEC PGM=IEWL,PARM='LET,NCAL,LIST,SIZE=1024K'
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=SYSDA,SPACE=(TRK,(30,10))
//GDDM DD DSN=gddmlibrary,DISP=SHR
//FOCCTL DD DSN=prefix.FOCCTL.DATA,DISP=SHR
//SYSLMOD DD DSN=prefix.FOCLIB.LOAD,DISP=OLD
//SYSLIN DD *
INCLUDE GDDM(ADMASNT) GDDM STUB
REPLACE CHART
INCLUDE SYSLMOD(FOCUS) BASIC FOCUS MODULE
INCLUDE FOCCTL(FOCUS) LINK EDIT CONTROL STATEMENTS
NAME FOCUS(R) FOCUS WITH GDDM STUB
/*
```

Before you run this link edit, make sure that the FOCLIB.LOAD library has at least 150 free tracks for the new FOCUS module. You can also create the new module in a scratch load library, delete FOCUS from FOCLIB.LOAD, compress the library to regain the deleted space, and copy the new FOCUS module back into FOCLIB.LOAD.

NOTE: The GDDM library is often named SYS1.GDDMLOAD. Contact your System Administrator for the name of this library.



## 2.10 Installing the IBI Subsystem

### 2.10.1 Installation Planning

---

This section discusses the impact of installing the IBI Subsystem on your MVS system. You should read both this section and the following section before starting the installation procedure. Install either the IBI Subsystem or the FOCUS SVC, not both.

#### Installation

---

If you are installing on an MVS system that is release 5.2 or later, it is necessary to update IEFSSNxx in SYS1.PARMLIB to install the IBI Subsystem.

Sample:

JES2, , , PRIMARY	JOB ENTRY SUBSYSTEM
RACF, IRRSSI00, #	RESOURCE ACCESS CONTROL FACILITY
CICS	CUSTOMER INFORMATION CONTROL SYSTEM
IBIS	IBI SUBSYSTEM

Even though you update IEFSSNxx with the subsystem name, you must still run the SUBSYSI utility with the START option to load and activate the IBI Subsystem. Alternately, you may use a name in IEFSSNxx if it is currently unused. The sample job SUBSYSNM in FOCCTL.DATA demonstrates how to change the default subsystem name to one of your own choosing.

The SUBSYSI utility requires an APF-authorized library. This library may be an existing library, a member of your system's LNKLIST, or a new library created just for SUBSYSI. Whichever option you choose, you may wish to protect SUBSYSI so that it may only be run by authorized personnel.

#### Capacity and Performance

---

See the "Capacity and Performance" in Chapter 1, for a detailed discussion of the impact of the IBI Subsystem on your MVS system.

#### Training

---

Appropriate operations and system support staff should be trained in the use of the SUBSYSI utility for starting and controlling the subsystem, and the operator commands which the subsystem provides, DISPLAY and SET. In addition, you might need to update your system IPL procedures to ensure the subsystem is properly started after MVS is initialized.

## 2.10.2 Installation

---

The following procedure lists the steps to follow to install the IBI Subsystem. The procedure assumes that FOCUS has already been installed and tested, and that the datasets FOCLIB.LOAD and FOCCTL.DATA have been loaded on your system.

All of the samples discussed are shipped in FOCCTL.DATA, and are reproduced in Appendix D for your reference.

### Installation Steps

---

1. Choose a subsystem name.

Choose a valid name to use for the subsystem. It must be four characters long. The default name is "IBIS".

2. Choose an APF-authorized library.

SUBSYSI must be run from an APF-authorized library. You may copy SUBSYSI to a library in the LNKLST or to an existing authorized library, or you may create a new library for this purpose.

As SUBSYSI should only be run by authorized personnel, it is recommended that the library in which it resides be protected from unauthorized access.

While FOCLIB.LOAD is APF-authorized when MSO is installed, all FOCUS users have access to it in order to run FOCUS. You will probably want to use a separate library, in order to facilitate protecting SUBSYSI from unauthorized use.

3. Update SYS1.PARMLIB (OPTIONAL before MVS 5.2)

Update IEFSSNxx member of SYS1.PARMLIB with the name of our subsystem.

If you need to create a new APF-authorized library, or are adding non-swappable programs (SSCTL for MSO, HLISNK for SU), update SYS1.PARMLIB as needed to specify these changes.

An IPL of MVS is required to activate a new subsystem name. Step 5, below, provides a method to change the name of the IBI subsystem. This job can be used to modify the IBI subsystem name to an already existing, unused, entry in IEFSSNxx. This will avoid the necessity of an IPL.

4. Apply the SUBSYSZP zap to activate the subsystem.

Run the JCL in SUBSYSZP to activate the IBI Subsystem instead of the FOCUS SVC.

5. Zap to change subsystem name (OPTIONAL).

If you have chosen a name other than "IBIS" for the IBI Subsystem, edit the JCL in SUBSYSNM with the appropriate name and run it to apply the zap.

6. Copy modules to an APF-authorized library and protect the library.

The JCL in SUBSYSCP will copy SUBSYSI and related modules from FOCLIB.LOAD into an APF-authorized library. This library may be a library dedicated to SUBSYSI, a member of the LNKLST, or any other library you choose. As SUBSYSI should only be run by authorized personnel, ensure the library is protected from unauthorized access.

7. Protect modules or delete them from FOCLIB.LOAD.

SUBSYSI and the other modules which were copied should be deleted from FOCLIB.LOAD or protected so that they may only be executed by authorized personnel.

8. Set up automatic start of the subsystem (OPTIONAL).

If desired, set up JCL to run SUBSYSI and initialize the IBI Subsystem at system startup. This is especially useful if HiperFOCUS is installed, to ensure all users are subject to the HiperBUDGET controls established by the subsystem. Sample JCL to run SUBSYSI is found in members SUBSYSIJ and SUBSYSIH of FOCCTL.DATA.

HiperFOCUS is documented in Chapter 17 of the *FOCUS for IBM Mainframe Users Manual Release 7.0*.

9. Set up a SUBSYSI cataloged procedure (OPTIONAL).

If desired, set up a cataloged procedure so SUBSYSI can be run as a started task. A sample procedure is found in SUBSYSP.

10. IPL the system (OPTIONAL).

The system must be IPLed to activate changes to SYS1.PARMLIB. If an authorized library was already available, the IPL is not necessary in order to use the subsystem, although some facilities, such as non-swappable servers, will not be available until after the IPL.

11. Start the IBI Subsystem.

Bring up the subsystem by running SUBSYSI with the START parameter. It is now ready to use.

The SUBSYSI program requires the allocation of a STEPLIB DD statement when performing the START function. If the load library used (FOCLIB.LOAD) is defined in the link list, it will still require APF authorization (IEAAPF00) so that it may be executed from the STEPLIB with the proper APF authorization.

If SUBSYSI START is executed without a STEPLIB statement, the following message will occur:

```
SUBSYS003 UNABLE TO LOAD MODULES TO CSA.
```

To remedy this situation, simply allocate the load library to STEPLIB.

## Product-Related Installation Issues

---

### HiperFOCUS

The IBI Subsystem must be running in order for the HiperBUDGET feature to be activated. You should ensure that the appropriate SET commands are issued before the first user accesses FOCUS.

It is recommended that the subsystem is initialized and the SET commands be issued at MVS startup. See the JCL in member SUBSYSIH of FOCCTL.DATA for an example.

### MSO, SU, SmartMode, IMS BMP

No changes are required for MSO, SU, SmartMode (which uses SU), or the IMS BMP. Use of the subsystem is transparent for these products. They do not distinguish between IBI SVC and IBI Subsystem communications.

## Maintenance Issues

---

### Applying Maintenance

After maintenance has been applied to FOCUS, be sure to re-run all of the appropriate zap and copy jobs, and then run SUBSYSI with the REPLACE parameter to load the new subsystem modules.

You should ensure that no servers are using the subsystem when the REPLACE operation is performed.

### Multiple FOCUS Releases

The subsystem supports use by multiple versions of FOCUS. The subsystem need not be re-installed when upgrading to a new release of an Information Builders product. Release prior to FOCUS Release 7.0 do not support the IBI Subsystem.

### Coexistence with the FOCUS SVC

The modules in any given copy of FOCLIB.LOAD will use either the FOCUS SVC or the IBI subsystem. If you want to use both, you will need two copies of FOCLIB.LOAD, each with the appropriate zap applied.

## De-installing the IBI Subsystem

You may de-install the IB Subsystem by running SUBSYSI with the REMOVE parameter. When doing this, be sure that no IBI server address spaces which use the IBI Subsystem are active.

### 2.10.3 IBI Subsystem Usage

#### The SUBSYSI Program

---

The SUBSYSI program provides control of the IBI Subsystem. Its basic function is to initialize the subsystem, and it can also be used to stop, unload, and replace the subsystem modules in storage.

SUBSYSI may be run as a batch job or as a started task. JCL samples are provided in members SUBSYSIJ and SUBSYSJP, respectively, of FOCCTL.DATA.

You should not use SUBSYSI to STOP or REMOVE the subsystem while any IBI server address spaces are running. All servers must be stopped before stopping the subsystem. While HiperFOCUS users may continue to run, the HiperBUDGET feature will be deactivated when the subsystem is stopped.

Valid parameters to SUBSYSI are:

- **START:** Initialize the subsystem -- load modules into CSA. If the modules have already been loaded but the subsystem is not active (i.e., STOPped), RESTART is assumed and executed.
- **STOP:** Logically disable the subsystem without removing the modules from CSA. Do not do this while any IBI servers are running.
- **RESTART:** Restart the subsystem following the STOP command. The modules already exist in CSA (i.e., START has already been run).
- **REPLACE:** Reload modules into CSA; this is used to upgrade without re-IPLing MVS. You should only run this after performing a STOP.
- **REMOVE:** Disable the subsystem and remove modules from CSA. This operation will leave the SSCT entry and small (2K) module, but the module will be inactive. Do not do this while any IBI servers are running.

When START or RESTART are specified, SUBSYSI reads commands from SYSIN, as if they were issued from the MVS console. This is useful for setting the HiperBUDGET limits from a batch job or started task. A sample is provided in member SUBSYSIH of FOCCTL.DATA.

## Operator Controls

---

To issue operator commands to the subsystem, prefix them with the IBI Subsystem name. There may be no leading blanks in the command, but extra blanks are permitted in the command text. For example, assuming the subsystem has been installed as “IBIS”, the following commands are all valid:

```
IBIS D SUBSYSTEM
IBIS   DISPLAY   IBIPROD
IBIS           D IBIPROD
IBIS SET  MVSLIM=500
IBIS T FILELIM=300, MVSLIM=500
```

The subsystem can also be controlled by submitting commands using the SUBSYSI program. In this case the PARM parameter on the EXEC card should not be specified. The commands are supplied via the SYSIN DD. The following example shows the several commands being given in this manner. The output of these commands will be in the job log.

```
//SUBSYSI EXEC PGM=SUBSYSI,REGION=4096K,
//STEPLIB DD DSN=*****.SUBSYS.LOAD,DISP=SHR
//SYSIN DD *
D SUBSYSTEM
D IBIPROD
D HIPERBUDGET
/*
```

## Display Commands

The DISPLAY commands show subsystem-related or subsystem-managed information. The keyword “DISPLAY” may be abbreviated as “D”. The following commands are available:

- DISPLAY IBIPROD displays one line of information for every IBI address space running on the system. For MSO and EDA/SQL servers, information such as current and maximum number of users and LU names are displayed.
- DISPLAY HIPERBUDGET shows current expanded storage usage and HiperBUDGET limit values.
- DISPLAY SUBSYSTEM shows internal information about the subsystem.

## SET Commands

The SET commands are used with the HiperBUDGET feature of HiperFOCUS to change the limit values. The keyword “SET” may be abbreviated as “T”. More than one limit may be set on the same command, as shown in the example above. Values which may be set are:

- SET MVSLIM = <value> | -1
- SET SERVLIM = <value> | -1
- SET TCBLIM = <value> | -1
- SET FILELIM = <value> | -1

The value of “-1” specifies that there is no limit checking for that particular parameter. Refer to Chapter 17 of *FOCUS for IBM Mainframe Users Manual Release 7.0* for further information on these limits.

### 2.10.4 Troubleshooting

---

In the event of a problem with the IBI Subsystem, the following may prove helpful before contacting Information Builders technical support.

The following are the most common causes of problems with the IBI Subsystem:

1. One or both of the zap jobs weren't run, or didn't run correctly, or an incorrect value was used for the zap.
2. The zap jobs were run against the wrong library (SUBSYSI, in particular, might be in two different libraries).
3. MVS wasn't IPLed, or the wrong system was IPLed, so the changes to SYS1.PARMLIB haven't taken effect.
4. The wrong SYS1.PARMLIB was updated, or the wrong PARMLIB members were updated (they must be specified in the appropriate IEASYSxx), or the wrong system's PARMLIB was updated.
5. SUBSYSI is not being run from an APF-authorized library.
6. SUBSYSI was not run to START the subsystem.
7. SUBSYSI was run with STOP or REMOVE and disabled the subsystem.
8. For MSO, SU, and so on, the communication datasets were not allocated.
9. The communication datasets were allocated, but with the wrong ddname.
10. The communication datasets were allocated incorrectly; for example, MSO requires two distinct datasets.

11. There is a conflict with an existing subsystem name.
12. There is a mismatch between the subsystem code and FOCUS code. If necessary for a new release of FOCUS, you must re-run the appropriate copy and zap jobs.
13. FOCUS is running out of an unzapped library.
14. The HiperBUDGET parameters were not properly set.
15. A S065 or S0C4 may result when starting any FOCUS product that requires IBI Subsystem on an MVS 5.2 release or later and the IBI subsystem name has not been placed in the MVS system name table, IEFSNxx.

### Diagnostic information

---

When contacting Information Builders technical support for subsystem problems, some or all of the following information may be needed, depending on the nature of the problem:

1. For installation problems, the JCL, input, and output for all zap and copy jobs run.
2. For installation problems, all SYS1.PARMLIB updates performed.
3. The output from the D SUBSYSTEM, D IBIPROD, and D HIPERBUDGET commands.
4. The JCL, input, and output for all SUBSYSI jobs run.
5. The JCL and output for the failing address space(s).



## 2.11 Install the FOCUS SVC

---

The FOCUS SVC is required only by Simultaneous Usage (SU), by MSO, and by the BMP extension of the FOCUS/IMS-DB interface. Skip this step if you will not use these features.

If your FOCUS SVC was installed prior to Release 6.0, it cannot be used with Release 7.0 of FOCUS. In this case you must replace it with the SVC from the 7.0 release tape. Release 6.0 SVCs will work with all current and prior FOCUS releases, under MVS/XA or MVS/ESA.

This step places the FOCUS SVC into the system library containing all other SVCs of this type. It does not modify the basic FOCUS system to use the SVC under its locally assigned number. Although its installation is a routine MVS function, it must be performed by the MVS system support group, and it must conform to the requirements described in the IBM manual, *System Programming Library Supervisor*.

The source code for this SVC is distributed as member FOCSVC of FOCCTL.DATA. The FOCUS SVC is of type 3, it runs enabled, and it does not set or require any locks. The FOCUS SVC is supported only for the MVS operating system, not VS1 or SVS.

To install the FOCUS SVC:

### 2.11.1 Select a user SVC number in the allowed user range.

---

Remember this number. It will be zapped into the basic FOCUS system when you install any FOCUS features that require the use of this SVC.

The attributes for this SVC number are established via the SVCTABLE macro during the MVS system generation or, for MVS/XA 2.2.0 and higher, via the IEASVCxx member of SYS.PARMLIB.

NOTE: The SVC chosen (from 140 to 255) must not be an "APF" authorized SVC number in the SVC table.

### 2.11.2 Modify the FOCUS SVC global variables.

---

The FOCSVC source code contains global variables that you can set to configure the FOCUS SVC. Therefore, you now set a global variable corresponding to the SVC number you have selected, rather than changing the CSECT name. Other global variables may be changed (there are eight in all). When the correct variables have been set (you must at least set the SVC number), assemble and link your SVC according to Section 2.11.3, below.

### 2.11.3 Assemble and link edit the modified source code.

---

The assembly step requires access to the IHARB macro, which is usually found in the SYS1.AMODGEN system macro library. The member name assigned to the SVC in the load library corresponds to the EBCDIC form of the SVC number set in Section 2.11.2, above. Therefore, if the SVC number is decimal 245, the module name becomes IGC0024E; decimal 245 in signed zoned format is hexadecimal F2F4C5, which in EBCDIC becomes 24E. This module name is then included in the assembly and link edit JCL as shown below.

Link edit the SVC with the following parameters:

```
AMODE=31 , RMODE=24
```

The SVC becomes effective with the next IPL.

Sample JCL for this step appears on the following page.

**Sample JCL for Assembly and Link Edit**

```

//FOCSVC JOB
//ASM EXEC PGM=IEV90,PARM='RENT,DECK,LIST,NOOBJECT'
//SYSLIB DD DSN=SYS1.MACLIB,DISP=SHR
// DD DSN=SYS1.AMODGEN,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSLIN DD DUMMY
//SYSPUNCH DD DSN=&&TEMP,DISP=(,PASS),UNIT=SYSDA,
// DCB=(BLKSIZE=3120,LRECL=80,RECFM=FB),
// SPACE=(TRK,(1,1))
//SYSIN DD DSN=prefix.FOCCTL.DATA(FOCSVC),DISP=SHR
//LINK EXEC PGM=IEWL,PARM='RENT,XREF,LIST,'
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//OBJECT DD DSN=&&TEMP,DISP=(OLD,DELETE)
//SYSLMOD DD DSN=SYS1.LPALIB,DISP=SHR
//SYSLIN DD *
INCLUDE OBJECT
MODE AMODE(31),RMODE(24)
NAME IGC0024E(R)
/*
//

```

### 2.11.4 Post the FOCUS SVC number into zap control statements.

---

The AMASPZAP control statements distributed as member SVCZAP of FOCCTL.DATA must be modified to change the locally assigned FOCUS SVC number, referred to as FOCSVC, to the actual SVC number selected in Section 2.11.1. This can be done using the ISPF or TSO editor:

```
EDIT 'prefix.FOCCTL.DATA(SVCZAP)' DATA
VERIFY ON
CHANGE * 1000 /FOCSVC/0Ahh/ ALL
END SAVE
```

The symbol hh in the change string 0Ahh is the hexadecimal equivalent of the FOCUS SVC number. For example, if that number is decimal 245, the change string becomes 0AF5 (0=zero).

### 2.11.5 Zap the FOCUS SVC number into the basic FOCUS system.

---

The control statements modified in the step above are applied to the modules in FOCLIB.LOAD:

```
//SVCZAP EXEC PGM=AMASPZAP
//SYSPRINT DD SYSOUT=*
//SYSLIB DD DSN=prefix.FOCLIB.LOAD,DISP=OLD
//SYSIN DD DSN=prefix.FOCCTL.DATA(SVCZAP),DISP=SHR
```

This step is required even if you installed the FOCUS SVC earlier, for example, to use the BMP extension of the FOCUS/IMS-DB interface. If you omit this step, you can expect a system abend FED when you attempt to use the Simultaneous Usage facility.

### 2.11.6 Test use of the FOCUS SVC by the zapped modules.

---

Following the zap of the modules in FOCLIB.LOAD, their use of the FOCUS SVC can be tested. The test must be made online, using TSO:

```
FREE F(SYSIN,SYSPRINT)
ALLOC F(SYSIN) DA(*)
ALLOC F(SYSPRINT) DA(*)
CALL 'prefix.FOCLIB.LOAD(SVCTEST)'
```

You will receive one of three messages:

SVC IS OK

MODULE WAS NOT ZAPPED (remember SVC number?)

SVC DOES NOT WORK PROPERLY

The second message means that you failed to carry out the instructions in Sections 2.11.1 and 2.11.2. If you get the third message, review the installation of the FOCUS SVC as described in this section and make sure that you posted the correct hexadecimal SVC number. If the failure persists, call your local IBI representative or the Installation Services Department in New York at 1-800-736-6130.

## 2.12 Install the Simultaneous Usage (SU) Facility

---

This optional feature is described briefly in the first section of this manual and it is fully documented in the *Simultaneous Usage Reference Manual - TSO Version*. The FOCUS SVC must be installed before you install SU.

The installation steps are:

- Install either the IBI Subsystem or the FOCUS SVC.
- Allocate the communication dataset FOCSU.
- Create the auxiliary FOCUS database FOCUSSU.
- Create JCL for the Simultaneous Usage (SU) central database job.

If you wish to install the optional FOCUS/SU Security Interface, see Section 2.12.3.

### 2.12.1 Allocate the communication dataset FOCSU

---

The database job that performs the centralized FOCUS I/O runs in its own address space, and its presence in the system is made known through a communication dataset that will be accessed by all the TSO users and batch jobs it will service. This dataset plays a role only in the initial handshake; the subsequent transfer of commands and of data between the central database job and its client jobs or TSO IDs takes place entirely in virtual storage.

The communication dataset must be allocated and catalogued on a permanently mounted volume, and its name must be chosen so as to allow WRITE access by the central database job and READ access by any other TSO ID or batch job. The allocation is for a minimal amount of space for a single 16-byte fixed-length record. The actual dataset name is irrelevant but we recommend something suggestive of its role:

```
ATTR ADCB LRECL(16) RECFM(F) BLKSIZE(16)
ALLOCATE DA('SYS1.FOCSU.DATA') CATALOG SP(1) TRACK USING(ADCB)
```

It is possible that you will want to run several central database jobs, each dedicated to a set of FOCUS databases associated with a particular application. To do so, you must allocate a different communication dataset to each such job.

### 2.12.2 Create the auxiliary FOCUS database FOCUSU.

---

This is an auxiliary FOCUS database that will be accessed in READ mode by the central database job, but not by the TSO users or batch jobs it will service. The same auxiliary database can be used by all central database jobs, if you have more than one. The following process creates it:

```
ALLOCATE F(MASTER) DA('prefix.FOCCTL.DATA') SHR
ALLOCATE F(ERRORS) DA('prefix.ERRORS.DATA') SHR
ALLOCATE F(FOCUSSU) DA('SYS1.FOCUSSU.FOCUS') CATALOG SP(1) TRACKS +
    LRECL(4096) BLKSIZE(4096) RECFM(F)
CALL 'prefix.FOCLIB.LOAD(FOCUS)'
CREATE FILE FOCUSSU
FIN
```

### 2.12.3 Installing the FOCUS/SU Security Interface

---

The only installation step is the authorization of the FOCLIB.LOAD library. You or your MVS systems programmer will need to authorize the entire FOCLIB.LOAD library. This allows some of the modules in FOCLIB.LOAD to issue certain privileged operating system functions, including making security requests. The only module that actually issues these security requests is module SUSI. The HLISECUR module in FOCLIB.LOAD is the only module with a non-zero authorization code, meaning that HLISECUR is the only module which can be invoked directly as an authorized program. All other modules in FOCLIB.LOAD have an authorization code of zero, which means that they can run authorized only if they are invoked properly by HLISECUR.

The program HLISECUR simply links to the module HLISNK, which controls the sink machine. This program must run authorized so that the interface can later invoke the module SUSI to do the security access checking.

The program SUSI checks security access on behalf of the source user. This program runs under the authorization of HLISECUR in order to issue the RACROUTE macro.

The modules HLISECUR and SUSI are the only modules in FOCLIB.LOAD that must run authorized.

Source code for these modules is provided on the tape so that you may verify, if you wish, that these modules do not pose a system security risk when installed as authorized modules. The source code for HLISECUR and SUSI can be found in the FOCCTL.DATA library in members HLISESRC and SUSISRC.

## Using the Interface

---

To use the interface, insert the following command in the SU profile:

```
SET SUSI=ON
```

The SU profile is member HLIPROF of a PDS allocated to ddname FOCEXEC on the sink machine.

Once the Interface is installed, you must change all sink machine jobs to execute the program HLISECUR, rather than HLISNK, as in the following JCL line:

```
//SINK EXEC PGM=HLISECUR,PARM='parameters'
```

You can use the same password and ECHO or STAT parameters for HLISECUR as for HLISNK.

For more information see the *Simultaneous Usage Reference Manual, TSO Version*.



## 2.13 TSO Logon Procedures and Allocation CLISTs

---

FOCUS users can logon using any normal TSO logon procedure that defines a 512K virtual region and allows for at least 25 ddnames to be allocated dynamically. The files to be accessed can be allocated via TSO ALLOCATE commands issued from CLISTs before FOCUS is entered, or from FOCUS procedures invoked inside FOCUS. FOCUS work files are allocated automatically by FOCUS itself, if no previous allocation exists. In most cases there is no need for a STEPLIB: FOCUS can be invoked via the CALL command without any loss of function.

Nevertheless, we recommend that you construct a common TSO logon procedure for all FOCUS users. This is a convenient place in which to allocate the FOCUS error message library (file ERRORS), the library of user-supplied subroutines (file USERLIB), and most commonly used work files, with disk space adequate for all but extreme cases. The FOCUS databases and other data files read through FOCUS, as well as their descriptions (file MASTER) and associated FOCUS procedures (file FOCEXEC), are best allocated in TSO CLISTs tailored for each application. These allocation CLISTs are executed once, immediately after the logon, and FOCUS itself is entered via a separate CLIST, which contains nothing more than the CALL command. This arrangement minimizes repeated unnecessary online allocations, that cause noticeable delays and use extra resources.

If FOCUS loads programs that load further modules from an unspecified ddname, a STEPLIB will be required in the logon procedure (for example, if FOCUS invokes GDDM to produce 3270 type graphics).

A STEPLIB in the logon procedure will also be required if FOCUS is loaded from another program (for example, if a user invokes program DFSRRC00 with IMS, which loads FOCUS).

The following is a typical FOCUS logon procedure set up to support GDDM and access to IMS databases:

```
//LOGON      EXEC      PGM=IKJEFT01,REGION=1200K,DYNAMNBR=25
//STEPLIB    DD        DSN=prefix.FOCLIB.LOAD,DISP=SHR      For FOCUS    1
//          DD        DSN=IMSVS.RESLIB,DISP=SHR           For IMS
//          DD        DSN=gddmlibrary,DISP=SHR            For GDDM    3
//IMS        DD        DSN=IMSVS.PCBLIB,DISP=SHR,          For IMS    2
//          DD        DSN=IMSVS.DBDLIB,DISP=SHR           For IMS
//*
//*          FOCUS error messages and work datasets
//*
//ERRORS     DD        DSN=prefix.ERRORS.DATA,DISP=SHR
//FOCSTACK   DD        UNIT=SYSDA,SPACE=(TRK,(1,1))        Needed only if
//FOCSORT    DD        UNIT=SYSDA,SPACE=(TRK,(1,2))        default allocation
//HOLD       DD        UNIT=SYSDA,SPACE=(TRK,(1,2))        of (TRK,(5,5)) is
//HOLDMAST   DD        UNIT=SYSDA,SPACE=(TRK,(1,2,1))     not acceptable.
//OFFLINE    DD        SYSOUT=*
//*
//*          datasets for other TSO utilities
//*
```

STEPLIB entries 1 and 2 are required for IMS; entry 3 is needed for GDDM. Normal FOCUS usage does not need a STEPLIB.

All FOCUS work datasets except FOCSORT are good candidates for VIO. Notice that SYSIN and SYSPRINT need not be allocated; they will default to the TSO terminal.

A sample FOCUS CLIST to allocate all necessary datasets follows:

```

/*                                                                    */
/* Allocation CLIST - to be executed once                             */
/*                                                                    */
/* FOCUS file descriptions, procedures and the User program library. */
/* Assumes that ERRORS is allocated in the LOGON PROC.               */
/*                                                                    */
      ALLOC F(MASTER) DA('userid.MASTER.DATA') SHR
      ALLOC F(FOCEXEC) DA('userid.FOCEXEC.DATA') SHR
      ALLOC F(USERLIB) DA('prefix.FUSELIB.LOAD') SHR
      ALLOC F(WINFORMS) DA('prefix.WINFORMS.DATA') SHR

/*                                                                    */
/*      Communication to the central database job                     */
/*                                                                    */
      ALLOC F(FOCSU) DA('SYS1.FOCSU.DATA')      SHR

/*                                                                    */
/*      files and databases accessed by FOCUS                       */
/*                                                                    */
      ALLOC F(CARS) DA ('userid.CARS.FOCUS') SHR
      .
      .
      etc.

```

Finally, the CLIST for entering FOCUS should simply read:

```
CALL 'prefix.FOCLIB.LOAD(FOCUS)'
```

If the logon procedure has allocated FOCLIB.LOAD to the STEPLIB, users can enter FOCUS by simply typing FOCUS at the TSO READY level, rather than from a CLIST that issues the CALL command. However, we recommend that you do not have a STEPLIB in the logon procedure, unless it is mandated by the conditions discussed above, because it will be searched first for all load modules (including the TSO command modules).

If IMS databases are to be accessed in IMS batch mode, without the benefit of the BMP extension of the FOCUS/IMS-DB interface, FOCUS must be entered as shown here:

```
CALL 'IMSVS.RESLIB(DFSRRCOO)' 'DLI, FOCUS,psbname'
```

The FOCUS load library must be in STEPLIB.

## 3 Customizing FOCUS

---

### 3.1 Changes to FOCUS Defaults

#### 3.1.1 Changing the Number of Active FOCUS Databases Allowed

---

Starting with FOCUS Release 7.0, the default number of FOCUS databases that can be active at one time has been increased to 255 (the maximum). It is no longer necessary to increase the number of active FOCUS databases.

#### 3.1.2 Increased Field Buffer Size

---

The maximum size of any record handled by FOCUS is 16,384 bytes. The size of a record is determined by adding together the internal lengths of the fields in the file, after word alignment. Include all real fields, all virtual fields (cross-referenced or joined segment data), and all temporary fields (defined or computed fields).

It is possible to increase the record buffer from 16K bytes up to a maximum of 32K bytes in increments of 4K bytes. This feature is necessary for a few very large applications involving large numbers of temporary fields.

The number of Dialogue Manager variables (& variables) allowed is dependent on the size of the record buffers. The default size of 16K in FOCUS Release 7.0 allows 512 Dialogue Manager variables (about 40 of these are reserved by FOCUS). Each additional 4K added to the record buffer size will increase the number of Dialogue Manager variables by 128, allowing a maximum of 1024 variables when the record buffer is set to 32K.

The total length of all Dialogue Manager variables must also fit within the record buffer size. With 16K buffers, we could have 512 variables of 32 bytes each or only 163 variables where the average length was 100 bytes (16300 bytes used of the total of 16\*1024).

Note that the buffer size of LINREC and of DATREC must be the same and may be any multiple of 4K between 16K and 32K.

### Installation

---

1. Edit the member LINREC of dataset FOCCTL.DATA and change the line that reads:

```
DS XL(16*1024) CURRENT SIZE IS 16K
```

replacing the 16 (for 16K bytes) with any multiple of 4 (e.g. 20, 24) up to a maximum of 32. For instance:

```
DS XL(20*1024) CURRENT SIZE IS 20K
```

2. Assemble the member LINREC from FOCCTL.DATA. The following is sample JCL for this:

```
//ASSEM      EXEC  PGM=IEV90 , PARM= 'LIST,OBJ,NOALIGN'
//SYSLIB     DD    DISP=SHR,DSN=SYS1.MACLIB
//SYSUT1     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT2     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT3     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSPRINT   DD    SYSOUT=*
//SYSPUNCH   DD    DUMMY
//SYSLIN     DD    DISP=SHR,DSN=prefix.OBJ(LINREC)
//SYSIN      DD    DISP=SHR,DSN=prefix.FOCCTL.DATA(LINREC)
```

3. Edit the member DATREC of dataset FOCCTL.DATA and change the line that reads:

```
DS XL(16*1024)    CURRENT SIZE IS 16K
```

replacing the 16 (for 16K bytes) with any multiple of 4 (e.g. 20, 24) up to a maximum of

32. For instance:

```
DS XL(20*1024)    CURRENT SIZE IS 20K
```

4. Assemble the member DATREC from FOCCTL.DATA. The following is sample JCL for this:

```
//ASSEM      EXEC  PGM=IEV90 , PARM= 'LIST,OBJ,NOALIGN'
//SYSLIB     DD    DISP=SHR,DSN=SYS1.MACLIB
//SYSUT1     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT2     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT3     DD    UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSPRINT   DD    SYSOUT=*
//SYSPUNCH   DD    DUMMY
//SYSLIN     DD    DISP=SHR,DSN=prefix.OBJ(DATREC)
//SYSIN      DD    DISP=SHR,DSN=prefix.FOCCTL.DATA(DATREC)
```

5. Link these new programs into FOCUS. The following is sample JCL for this:

```
//LINKFOC    EXEC  PGM=IEWL, PARM= 'LET,NCAL,LIST,SIZE=1024K'
//SYSPRINT   DD    SYSOUT=*
//SYSUT1     DD    UNIT=SYSDA,SPACE=(CYL,(10,1))
//SYSLMOD    DD    DISP=OLD,DSN=prefix.FOCLIB.LOAD    <- FOCUS
```

```

//MAINTAIN DD  DISP=SHR,DSN=prefix.FOCCTL.DATA  <- FOCCTL
//OBJECT DD  DISP=SHR,DSN=prefix.OBJ           <- Assembled code
//SYSLIN DD  *
        INCLUDE OBJECT(LINREC) <----- Assembled LINREC
        INCLUDE OBJECT(DATREC) <----- Assembled DATREC
        INCLUDE SYSLMOD(FOCUS) <----- Module to be Changed
        INCLUDE MAINTAIN(FOCUS) <----- Linkedit Control Statements
        NAME FOCUS(R)          <----- New Module

/*

```

Note that the buffer size of LINREC and of DATREC must be the same and may be any multiple of 4K between 16K and 32K.

### 3.1.3 Setting the Maximum Number of Cache Pages

---

FOCUS sites may limit the maximum number of cache pages allowed per FOCUS session. The limit may be set during the installation of FOCUS to a value between 3 and 524,288 pages (the hardware maximum), where each cache page is 4096 bytes in length. The default limit is 0 pages. The default limit for HiperFOCUS is 256. To change the limit on the maximum number of cache pages that can be requested, follow the installation instructions below.

An individual user or batch job may specify cache pages up to one less than the site-selected maximum with the SET CACHE command:

```

SET CACHE = <0>
           <n>

```

where:

- 0 Allocates no space to cache; this is the default. Cache is inactive.
- n Is the number of 4K-byte pages of contiguous storage allocated to cache memory. The minimum is two pages. The maximum value is 524287, or one less than the site-selected maximum that was specified during installation.

In normal use, the number of pages requested by users should be far less than the hardware maximum; otherwise the paging subsystem may be unfavorably affected.

## Installation

---

1. After unloading the FOCUS distribution tape, edit the member CCHCOM within the FOCCTL.DATA library, to specify your site's new maximum number of cache pages. Do this by locating the following line:

```

MAXPAGES EQU 524288

```

replacing 524288 with the maximum cache pages you want to allow. This value of MAXPAGES must be a value from 3 to 524288. Do not change any other lines in the file. After you make the change to MAXPAGES, save the member CCHCOM back into the dataset FOCCTL.DATA.

2. Assemble the member CCHCOM from FOCCTL.DATA. The following is sample JCL for this:

```
//ASSEM      EXEC PGM=IEV90,PARM='LIST,OBJ,NOALIGN'
//SYSLIB    DD   DISP=SHR,DSN=SYS1.MACLIB
//SYSUT1    DD   UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT2    DD   UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSUT3    DD   UNIT=SYSDA,SPACE=(TRK,(30,10))
//SYSPRINT  DD   SYSOUT=*
//SYSPUNCH  DD   DUMMY
//SYSLIN    DD   DISP=SHR,DSN=prefix.OBJ(CCHCOM)
//SYSIN     DD   DISP=SHR,DSN=prefix.FOCCTL.DATA(CCHCOM)
```

3. Link this new program into FOCUS. The following is sample JCL for this:

```
//LINKFOC   EXEC PGM=IEWL,PARM='LET,NCAL,LIST,SIZE=1024K'
//SYSPRINT  DD   SYSOUT=*
//SYSUT1    DD   UNIT=SYSDA,SPACE=(CYL,(10,1))
//SYSLMOD   DD   DISP=OLD,DSN=prefix.FOCLIB.LOAD   <- FOCUS
//MAINTAIN  DD   DISP=SHR,DSN=prefix.FOCCTL.DATA   <- FOCCTL
//OBJECT    DD   DISP=SHR,DSN=prefix.OBJ          <- Assembled code
//SYSLIN    DD   *
            INCLUDE OBJECT(CCHCOM) <----- Assembled CCHCOM
            INCLUDE SYSLMOD(FOCUS) <----- Module to be Changed
            INCLUDE MAINTAIN(FOCUS) <----- Linkedit Control Statements
            NAMES FOCUS(R)          <----- New Module
/*
```

4. If you are running HLI or SU, you must also link this new program into the HLI module. The following is sample JCL for this:

```
//LINKFOC   EXEC PGM=IEWL,PARM='REUS,LET,LIST,NCAL,SIZE=1024K'
//SYSPRINT  DD   SYSOUT=*
//SYSUT1    DD   UNIT=SYSDA,SPACE=(CYL,(10,1))
//SYSLMOD   DD   DISP=OLD,DSN=prefix.FOCLIB.LOAD   <- FOCUS
```

```

//MAINTAIN DD  DISP=SHR,DSN=prefix.FOCCTL.DATA  <- FOCCTL
//OBJECT DD   DISP=SHR,DSN=prefix.OBJ          <- Assembled code
//SYSLIN DD   *
            INCLUDE OBJECT(CCHCOM) <----- Assembled CCHCOM
            INCLUDE SYSLMOD(HLI)   <----- Module to be Changed
            INCLUDE MAINTAIN(HLI)  <----- Linkedit Control Statements
            NAMES HLI(R)           <----- New Module
/*

```

5. If you are running SU, you must also link the new program CCHCOM into the HLISNK module. The following is sample JCL for this:

```

//LINKFOC EXEC PGM=IEWL,PARM='LET,NCAL,LIST,SIZE=1024K'
//SYSRINT DD   SYSOUT=*
//SYSUT1 DD   UNIT=SYSDA,SPACE=(CYL,(10,1))
//SYSLMOD DD   DISP=OLD,DSN=prefix.FOCLIB.LOAD  <- FOCUS
//MAINTAIN DD  DISP=SHR,DSN=prefix.FOCCTL.DATA  <- FOCCTL
//OBJECT DD   DISP=SHR,DSN=prefix.OBJ          <- Assembled code
//SYSLIN DD   *
            INCLUDE OBJECT(CCHCOM) <----- Assembled CCHCOM
            INCLUDE SYSLMOD(HLISNK) <----- Module to be Changed
            INCLUDE MAINTAIN(HLISNK)<----- Linkedit Control Statements
            NAMES HLISNK(R)        <----- New Module
/*

```



### 3.1.4 Default Space Allocation Table for Work Files: TABLA

---

FOCUS output datasets not allocated by the user are allocated dynamically by FOCUS itself. The default space attributes associated with each dynamically allocated ddname are now set by editing the member IBITABLA in the PDS called FOCCTL.DATA.

In order to change the defaults for these FOCUS output datasets, the file IBITABLA must be copied to a dataset allocated to the DDname ERRORS. The file is a fixed columnar file. All changes must be made in the appropriate column. The columns are:

Column name	Starting column	length
DDname	01	8
Allocation units ( <u>CYLS</u> ,TRKS)	10	4
Primary space	15	3
Secondary space	19	3
Number of Directory entries (PDS)	23	2
Sysout class (OFFLINE only)	26	1
Volume	28	6
Unit (SYSDA,DASD,HIPER,NOHIPER*,etc)	35	8
Unit Count (FOCPOOLT only)	44	2

\* Use NOHIPER as a unit name to exclude particular datasets from HiperFOCUS.

If the file IBITABLA is not available in a PDS allocated to the ddname ERRORS, the defaults are in affect for all FOCUS output datasets. The default for all FOCUS output datasets except FOCPOOLT is 5 CYLs with secondary extent size of 5. The default for FOCPOOLT is 5 CYLs with secondary extent size of 20.

---

This is a copy of IBITABLA as shipped.

```

* DDNAME*A.UN*SP1*SP2*DR*C*VOLUME* UNIT *UC* FIELD NAME
*0-----1-----1-----1---2--2-----3-----4-* STARTING
*1-----0-----5---9---3--6-8-----5-----4-* COLUMN
*---8-----4-- -3- -3- 2- 1 --6--- --8--- 2-* LENGTH
HOLD CYLS 5 5 , /* 1 */
HOLDMAST TRKS 5 5 36 , /* 2 */
SAVE CYLS 5 5 , /* 3 */

```

```

REBUILD  CYLS  5  5          , /* 4 */
FOCSML   CYLS  5  5          , /* 5 */
FOCUS    CYLS  5  5          , /* 6 */
FOCSTACK TRKS  5  5          , /* 7 */
FOCSORT  CYLS  5  5          , /* 8 */
OFFLINE  CYLS  5  5  A      , /* 9 */
SESSION  TRKS  5  5          , /*10 */
FOCCOMP  TRKS  5  5  12     , /*11 */
HOLDACC  TRKS  5  5  12     , /*12 */
FMU      TRKS  5  5  12     , /*13 */
TRF      TRKS  5  5  12     , /*14 */
FOCPOOLT CYLS  5  20       2, /*15 */
*
*
* The UC or unit count column may be specified for FOCPOOLT only.
***** Bottom of Data *****

```

### 3.1.5 Default System File Naming Conventions

---

To take advantage of the automatic allocation facility of FOCUS, certain naming conventions must be followed. These may be adjusted to your particular needs at the time of the FOCUS installation.

The following is a list of default naming conventions:

prefix.FOCEXEC.DATA	For FOCEXEC
prefix.FOCCOMP.DATA	For compiled MODIFY procedures
prefix.MASTER.DATA	For MASTER files
prefix.ddname.DATA	For data files
prefix.MODEL.DATA	For FOCCALC spreadsheets
SYS1.ADMCFORM	For ICU format files
SYS1.ADMCDATA	For ICU data files
prefix.TTEDIT.DATA	For TABLETALK saved sessions
prefix.FMU.DATA	For FMU files
prefix.TRF.DATA	For TRF files
prefix.WINFORMS.DATA	For online MAINTAIN applications

The prefix is selected based on the following rules:

- If the FOCUS SET PREFIX command was issued before, then the set value is selected.

- If no SET PREFIX command was issued and FOCUS is executed under TSO, then the value of the standard TSO prefix (as set by the TSO PREFIX command) is selected.

## NOTES:

- Existing FOCUS files must be allocated before using them. The CREATE command allocates a FOCUS database to a temporary dataset.
- FOCUS does not attempt to dynamically allocate a dataset already allocated via a CLIST, JCL, user-issued TSO ALLOC, or FOCUS DYNAM command.
- The naming conventions may be set up so that for some or all of the mentioned datasets, the prefix will be set to a fixed word (e.g., SYS1), in case the SET PREFIX command was not issued.
- If for any reason FOCUS fails to allocate a dataset (e.g., the dataset is not catalogued), it will issue the customary error message indicating that the file requested cannot be found. The interactive user will still have the opportunity to allocate the right dataset via the TSO ALLOC or FOCUS DYNAM command.
- For MVS batch execution, since a standard prefix does not exist in this environment, the SET PREFIX command must be issued if any of the datasets whose prefix is not set to a fixed value needs to be dynamically allocated.
- The facility may be disabled for any or all datasets mentioned.

## Modifying Naming Conventions for Automatic Allocation: DSNTAB

---

The table of naming conventions for datasets which can be dynamically allocated can be adjusted to your particular needs at the time of FOCUS installation. The FOCUS database is no longer dynamically allocated. The ERRORS PDS must be allocated prior to the execution of FOCUS.

The current table can be found as member DSNTAB in FOCCTL.DATA. You can adjust this table by editing, compiling, and link-editing it into FOCUS. You can also adjust this table by using the AMASPZAP utility.

DSNTAB contains information for dynamic allocation of MASTER, FOCEXEC, FIXFORM, and other files in MVS:

```

DSNTAB  CSECT
* 1  MASTER
           DC C'PREFIX  MASTER.DATA           1'
* 2  FOCEXEC
           DC C'PREFIX  FOCEXEC.DATA          1'
* 3  FOCUS Database

```

```

          DC C'PREFIX  DATABASE.FOCUS                1'
* 4  FIXFORM File
          DC C'PREFIX  DDNAME.DATA                  1'
* 5  Errors
          DC C'SYS1    FOCUS.ERRORS.DATA            1'
* 6  Set_prefix_value - 8 characters long
          DC C'                                           '
* 7  FOCCOMP File
          DC C'PREFIX  FOCCOMP.DATA                  1'
* 8  FOCCALC Spreadsheet
          DC C'PREFIX  MODEL.DATA                    1'
* 9  ICU Format File
          DC C'SYS1    ADMCFORM                       1'
* 10 ICU Data File
          DC C'SYS1    ADMCDATA                       1'
* 11 TTEDT TableTalk saved session PDS
          DC C'PREFIX  TTEDIT.DATA                    1'
* 12 FMU
          DC C'PREFIX  FMU.DATA                       1'
* 13 TRF
          DC C'PREFIX  TRF.DATA                       1'
* 14 WINFORMS
          DC C'PREFIX  WINFORMS.DATA                  1'
*
          END

```

Each entry in the preceding table is 53 bytes long and has the following format:

prefix	datasetname	flag
0	8	52

where:

- |             |  |
|-------------|--|
| prefix      | Is an 8 character field beginning at byte 0. If it contains the word PREFIX, the TSO prefix will be substituted for the prefix value; otherwise, the prefix will be whatever is entered in this field. If the SET PREFIX command is issued, it has a priority and the SET value will be substituted. |
| datasetname | Is a 44 character field beginning at byte 8. Datasetname must contain the reserved word DATABASE for FOCUS files and DDNAME for the FIXFORM files needed to be allocated. These reserved words can appear anywhere in the 44 character string.   |
| flag        | Is a 1 character field beginning at byte 52 which indicates whether the DDNAME should be dynamically allocated or not. If FLAG=1 then FOCUS reads that the file should be allocated. If FLAG=0 then FOCUS reads that the file should not be allocated.   |

NOTE: In the 44-character datasetname field above, the low-level qualifiers must remain as defined in the DSNTAB table(except in the case of FOCUS and FIXFORM files). For example, the dataset prefix.TTEDIT.DATA may be additionally qualified as prefix.xxxx.TTEDIT.DATA, but not as prefix.xxxx.TTEDIT. In order to use a variant form of the low level qualifiers, the DSNTAB table would either have to be edited, compiled and linked, or zapped as stated previously.

Note: Changes to entries 3 and 5 will have no affect.

**DSNTAB Installation:**

---

1. Change member DSNTAB of FOCCTL.DATA.
2. Assemble it.
3. Link edit the new DSNTAB into FOCUS:

```
//LINK1 EXEC   PGM=IEWL,PARM='LET,NCAL,LIST,MAP,SIZE=1024K'
//SYSPRINT DD  SYSOUT=*
//SYSUT1 DD   UNIT=SYSDA,SPACE=(CYL,(10,1))
//OLDMOD DD   DSN=prefix.FOCLIB.LOAD,DISP=SHR
//DSNTAB DD   DSN=assembled.dsntab,DISP=SHR
//MAINTAIN DD  DSN=prefix.FOCCTL.DATA,DISP=SHR
//SYSLMOD DD  DSN=prefix.TEST.FOCLIB.LOAD,DISP=SHR
//SYSLIN DD   *

INCLUDE DSNTAB
INCLUDE OLDMOD(FOCUS)           <---- MODULE TO BE CHANGED
INCLUDE MAINTAIN(FOCUS)        <---- LINK1 CNTL STATEMENTS
NAME FOCUS(R)                   <---- NEW MODULE
```

### 3.1.6 Modifying the Efficiency List of TSO Commands in FOCUS: TSOTAB

---

FOCUS maintains a list of commonly executed TSO commands. For efficiency, commands on this list are assumed valid without verification and are passed control when called. Commands not on this list may still be run, but are searched for through the STEPLIB, JOBLIB, and linklist libraries using the BLDL macro.

The current list can be found as member TSOTAB on FOCCTL.DATA. Users can add to it by editing, compiling, and link-editing it into FOCUS. JCL for this is included.

The order of the commands is irrelevant, with the list terminated by the first string of 8 bytes of hex FF.

The TIME command should not be included in this list.

### TSOTAB Installation

---

The following JCL allows compilation and link-editing of TSOTAB into FOCUS.

```
//LINK1 EXEC   PGM=IEWL,PARM='LET,NCAL,LIST,MAP,SIZE=1024K'
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(10,1))
//OLDMOD DD DSN=prefix.FOCLIB.LOAD,DISP=SHR
//TSOTAB DD DSN=assembled.tsotab,DISP=SHR
//MAINTAIN DD DSN=prefix.FOCCTL.DATA,DISP=SHR
//SYSLMOD DD DSN=prefix.TEST.FOCLIB.LOAD,DISP=SHR
//SYSLIN DD *
           INCLUDE TSOTAB
           INCLUDE OLDMOD(FOCUS)           <---- MODULE TO BE CHANGED
           INCLUDE MAINTAIN(FOCUS)        <---- LINK1 CNTL STATEMENTS
           NAME FOCUS(R)                   <---- NEW MODULE
```

### 3.1.7 Resetting the Maximum Allowed Data Exceptions

---

FOCUS by default allows for ten data exception interrupts to occur before terminating with the following error:

```
(FOC204) TOO MANY INTERRUPTS. RUN TERMINATED.
```

This imposed limit of ten may be changed by using the AMASPZAP utility to zap the necessary bytes in CSECT LIMIT in FOCUS TEXT. For example, to change the allowable data exception limit from 10 to 20, zap the existing limit of x'0000000A' to x'00000014'. Verify the contents of the bytes to be changed before zapping. The following is sample JCL for this:

```
//STEP1 EXEC PGM=AMASPZAP
//SYSLIB DD DSN=prefix.FOCLIB.LOAD,DISP=OLD
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
NAME FOCUS LIMIT <--- Specify the module and CSECT.
VER 00 0000000A <--- Verify that pos 0 contains x'0000000A'.
REP 00 00000014 <--- Replace fullword with x'00000014'.for 20
/*
```

REP will replace the fullword at LIMIT with the new limit.

If you want unlimited data exceptions, change the allowable data exception limit to zero.

A version of FOCUS can have only one limit. If you need more than one limit, more than one copy of FOCUS must be created, each having a different value for the limit.

The aforementioned procedure must be performed each time a new version or Release of FOCUS is installed.



## 3.2 Installing HiperFOCUS for MVS/ESA

---

This section describes the installation of HiperFOCUS for MVS/TSO for FOCUS Release 7.0 or a subsequent release. It also discusses the new HiperBUDGET facility of HiperFOCUS.

### 3.2.1 Pre-Installation Requirements

---

The HiperFOCUS installation can be performed by anyone with a working knowledge of MVS/TSO. Some FOCUS knowledge is required to restrict options or resource usage. FOCUS must be installed prior to installing HiperFOCUS. It is also necessary to ensure that your processor is running MVS/ESA as the operating system since HiperFOCUS cannot be installed under the MVS/XA environment.

The hardware and software requirements for the HiperFOCUS option are:

- \* MVS/ESA V3 or higher.
- \* MVS/DFP V3.1 or higher.
- \* IBM ES/3090S, ES/3090J, ES/3090JH, ES/3090T, any ES/9000 model 9021, or an Amdahl or Hitachi compatible.

If you are installing HiperFOCUS with MVS/ESA V4.2.0, V4.2.2, or V4.3.0, please contact IBM for APAR OY61461.

### 3.2.2 HiperFOCUS Installation

---

HiperFOCUS can now be installed and customized by the use of SET commands in the FOCPARM member of ERRORS.DATA. You may install HiperFOCUS by just adding SET HIPERINSTALL=ON to the FOCPARM entries and accept the defaults. However, because there would be no limit to the number of hiperspaces that FOCUS can create, it is strongly recommended that you set limits via the SET parameters listed below and in the following section on the IBI Subsystem. Together, they comprise the HiperBUDGET facility of HiperFOCUS.

Here is a list of the SET parameters available for HiperFOCUS:

```
SET HIPERINSTALL=ON/OFF
```

where

- |     |                          |
|-----|--------------------------|
| ON  | Will install HiperFOCUS. |
| OFF | Will disable HiperFOCUS. |

SET HIPERSPACE=nnn

where

nnn Is the number of (4K) pages for aggregate hiperspace size. By default there is no limit. This parameter is equivalent to SET TCBLIM=nnn under the IBI Subsystem that is discussed in the next section. If both parameters are used and different limits are set, then the lower limit will be enforced.

SET HIPERFILE=nnn

where

nnn Is the maximum number of (4K) pages for an individual hiperspace size. The default is 524288 (4K) pages or 2 GB in size. This parameter is equivalent to SET FILELIM=nnn under the IBI Subsystem that is discussed in the next section. If both parameters are used and different limits are set, then the lower limit will be enforced.

SET HIPERCACHE=nnn

where

nnn Is the number of hipercache (4K) pages. The default is 256 (4K) pages or 1M in size.

SET HIPEREXTENTS=nnn

where

nnn Is the number of extend operations to be performed. The default number is 127.

SET HIPERLOCKED=ON/OFF

where

ON Disallows processing of user interface commands.

OFF Allows processing of user interface commands (such as SET HIPERFOCUS=ON/OFF). Default.

```
SET HIPERFOCUS=ON/OFF
```

where

ON Will activate a HiperFOCUS session after HiperFOCUS has been installed. Default.

OFF Will deactivate a HiperFOCUS session.

All of the SET options described above (except for SET HIPERFOCUS) can only be executed from the FOCARM member of the ERRORS file. Any attempt to execute these commands from within a FOCUS session will result in a FOC964 error.

### 3.2.3 HiperBUDGET and the IBI SUBSYSTEM

---

HiperBUDGET and the IBI Subsystem are two new features introduced in Release 7.0 . The IBI Subsystem is an optional replacement of the IBI SVC that provides communication among address spaces running IBI products on the same MVS system. HiperBUDGET uses this subsystem to regulate and report the overall use of hiperspace on that system. It accomplishes this through the enforcement of pre-defined limits on hiperspace consumption at the system, server, user and file levels whereby the limits set at the lower levels cannot exceed those set at a higher level.

These are the limits that can be set for the IBI Subsystem:

```
SET MVSLIM = nnn/-1
```

where

nnn Is the maximum number of (4k) pages of hiperspace for all of IBI's products on that MVS system.

-1 Specifies that there is no limit checking.

```
SET SERVLIM = nnn/-1
```

where

nnn Is the maximum number of (4k) pages of hiperspaces allowed for multiple users on a per server basis.

-1 Specifies that there is no limit checking.

```
SET TCBLIM = nnn/-1
```

where

- nnn Is the maximum number of (4K) pages available for each individual user.
- 1 Specifies that there is no limit checking.

This parameter is equivalent to SET HIPERSPACE=nnn described in the previous HiperFOCUS installation section. If both parameters are specified, then the smaller value takes precedence.

```
SET FILELIM = nnn/-1
```

where

- nnn Is the maximum number of 4K pages of hiperspace on an individual file basis.
- 1 Specifies that there is no limit checking.

This parameter is equivalent to SET HIPERFILE=nnn under the IBI Subsystem that is discussed in the next section. If both parameters are specified, then the smaller value takes precedence.

```
? HBUDGET
```

A new query has been added to report on the limits specified and the actual utilization of hiperspace. Among the statistics displayed will be the limits set at the system, server, user and file levels, the number of busy pages, the number of hiperextents allowed, and the ddname and size of files allocated in hiperspace or "spilled" to disk.

```
>? HBUDGET
```

```
Total system      limit is not set
Total server       limit is not set
Total hiperspace  limit is not set
Single file size  limit is      524288 pages
Total amount of busy pages is      616 pages
Number of extents is set to      127
```

---

```
DDname :Reserved :Hiperspace : Spilled :Spill DDn
```

### 3.3 Installing the FOCUS Menu

---

FOCUS Release 7.0 includes the FOCUS Menu, a convenient way to access FOCUS facilities using windows. A few installation steps are required to enable the menu.

The installation process for the FOCUS Menu should be performed at the same time MVS FOCUS is installed. If you are installing MSO FOCUS, refer to the *FOCUS for IBM Mainframe Multi-Session Option Installation and Technical Reference Guide, Release 7.0*.

1. Allocate the FMU.DATA dataset. The dataset FMU.DATA requires the following allocations in addition to the allocations made in Chapter 2, Step 1:

Dataset	Allocation Parameters	Disk Type trk
FMU.DATA	Primary Allocation Secondary Allocation Directory Blocks  (LRECL=4096, BLKSIZE=12288, RECFM=FB)	115 15 10

2. Modify the FOCLOAD JCL.

Add the following allocations to the corresponding allocations in the JCL used to unload the distribution tape found in Chapter 2, "Unload the FOCUS Distribution Tape". FMU.DATA is file 40 on the distribution tape.

```
//I40      DD      DSN=FMU.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//
//          VOL=SER=volid,LABEL=(40,SL)
```

```
//O40      DD      DSN=prefix.FMU.DATA,DISP=OLD
```

```
COPY      INDD=I40,OUTDD=O40
```

3. In Chapter 2 perform Steps titled "Install the FOCUS Site Code" and "Test the Basic FOCUS System". For MSO FOCUS, also perform all of the installation steps in the *FOCUS for IBM Mainframe Multi-Session Option Installation and Technical Reference Guide, Release 7.0*.

4. To enable the FOCUS Menu, edit member SHELPROF in FOCEXEC.DATA, which was unloaded from the distribution tape.

The FOCEXEC contains the following commands:

```
-* This FOCEXEC will be executed only after PROFILE FOCEXEC
-* execution has been completed.
-* INCLUDE FMMAIN
```

Remove the comment from the line with the INCLUDE command.

```
-INCLUDE FMMAIN
```

5. Test the FOCUS Menu by entering FOCUS with the following allocations:

```
DYNAM FREE FILE MASTER FOCEXEC FMU CAR ERRORS
DYNAM ALLOC FILE MASTER DA prefix.MASTER.DATA SHR REUSE
DYNAM ALLOC FILE FOCEXEC DA prefix.FOCEXEC.DATA SHR REUSE
DYNAM ALLOC FILE FMU      DA prefix.FMU.DATA SHR REUSE
DYNAM ALLOC FILE CAR      DA prefix.CAR.FOCUS SHR REUSE
```

The minimal allocation for testing the FOCUS Menu requires ddnames FOCEXEC, FMU, and ERRORS. Allocate MASTER and a database (CAR in this example) to test specific FOCUS Menu facilities. Any of the ddnames in the example above may have concatenated datasets.

NOTE: In MSO FOCUS, ERRORS should not be freed and re-allocated.

The FOCUS Menu will appear.

The cursor should be positioned next to the first option on the menu, "Next, Next Menu Option".

You can exit FOCUS by pressing PF3 to go to the Quit Menu.

Move the cursor next to the last option on the menu, "Quit FOCUS", and press ENTER.

6. Continue the installation of MVS FOCUS with, "Install High-Resolution Graphics for 3270 Type Terminals".

## 3.4 Installing the FOCUS ToolKit

---

The FOCUS ToolKit is a convenient way to access FOCUS facilities using windows. A few installation steps are required to enable the menu.

The installation process for the FOCUS ToolKit should be performed at the same time MVS FOCUS is installed.

1. Allocate the FMU.DATA dataset. This dataset requires the following allocation parameters and disk space in addition to the allocations made in Chapter 2:

Dataset	Allocation Parameters	Disk Type trk
FMU.DATA	Primary Allocation Secondary Allocation Directory Blocks  (LRECL=4096, BLKSIZE=12288, RECFM=FB)	115 15 10

2. In "Unload the FOCUS Distribution Tape", add the following commands to the appropriate sections of the JCL used to unload the tape. FMU.DATA is file 40 on the distribution tape.

```
//I40      DD      DSN=FMU.DATA,DISP=(OLD,PASS),UNIT=TAPE,
//
//          VOL=SER=valid,LABEL=(40,SL)
.
.
.
//O40      DD      DSN=prefix.FMU.DATA,DISP=OLD
.
.
.
COPY      INDD=I40,OUTDD=O40
```

3. Continue the installation of MVS FOCUS with "Install the FOCUS Site Code". For MSO FOCUS, also perform all of the installation steps in *FOCUS for IBM Mainframe Multi-Session Option Installation and Technical Reference Guide, Release 7.0*.

4. Create a partitioned dataset with an LRECL of 80. This dataset will be allocated to ddname ISHFALOC during installation and execution of the ToolKit. All users must have read access to this dataset when executing the ToolKit. This dataset will contain a site-specific background for the ToolKit (member ISHFPROF), user selections for print destination (member ISHDPNTR), and customized file allocation FOCEXECs. The following provides recommendations for dataset naming conventions and DCBs.

```
ALLOC F(ISHFALOC) DA('prefix.ISHFALOC.DATA') NEW CATALOG -
SPACE(5,5) TRACKS LRECL(80) RECFM(F B) BLKSIZE(23440) DIR(30)
```

NOTE: The ToolKit can be customized for a user group by creating a separate ISHFALOC dataset for each group. Step 5 must be executed for each of these datasets.

5. Customize the ToolKit with site-specific information.

To prepare the background window for the ToolKit, you will be asked to provide the company name, a support center phone number, the installation date, the IBI FOCUS Site Code, a default printer destination, and a list of optional printer destinations. The company information will be displayed on the background of the ToolKit Menuing system. The printer information will be used to assign a default printer when first executing the ToolKit and allow the users to easily change the destination of printed output. You will be asked for this information in simple fill-in the blank screens.

- 5a. Enter FOCUS with your FOCUS CLIST.
- 5b. Execute the ToolKit Installation Procedure.

```
EX ISHFINST
```

Enter company name at the prompt and press ENTER.

Enter telephone number for support at the prompt and press ENTER.

Enter install date (in MM/DD/YY format) at the prompt and press ENTER.

Enter the FOCUS site code at the prompt and press ENTER.

Enter the default printer at the prompt and press ENTER.

The format for the printer prompt is: 'c (node.)dest', where 'c' is the one character sysout class, node is the JES printer node, and dest is the JES destination parameter. The node and dest parameters are optional; when provided, a single blank must separate them from the sysout class.



Reply to the prompt asking for additional company printer IDs that can be selected by the user. If the reply is no, installation is complete. If the reply is yes, enter the additional printer(s) in the format: one character sysout class, 17 character node and destination (node.dest where node is optional), and 30 character description. The sysout class cannot be blank or zero (you may delete entries by blanking out the sysout class). Note that the first printer in the list will be assigned as the user's default printer. You may also designate a dataset as the output by providing an arbitrary sysout class, a destination code of 'DISKFILE', and the low level qualifiers of the dataset name as the description. The user's userid will be prefixed to this name. The dataset must be typed in capital letters, as it will not be translated. Up to 18 printers can be added at this time, including the default printer. Changes can be made after the installation by allocating the ISHFALOC dataset and executing FOCEXEC ISHFNPTR, or by editing the member ISHDPNTR in the ISHFALOC dataset (be sure to maintain columnar spacing).

Upon completion of the installation, the ToolKit will be executed. Move the cursor next to the last option on the menu, "Exit", and press ENTER.

6. Provide batch submission capability.

The ToolKit provides capability for batch submission of FOCEXECs by the user. The JCL submitted must be customized for site dataset naming and JOBCARD standards.

6a. Edit member ISHFBJCL in FOCEXEC.DATA.

Replace the jobcard information with a site-specific jobcard. The variable &&USER, which is the userid set in the ToolKit startup program (ISHFSHLL), may be used in the jobcard.

Replace the dataset names for STEPLIB, ERRORS, FOCEXEC, MASTER, and FOCSQL (where appropriate), with the site-specific dataset names for the FOCUS datasets.

6b. Edit member ISHFBTCH in FOCEXEC.DATA.

Delete the following lines that preclude batch execution until the JCL has been updated in the previous

```
-SET &&ERROR='BATCH EXECUTION NOT INSTALLED-SEE DOCUMENTATION' ;
-GOTO EXIT
```

To provide direct submission of the generated JCL in TSO, remove the comment characters (-\* ) from the following lines:

```
-* TSO SUBMIT BATCHJCL
-* -GOTO DONE
```

Otherwise, the JCL will be displayed in TED. The user may then submit the batch job by typing SUBMIT on the TED command line.

7. Disable the COPY function.

The ToolKit MVS Utilities and FOCUS Utilities menus support copying of datasets and members in TSO (but not MSO) by utilizing the TSO COPY command. If your site supports the COPY command, proceed to Step 8. If not, you will have to disable this function.

7a. Edit member ISHFEDTF in FOCEXEC.DATA.

After the -COPY label, insert the following lines:

```
-SET &&ERROR='COPY option is not available at -company name-';
-GOTO STRT
```

7b. Edit member ISHFTUTL in FOCEXEC.DATA.

After the -PDSCOPY label, insert the following lines:

```
-SET &&ERROR='COPY option is not available at -company name-';
-GOTO TOP
```

8. Customize user program dataset allocations.

Many of the ToolKit options allow the user to write to their own datasets. To that end, the ToolKit assumes that user files follow the suggested IBI dataset naming convention:

```
userid.FOCEXEC.DATA
userid.MASTER.DATA
userid.FOCDEF.DATA    (optional)
userid.FOCSQL.DATA   (optional)
userid.filename.FOCUS
    (where filename is the MFD name that describes the FOCUS data)
```

If your site follows these conventions, continue with Step 9. Otherwise, you may alter these assumptions as follows:

8a. Change high-level qualifier only.

Edit ISHFSHLL in FOCEXEC.DATA. Change the following line to return the desired value (the default is userid).

```
-SET &&USER=GETUSER('A8');
```

NOTE: If &&USER was used in ISHFBJCL in JOBCARD processing (Step 6), modifications must be made in that program as well.

## 8b. Change the low-level qualifiers.

Edit ISHFSHLL in FOCEXEC.DATA.

Change the following lines to return the desired value.

```
-SET &DEFMAS = &&USER || '.MASTER.DATA' ;
-SET &DEFFEX = &&USER || '.FOCEXEC.DATA' ;
-SET &DEFSQL = &&USER || '.FOCSQL.DATA' ;
-SET &DEFDEF = &&USER || '.FOCDEF.DATA' ;
```

Edit ISHFFOCM in FOCEXEC.DATA.

Change the following lines to return the desired value.

```
-SET &DSN=&&USER || &MASTNAME || '.FOCUS' ;
```

## 8c. Pre-allocation of user program datasets

The ToolKit will attempt to allocate the program datasets to ddnames ISHXFOCX, ISHXMAST, ISHXFDEF, and ISHXFSQL respectively, unless those ddnames have been pre-allocated. Therefore, you may pre-allocate any user dataset (a PDS with LRECL 80) to these ddnames prior to executing the ToolKit. Those allocations will be respected. (You must still change the ISHFFOCM program in the previous step). It is recommended that these datasets also be concatenated to ddnames FOCEXEC, MASTER, FOCDEF, and FOCSQL.

## 8d. Dynamic allocation of user program libraries

If the files have not been pre-allocated and the user dataset is not found, the ToolKit will allocate, to the above ddnames, the first dataset in the concatenation of ddnames FOCEXEC, MASTER, FOCDEF, and FOCSQL, respectively.

## 9. Customize database allocations.

The ToolKit will attempt to dynamically access FOCUS, sequential, and comma delimited files that have been identified to the ToolKit. This is accomplished by creating a FOCEXEC that performs the necessary allocations and USE statements for a given file. This FOCEXEC is stored in the ISHFALOC dataset, with a membername that corresponds with the filename (as used by FOCUS).

For example, to allow the ToolKit to dynamically allocate the CAR FOCUS file, create a FOCEXEC in the ISHFALOC dataset with member name CAR. This FOCEXEC may appear as follows:

```
DYNAM ALLOC FILE CAR DA prefix.CAR.FOCUS SHR REUSE
-EXIT
```

Note that these members are FOCEXECs and any FOCUS commands may be included within them.

10. To provide automatic execution of the ToolKit upon the user's entry to FOCUS, edit the member SHELPROF in FOCEXEC.DATA (this step is optional).

The original FOCEXEC contains the following commands, which support automatic execution of the FOCUS Menu:

```

- * This FOCEXEC will be executed only after PROFILE FOCEXEC
- * execution has been completed.
- * INCLUDE FMMAIN

```

Change the FOCEXEC to read:

```

-IF &FOCMODE EQ 'CRJE' GOTO EXIT;
EX ISHFSHLL
-EXIT

```

This will allow the ToolKit to execute online only; it will be bypassed in batch.

NOTE: You may give the user the option to execute the ToolKit, the FOCUS Menu, or exit to native FOCUS with this FOCEXEC.

11. To execute the ToolKit, enter FOCUS with one of the following sample CLISTS. If the FOCUS/DB2 Interface is not installed, use the first example. If the FOCUS/DB2 Interface is installed, use the second example. If the SHELPROF option is not incorporated, the ToolKit may be executed from command line by issuing:

```
EX ISHFSHLL
```

NOTE: The DB2 Interface must already be installed to take advantage of ToolKit DB2 features. The DB2 installation creates the prefix.FOCSQL.DATA, prefix.TRF.DATA, and prefix.FOCSQL.LOAD datasets. The installation must also copy AUTODB2 from prefix.FOCSQL.DATA to prefix.FOCEXEC.DATA as this is used in the ToolKit.

- 11a. CLIST for non-DB2 interface installation:

```

ALLOC F(MASTER) DA('&SYSUID..MASTER.DATA' -
                  'prefix.MASTER.DATA') SHR
ALLOC F(FOCEXEC) DA('&SYSUID..FOCEXEC.DATA' -
                  'prefix.FOCEXEC.DATA') SHR
ALLOC F(FMU) DA('prefix.FMU.DATA') SHR
ALLOC F(ISHFALOC) DA('prefix.ISHFALOC.DATA') SHR
ALLOC F(ERRORS) DA('prefix.ERRORS.DATA') SHR
ALLOC F(USERLIB) DA('prefix.FUSELIB.LOAD') SHR
CALL 'prefix.FOCLIB.LOAD(FOCUS)'

```

## 11b. CLIST for DB2 installation:

```

ALLOC F(MASTER)   DA('&SYSUID..MASTER.DATA' -
                    'prefix.MASTER.DATA') SHR
ALLOC F(FOCEXEC)  DA('&SYSUID..FOCEXEC.DATA' -
                    'prefix.FOCEXEC.DATA') SHR
ALLOC F(FMU)      DA('prefix.FMU.DATA') SHR
ALLOC F(TRF)      DA('prefix.FOCSQL.DATA') SHR
ALLOC F(FOCSQL)   DA('&SYSUID..FOCSQL.DATA' -
                    'prefix.FOCSQL.DATA') SHR
ALLOC F(ISHFALOC) DA('prefix.ISHFALOC.DATA') SHR
ALLOC F(ERRORS)   DA('prefix.ERRORS.DATA') SHR
ALLOC F(USERLIB)  DA('prefix.FOCSQL.LOAD' -
                    'prefix.FOCLIB.LOAD' -
                    'prefix.FUSELIB.LOAD') SHR

CALL 'prefix.FOCLIB.LOAD(FOCUS)'

```

**NOTE:** If you have installed the DB2 interface using the TSO Attach Facility, replace the CALL statement with the following lines:

```

DSN SYSTEM (ssid)

RUN PROGRAM(FOCUS) PLAN(plan) LIBRARY('prefix.FOCLIB.LOAD')

END

```

## 3.5 Enabling National Language Support

---

FOCUS includes the ability for sites to provide FOCUS messages in a variety of languages. NLS is not available for Release 7.0 Version 1. IBI translated messages may not be on all FOCUS distribution tapes.

The steps below describe how to unload the IBI-translated messages from the FOCUS 7.0 tape:

1. Unload ERRNLS.DATA from the FOCUS installation tape.

```
//STEP1      EXEC PGM=IEBCOPY
//SYSPRINT   DD SYSOUT=*
//I44        DD DSN=ERRNLS.DATA,DISP=OLD,UNIT=TAPE,VOL=SER=volid,
//           LABEL=(44,SL,EXPDT=98000)
//O44        DD DSN=prefix.ERRNLS.DATA,DISP=(NEW,CATLG,DELETE),
//           SPACE=(CYL,(20,2,30)),UNIT=unit,VOL=SER=volid,
//           DCB=(LRECL=80,BLKSIZE=1600,RECFM=FB)
//SYSIN      DD *
            COPY INDD=I44,OUTDD=O44
/*
```

2. You may copy the members of ERRNLS.DATA into ERRORS.DATA, or concatenate this PDS under ddname ERRORS.

## 4 User Exits

---

Several user exits are available which allow you to customize FOCUS for your site. For FOCUS Release 7.0, XA-compatible versions of any user exits must be used. Otherwise, unpredictable results may occur.

### 4.1 Usage Accounting Exit: UACCT

---

FOCUS contains a user exit designed to log FOCUS usage statistics, to capture attempted violations of the DBA provisions as specified in the Master File Description, and to trap violations detected by external security systems such as RACF and ACF2. Usage statistics are provided following the completion of FOCUS commands which access data, such as TABLE, MATCH, MODIFY, etc., whether entered live or executed from a procedure, and DBA violations are logged as detected. External security violations can be trapped before any data is accessed and cause an orderly exit from FOCUS. The exit routine is also called once at the start of the FOCUS session and once at the end, to permit orderly initialization and wrap-up procedures. It should be understood that FOCUS itself merely provides the necessary information in the exit routine calling sequence, but it does not write an accounting or security log. The responsibility for managing these files and any other resource needed by this facility resides with the installation-written accounting exit.

The distributed copy of FOCUS contains a dummy version of this exit routine. To activate a live version, a small interface routine and the user code are link edited into FOCUS. The user code can be written in any language or mixture of languages, and it can issue all system macros except STAI and STAX. It is entered through a standard IBM CALL macro, at entry point UACCT. Since this code becomes an integral part of the FOCUS code, name conflicts with FOCUS routines must be avoided and it is suggested that all user-written programs start with the letters UA.

Note that XA-compatible versions of all user exits are required since the user exit code resides within the FOCUS load module.

#### 4.1.1 UACCT Calling Sequence

---

The UACCT exit is called at different times for different purposes, but the calling sequence is the same in all cases. The first argument indicates which type of call is being made. For example, in COBOL the UACCT calling sequence is:

```
ENTRY  UACCT USING CALLTYPE ATCB BILLCODE  
      COMMAND FILE LINENO PROCEDURE STATS UACVT
```

All arguments are input arguments and must not be modified by the called program, except as noted in the section dealing with external security violations.

CALLTYPE	4-byte binary integer
-1	Initialization call, made at start of job or session.
-2	Wrap-up call, made at end of job or session.
0	Usage statistics call, made following the completion of most FOCUS commands.
1	DBA violation call, made as detected.
2	FOCUS starting processing call, made after parsing but before data access.

ATCB	4-byte address
	The address of the FOCUS task control block. The TSO userid or batch job name and various other items can be obtained from this standard IBM control block.

The ATCB parameter is present in all calls. The remaining arguments are also present but they are meaningful only when the CALLTYPE is 0, 1 or 2.

BILLCODE	8 characters FOCUS billing code, set by the FOCUS command SET BILLCODE=value. This argument is included to enable accounting distinctions within a user ID.
COMMAND	8 characters FOCUS command name to which the exit call applies.
FILE	8 characters FOCUS filename.
LINENO	4-byte binary integer The current FOCEXEC procedure line number.
PROCEDURE	8 characters The name of the FOCEXEC containing the command that triggers the call to the Usage Accounting exit. Dialogue Manager commands and comment lines do not trigger calls to Usage Accounting. If the command that triggers the call is terminal input, PROCEDURE will be blank.
STATS	17 fullword binary integers Contains either usage statistics or DBA violation data. Ignore it if the call is to signal start of processing (CALLTYPE = 2)
UACVT	Not currently in use.



## 4.1.2 Usage Statistics Calls (CALLTYPE=0)

---

The UACCT exit will be called with CALLTYPE=0 for the following FOCUS commands:

TABLE\*  
 TABLEF\*  
 MODIFY\*  
 GRAPH\*  
 SCAN\*  
 MATCH  
 HOLD  
 SAVE  
 SAVB  
 ANALYSE  
 REBUILD  
 CMS/TSO  
 CREATE  
 RETYPE  
 REPLOT

The commands marked with an asterisk also generate a start of processing call to UACCT (CALLTYPE=2).

When CALLTYPE is 0 the 17 STATS words contain the information displayed by the ? STATS command. These statistics are also posted as Dialogue Manager parameters, under the &name shown. All statistics are per command, not cumulative over the FOCUS session.

- |          |   |
|----------|---|
| STATS(1) | &RETCODE<br>FOCUS return code posted by the command (this is not the return code posted when FOCUS exits). Zero indicates that the command executed successfully.                       |
| STATS(2) | &RECORDS<br>The number of records retrieved by a TABLE, TABLEF or GRAPH command. When more than one segment is referenced by the command, a record is a path in the referenced subtree. |
| STATS(3) | &LINES<br>The number of lines created by the report, or the number of records SAVED or HELD, before the application of IF TOTAL tests.  |
| STATS(4) | Reserved.   |
| STATS(5) | &BASEIO<br>The number of 4096 byte pages read or written in the FOCUS database. The value will be zero for non-FOCUS files.   |

STATS(6)      The number of reads issued when reporting from a non-FOCUS file. The value will be zero for FOCUS files. Each read is a QSAM GET, ISAM GET, or one call to the interface module, as the case may be. In the case of data interfaces, such as the IMS/DB, IDMS, TOTAL, ADABAS, and FOCSAM interfaces, each call produces one instance of one segment.

STATS(7) to STATS(16) are statistics posted by the MODIFY and the SCAN commands. These statistics will have the value zero for all other commands.

STATS(7)      &TRANS

STATS(8)      &ACCEPTS

STATS(9)      &INPUT

STATS(10)     &CHANGED

STATS(11)     &DELETED

STATS(12)     &NOMATCH

STATS(13)     &DUPLS

STATS(14)     &FORMAT

STATS(15)     &INVALID

STATS(16)     &REJECTS

STATS(17)     &SORTIO

The number of reads and writes to the FOCSORT scratch file. Zero indicates that the sort phase of the report is executed entirely in memory.

NOTE: FOCUS databases and the FOCSORT scratch file are BDAM datasets and as such must be formatted before they can be used. Formatting occurs when the CREATE command is issued or when the FOCUS database is extended in the course of a MODIFY command; FOCSORT is formatted by the first write into it. The formatting I/O activity is not included in the &BASEIO or in the &SORTIO statistics.

### 4.1.3 DBA Violation Calls (CALLTYPE=1)

---

Execution time DBA violations arising from insufficient access rights generate one of several FOCUS error messages, depending on the nature of the violation, and the accounting exit routine is called with CALLTYPE=1 to log this event. The call to log the violation occurs before the corresponding error message is displayed (i.e., before the user is made aware that the violation was detected). All the calling arguments are as documented above, except the STATS area, which in lieu of statistics contains the following items:

- STAT(1)      4 byte binary integer  
              The FOCUS error message number (e.g. 47, 56, etc.).
- STAT(2)      4-byte binary integer  
              The length, in bytes, of the variable portion of the error message.
- STAT(3)      A character string of the length indicated by STAT(2), the variable  
              portion of the error message. This will be the FOCUS filename or  
              fieldname to which the message refers.

The possible error messages are listed below. The variable portion of the message is shown in brackets.

```
(FOCO47)  USER DOES NOT HAVE SUFFICIENT ACCESS RIGHTS TO FILE:<file>
(FOCO52)  USER DOES NOT HAVE SUFFICIENT ACCESS RIGHTS TO FIELD:<fieldname>
(FOCO54)  FILE PROTECTED BY DIFFERENT DBA:<file>
(FOCO56)  USER ACCESS RIGHTS DO NOT ALLOW INCLUDE OR DELETE:<file>
```

Error messages 54 and 56 can occur only with FOCUS databases, but 47 and 52 can occur in any FOCUS readable file protected by the DBA feature.

### 4.1.4 Start of Processing Calls (CALLTYPE=2)

---

These exit calls are made following the completion of the parsing and verification of the retrieval commands (TABLE, TABLEF, GRAPH, MATCH) and of the SCAN and MODIFY commands, but before any attempt is made to access the data. For retrieval commands the start of the processing call is made for all types of data -- sequential and VSAM files, IMS databases, FOCUS databases -- and, in the case of VSAM and QSAM files, before they are opened. The CALLTYPE is 2 and all other arguments are exactly as for the Usage Statistics call which will follow, except, of course, that the STATS argument is not available.

## 4.1.5 External Security Violations

---

The UACCT exit routine can perform any type of accounting or logging activity, and it can also interrogate external security systems such as RACF or ACF2 to determine if the current FOCUS command is dealing with secured files or datasets. If a security violation is detected UACCT can terminate FOCUS gracefully by changing the input COMMAND argument to 8 bytes of binary zeroes. In this event FOCUS will issue a diagnostic message:

```
(FOC999) FOCUS TERMINATED BY USER EXIT ROUTINE UACCT
```

and return to the operating system with a return code of 8.

The COMMAND argument is the only modifiable argument in the UACCT calling sequence; all other arguments must be left as they were.

FOCUS can be terminated in this way in all UACCT calls in which the CALLTYPE is 0, 1 or 2, but the most logical time to do it is in a type 2 call, after all the parsing has been done and before any data access is attempted. In TSO, the ATCB argument points to the FOCUS task control block, which points at the userid or jobname, and the FILE argument is FOCUS name of the file against which access will be attempted. In the case of QSAM and VSAM files, it is the ddname to which the dataset was allocated, and the dsname can easily be obtained via a RDJFCB macro. In the case of FOCUS databases it is also the ddname to which the database has been allocated, except when the logical database is spread over several datasets, allocated under different ddnames. In the case of IMS or IDMS databases there is no connection between this FOCUS filename and the physical datasets which will be accessed, and the FOCUS filename corresponds to a logical view of these databases -- a PCB in the case of IMS, a subschema for IDMS.

## 4.1.6 UACCT Installation

---

The following listing shows how to link edit the UACCT exit routine into FOCUS. In this example, COBOL user code (in object form) is allocated to ddname USERCODE and is brought in with the INCLUDE USERCODE statement. Any other routines called should also have INCLUDE statements code before the INCLUDE MAINTAIN (UAIN) statement.

```

//LINK1 EXEC PGM=IEWL,PARM='LET,NCAL,LIST,MAP,SIZE=1024K'
//SYSLMOD DD DISP=OLD,DSN=prefix.FOCLIB.LOAD <-- FOCUS load library
//MAINTAIN DD DISP=SHR,DSN=prefix.FOCCTL.DATA <-- Maintenance library
//UACCT DD DISP=SHR,DSN=prefix.user.code <-- Compiled UACCT code
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSPRINT DD SYSOUT=*
//SYSLIN DD *
INCLUDE UACCT <-- UACCT and any programs it calls
INCLUDE MAINTAIN(UAINT) <-- Real exit-calling routine
REPLACE CHARGE <-- Use the new exit-calling routine
INCLUDE SYSLMOD(FOCUS) <-- FOCUS itself
INCLUDE MAINTAIN(FOCUS) <-- Linkage editor control cards
NAME FOCUS(R) <-- Output module
/*

```

The last three INCLUDE statements must execute in the order shown. The UAINT member of the FOCCTL.DATA PDS replaces the dummy interface contained in the distributed version of FOCUS.

## 4.2 Dataset Naming Enforcement (DSNCHK)

---

To manage disk space effectively, many installations enforce strict dataset naming conventions. Since FOCUS refers to files by ddname, the dataset names (dsnames) are not significant for processing. However, as an optional feature to support your installation's standards, FOCUS can call special installation-coded verification routines to determine whether the names are acceptable. This optional facility can be used with the following files: FOCUS MASTER files, FOCEXEC files, ERROR files, and FOCUS data files. If you install the optional feature, FOCUS calls your dataset name verification routine (it must be named DSNCHK) whenever any of these files is opened.

The argument in the call has the following syntax:

```
CALL SUBROUTINE DSNCHK(ddname,dsname,RETCOD)
```

where:

ddname	Is an 8-byte, left-justified filename with trailing blanks.
dsname	Is a 44-byte dataset name field, which is also left-justified with trailing blanks.

**RETCOD** Is a fullword binary integer set by DSNCHK. A return code of 0 means the name is acceptable; a non-zero code means the name is unacceptable. If the return code is not zero, FOCUS issues a diagnostic message:

```
(FOC333) INVALID DATASET NAME: dsname
```

where "dsname" is the actual datasetname. FOCUS then proceeds as if the file was not allocated, which may generate further error messages.

The dsname argument field is actually the start of a 176-byte area which is returned by the RDJFCB macro. Refer to IBM publications for further file information returned by the macro.

Subroutine DSNCHK is link-edited to FOCUS. Normally, FOCUS is delivered with a link-edited dummy DSNCHK which accepts all dataset names.

Note that XA-compatible versions of all user exits are required since the user exit code resides within the FOCUS load module.

### **DSNCHK Installation:**

---

- 1 Create member DSNCHK of FOCCTL.DATA.
2. Assemble it.
3. Link edit the new DSNCHK into FOCUS:

```
//LINK1 EXEC   PGM=IEWL,PARM='LET,NCAL,LIST,MAP,SIZE=1024K'
//SYSPRINT   DD SYSOUT=*
//SYSUT1     DD UNIT=SYSDA,SPACE=(CYL,(10,1))
//OLDMOD     DD DSN=prefix.FOCLIB.LOAD,DISP=SHR
//DSNCHK     DD DSN=assembled.dsnchk,DISP=SHR
//MAINTAIN   DD DSN=prefix.FOCCTL.DATA,DISP=SHR
//SYSLMOD   DD DSN=prefix.TEST.FOCLIB.LOAD,DISP=SHR
//SYSLIN     DD *

INCLUDE DSNCHK
INCLUDE OLDMOD(FOCUS)           <---- MODULE TO BE CHANGED
INCLUDE MAINTAIN(FOCUS)        <---- LINK1 CNTL STATEMENTS
NAME FOCUS(R)                  <---- NEW MODULE
```

## 4.3 User-Coded Password Security Interface

---

The FOCUS password (normally set by the FOCUS command SET PASS) can also be set automatically by an external security system such as RACF, ACF2 or TOP SECRET. Passwords issued this way are set when FOCUS is first entered and may be permanent (i.e., not overridable by subsequent SET USER, PASS or -PASS commands) or they may be defaults and may be subsequently overridden. They may be permanent for some users, defaults for other users, or not set at all for yet other users. The advantage of setting FOCUS passwords externally is that the password need not be known by the user and does not have to be prompted for or be embedded in a PROFILE FOCEXEC or in an encrypted FOCEXEC.

The FOCUS passwords set this way must match the FOCUS password in the Master File Descriptions of the files being accessed.

The FOCUS password is set by a CSECT in FOCUS called FOCUSID, which is a dummy in the distributed version of FOCUS. The ACF2 Interface is a working version of this routine, supplied by the ACF2 vendor. Functionally equivalent versions can be written for RACF and for other security systems. In all cases, the FOCUS password is inferred from the FOCUSID code. The function of the FOCUSID CSECT is to provide FOCUS with an overridable or non-overridable FOCUS password. The FOCUSID user exit is supported in the MVS/TSO and MVS/Batch environments.

Using this exit, a system administrator can restrict access to datasets by causing FOCUS to be entered using an inferred user ID. The installation and use of the external password security exit for ACF2 is documented in Chapter 5.

### 4.3.1 FOCUSID Calling Sequence

---

```
CALL FOCUSID(ENV,REG1,FOCID,LOGID,RC)
```

where:

ENV	Input	Fullword binary integer.
		Environment from which call is made.
	0	CMS
	1	TSO
	2	MVS batch job
	3	CICS
	4	open
	5	IMS/DC MP region
	6	IMS/DC BMP region

REG1	Input	Fullword binary integer. Register 1 as passed to FOCUS. Useful only when it points at an IMS PSB.
FOCID	Output	8 characters. 8-byte FOCUS password returned by FOCUSID. Left-justified, trailing blanks. Upper-case printable EBCDIC character only. Leading and embedded blanks respected. Trailing blanks ignored. The last non-blank character of FOCID determines whether the FOCID is overridable:  If the last non-blank character of FOCID is a period (.) or if the eighth character is not a blank then the FOCUS password is not overridable. If the last non-blank character is not a period and the password is less than eight characters then the password may be overridden using standard PASS, -PASS or SET PASS commands.  The FOCID returned, less the period if any, is set as the user's FOCUS password. The FOCUS password must match the USER= declarations stored in the appropriate Master File Descriptions.
LOGID	Output	8 characters. Not used at present.
RC	Output	Fullword binary integer. Return code posted by FOCUSID.  0 FOCID and LOGID posted by FOCUSID. 4 External Security System not installed. 8 No FOCID field (FOCUS Password) for this user. 16 Fatal error in FOCUSID (FOCUS exits via BDEXIT).

Note that the FOCUS password will not be set if RC is non-zero.

### 4.3.2 Installation

---

The object code of FOCUSID, plus any other object code which it calls, is link-edited into load module FOCUS with the JCL given below. The new FOCUSID module will replace the dummy version contained in the distributed FOCUS load module. The steps for installing your password exit are the following:

1. Write your FOCUSID code, compile and link edit it.



2. Change the FOCUSID DD statement in FOCCTL.DATA(JCLACF21) to point to your new FOCUSID module.
3. Continue editing JCLACF21 according to the steps given for the FOCUS/ACF2 Interface. After editing the JCL according to your installation's specifications, submit the job, which creates the module FOCUSID, and links the password interface modules into FOCUS.

The JCL for installing your interface, found in FOCCTL.DATA(JCLACF21) is the following:

```
//LINK1 EXEC   PGM=IEWL,PARM='LET,NCAL,LIST,MAP,SIZE=1024K'
//SYSPRINT   DD  SYSOUT=*
//SYSUT1     DD  UNIT=SYSDA,SPACE=(CYL,(10,1))
//OLDMOD     DD  DSN=prefix.FOCLIB.LOAD,DISP=SHR
//FOCUSID    DD  DSN=user.compiled.focusid,DISP=SHR
//MAINTAIN   DD  DSN=prefix.FOCCTL.DATA,DISP=SHR
//SYSLMOD    DD  DSN=prefix.TEST.FOCLIB.LOAD,DISP=SHR
//SYSLIN     DD  *
MODE AMODE(24),RMODE(24)
INCLUDE FOCUSID                <---- FOCUSID CODE
INCLUDE MAINTAIN(ACFINT1)      <---- MODE-SWITCHING CODE
ENTRY ACFINT1                  <---- CNTL STATEMENT
NAME FOCUSID(R)                <---- NEW MODULE
INCLUDE MAINTAIN(ACFINT0,ACFID) <---- ACTIVATION PROGRAMS
INCLUDE OLDMOD(FOCUS)          <---- MODULE TO BE CHANGED
INCLUDE MAINTAIN(FOCUS)        <---- CNTL STATEMENTS
NAME FOCUS(R)                  <---- NEW MODULE
```

## 5 Interface Installation

---

### 5.1 Installing the ICU Interface

---

The default ICU format files are contained in a PDS distributed on the FOCUS installation tape. These format files can be unloaded with the following JCL:

```
//COPYIT      EXEC PGM=IEBCOPY
//IN24        DD  DISP=OLD,DSN=ADMCFORM,UNIT=TAPE,
//            VOL=SER=volid,LABEL=(24,SL)
//OUT24       DD  DISP=(NEW,CATLG),DSN=prefix.ADMCFORM,
//            VOL=SER=volume,UNIT=SYSDA,SPACE=(TRK,(2,1,2)),
//            DCB=(LRECL=400,RECFM=F,BLKSIZE=400)
//SYSPRINT    DD  SYSOUT=*
//SYSIN       DD  *
              COPY INDD=IN24,OUTDD=OUT24
/*
```

NOTE: If you are installing from a 3480 cartridge, you may have to change TAPE to your site-defined name for 3480 drives in DD statement IN24.

Reminder: The basic FOCUS system installation and testing must be completed before the ICU Interface can be used.

### Considerations for Running the ICU Interface

1. FOCUS must be installed with the "High Resolution Graphics for 3270 Terminals" option.
2. The terminal must have extended attributes and PS storage or Vector Graphics in order to be able to run GDDM graphics. (See the IBM GDDM manual for model numbers).
3. Enough main memory must be available to execute ICU with FOCUS. ICU usually requires a minimum virtual storage of 4Mb.
4. The following is a typical CLIST to use FOCUS and the ICU Interface:

```
ALLOC F(ERRORS)      DA('prefix.ERRORS.DATA')                SHR
ALLOC F(USERLIB)     DA('prefix.FUSELIB.LOAD' 'SYS1.GDDM.V2R1.GDDMLOAD') SHR
ALLOC F(ADMSYMBL)    DA('SYS1.GDDM.V2R1.GDDMSYM')            SHR
ALLOC F(ADMCFORM)    DA('prefix.ADMCFORM')                   SHR
ALLOC F(ADMCDATA)    DA(ADMCDATA)                             SHR
```

```
ALLOC F(ADMDECK) SPACE(2 2) TRACKS  
CALL 'prefix.FOCLIB.LOAD(FOCUS)'
```

- The GDDM load library should usually be allocated to ddname STEPLIB in the TSO logon proc since STEPLIB is where GDDM expects to find the other modules that it loads. Some sites may be able to allocate the GDDM load library to ddname USERLIB instead.
- The GDDM symbols library must be allocated to ddname ADMSYMBL.
- The ADMCFORM format PDS must be allocated to ddname ADMCFORM.
- The ddname ADMCDATA must be allocated to either a scratch dataset if no permanent storage of graph data is required or to a permanent dataset. The DCB for this dataset should be:

```
LRECL=400,RECFM=F,BLKSIZE=400
```

This dataset is a PDS and therefore must be given directory blocks.

- The ddname ADMDECK must be allocated to a scratch disk. There is no need to supply DCB information for this dataset.

## 5.2 Installing and Using an External Password Security System

---

There are two options concerning the use of an external password security system with MVS FOCUS:

1. Use the FOCUS-supplied ACF2 Interface.
2. Write and use your own security interface. (See Chapter 4)

### 5.2.1 The FOCUS-Supplied ACF2 Interface

---

The FOCUS/ACF2 Interface is an optional FOCUS facility which integrates FOCUS security features with the ACF2 file security system. The Interface provides the following benefits to FOCUS users and security administrators:

- The FOCUS userid set by the ACF2 Interface is not echoed to the terminal, and it is not shown in any batch job output listings. This ensures greater security than explicit SET USER and SET PASS commands, which echo output to the screen and show it in batch output listings.
- FOCUS users need not know their FOCUS userids, increasing security.
- Many logon IDs can be translated into the same FOCUS userid. This enables the creation of classes of FOCUS users, all of whom have different logon IDs but the same level of access to FOCUS files.
- There is no need to embed FOCUS userid declarations in catalogued FOCUS procedures. When employees leave or FOCUS userids are changed, security can be maintained without changing production procedures. In addition, the database administrator is able to assume all responsibility for FOCUS userid maintenance.

Each logon ID exists on an ACF2 LOGONID record and is paired with a FOCUS userid which provides access to FOCUS databases. Upon entering FOCUS, the ACF2 Interface infers the FOCUS userid from the logon ID supplied by the user. This FOCUS userid is maintained for the duration of the session, and may or may not be overridden by the user. The result of the inference is equivalent to the FOCUS command

```
SET USER = FOCUS userid
```

### 5.2.2 Usage

---

Once installed, normal FOCUS users will see no difference between the default version of FOCUS and the ACF2 version until attempt to override the FOCUS userids. The database administrator can use the ACF2 CHANGE command to assign FOCUS userids to logon IDs as a default FOCUS userid or as an imposed FOCUS userid.

## Default FOCUS Userids

---

When the LOGONID record contains a default FOCUS userid, a FOCUS userid is inferred for the user from the LOGONID record, and the user can override this FOCUS userid. A default FOCUS userid can be up to eight bytes long.

To assign a default FOCUS userid for a logon ID, use the CHANGE command with a FOCUS userid (up to eight bytes long) which does not end in a period. For example, if the logon user SMITH wishes to change his accessibility to that of TOM, the following ACF2 command would be issued:

```
CHANGE SMITH FOCUSID(TOM)
```

This is functionally equivalent to the FOCUS command

```
SET USER = TOM
```

or the command

```
SET PASS = TOM
```

issued from the logon userid SMITH. SMITH could override this latest access by issuing another SET USER or SET PASS command, or a Dialogue Manager -PASS command.

## Imposed FOCUS Userids

---

When the ACF2 LOGONID record contains an imposed FOCUS userid, the user cannot override it. An imposed FOCUS userid can be up to seven bytes long plus a period, for a maximum length of eight bytes.

To assign an imposed FOCUS userid to a logon ID, use the ACF2 CHANGE command and include a period (.) at the end of the FOCUS userid. For example,

```
CHANGE SMITH FOCUSID(TOM.)
```

assigns an imposed FOCUS userid, TOM, to the logon ID, SMITH. In this case, the user operates FOCUS with an imposed level of access to FOCUS files that cannot be changed by the user. FOCUS behaves as if user SMITH had entered

```
SET USER = TOM
```

as the first FOCUS command. However, the period at the end of the FOCUS userid in the LOGONID record prevents the user from overriding this level of access. Any attempt to override the imposed ID results in a FOCUS diagnostic message.

## Unidentified FOCUS Users

---

It is not necessary to identify all possible FOCUS users in the ACF2 LOGONID records. Users whose LOGONID records provide no FOCUS userid operate FOCUS as if the FOCUS/ACF2 Interface were not installed. If they want to access files protected by FOCUS security features, they must identify themselves through an explicit SET USER or SET PASS command or the Dialogue Manager

-PASS command.

## 5.2.3 Installation

---

To install the FOCUS/ACF2 Interface, follow the steps below:

1. Edit dataset member JCLACF21 (shown below), contained in the FOCCTL.DATA library, according to the steps given below. After editing the JCL according to your installation's specifications, submit the job, which creates the module FOCUSID, and links the live ACF2 modules into FOCUS.

```
//LINK1 EXEC   PGM=IEWL,PARM='LET,NCAL,LIST,MAP,SIZE=1024K'
//SYSPPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(10,1))
//OLDMOD DD DSN=prefix.FOCLIB.LOAD,DISP=SHR
//FOCUSID DD DSN=prefix.FOCCTL.DATA(FOCACF21),DISP=SHR
//MAINTAIN DD DSN=prefix.FOCCTL.DATA,DISP=SHR
//*ACFMOD DD DSN=***.***,DISP=SHR
//SYSLMOD DD DSN=prefix.TEST.FOCLIB.LOAD,DISP=SHR
//SYSLIN DD *

MODE AMODE(24),RMODE(24)

INCLUDE FOCUSID <---- FOCUSID CODE
INCLUDE MAINTAIN(ACFINT1) <---- MODE-SWITCHING CODE
ENTRY ACFINT1 <---- CNTL STATEMENT
NAME FOCUSID(R) <---- NEW MODULE

INCLUDE MAINTAIN(ACFINT0,ACFID) <---- ACTIVATION PROGRAMS
INCLUDE OLDMOD(FOCUS) <---- MODULE TO BE CHANGED
INCLUDE MAINTAIN(FOCUS) <---- CNTL STATEMENTS
NAME FOCUS(R) <---- NEW MODULE
```

- A. If the ACF2 ACCVT is not anchored in the subsystem vector control table (SSVT), activate the ACFMOD DD statement. Also include the following under the SYSLIN DD statement immediately preceding the INCLUDE FOCUSID card:

```
INCLUDE ACFMOD($ACFGCVT)
```

- B. Note that the MAINTAIN DD statement should point to the maintenance dataset for the load library specified on the OLDMOD DD statement. The new module is created in the dataset pointed to by SYSLMOD, from the like-named module contained in the dataset pointed to by OLDMOD. Note that if the two statements point to the same dataset, the new module will replace the old. Otherwise, after testing the new module, you can copy it to the old.
2. Add a field to member USERLID of SYS1.ACFMAC:

```
LIDFOCID DS CL8 FOCUSID
```

3. Add the following @CFDE macro to member USERCFDE of SYS1.ACFMAC. Note that the ALTER and LIST attributes may be different at your installation.

```
@CFDE FOCUSID,LIDFOCID,CHAR, X
ALTER=ACCOUNT,LIST=ALL,FLAGS=NULL,PRTN=9, X
RRTN=1,GROUP=4
```

4. Reassemble and link the ACFFDR into LPALIB.
5. IPL with CLPA.

## 5.3 Installing the FOCCALC Graphics Facility

---

FOCCALC includes a Graphics facility that enables the display of spreadsheet data as graphs.

The Graphics facility requires a 3270 type terminal fitted for graphics with access to the IBM Graphics Data Display Manager (GDDM) software product.

The FOCCALC Graphics facility is installed separately from GDDM graphics support for FOCUS high-resolution graphics. A minimum of 2500K bytes of virtual memory is required to run FOCCALC with the Graphics facility.

### 5.3.1 Installation

---

On MVS/TSO, FOCCALC Release 2 is installed automatically when FOCUS is installed. To include the FOCCALC Graphics facility, you will need to link edit a GDDM stub from the GDDM load library into FOCCALC. This step can be done any time after the installation of the basic FOCUS system.

1. There is one dataset on the FOCUS installation tape that must be unloaded for FOCCALC Release 2, whether or not the Graphics facility is installed. This dataset is named MODEL.DATA and may be loaded from tape to disk with the following JCL. This partitioned dataset should be available to all FOCCALC users.

```
//STEP1      EXEC PGM=IEBCOPY
//SYSPRINT   DD SYSOUT=*
//I23        DD DSN=MODEL.DATA,DISP=OLD,UNIT=TAPE,VOL=SER=volid,
//            LABEL=(23,SL,EXPDT=98000)
//O23        DD DSN=prefix.MODEL.DATA,DISP=(NEW,CATLG,DELETE),
//            SPACE=(TRK,(1,1,2)),UNIT=unit,VOL=SER=volid,
//            DCB=(LRECL=1028,BLKSIZE=4112,RECFM=VB)
//SYSIN      DD *
            COPY INDD=I23,OUTDD=O23
/*
```

2. To install FOCCALC Release 2 with the Graphics facility, you must relink module FCALC2 with GDDM Graphics. Sample JCL can be found in dataset FOCCTL.DATA(LINKCA2G). This JCL is shown below.

```
//*****
/* Relink module FCALC2 with GGDM Graphics.
/* Change DSN names as appropriate at your location.
//*****
//LFCALC2 EXEC PGM=IEWL,PARM='NCAL,LET,LIST,SIZE=1024K'
//OLDMOD DD DSN=prod.FOCLIB.LOAD,DISP=SHR
//FOCCTL DD DSN=prod.FOCCTL.DATA,DISP=SHR
//SYSLMOD DD DSN=new.FOCLIB.LOAD,DISP=SHR
//ADMRLIB DD DSN=SYS1.GDDMLOAD,DISP=SHR
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(10,1))
//SYSPRINT DD SYSOUT=*
//SYSLIN DD *
INCLUDE FOCCTL(GDDMDRV2) /
INCLUDE FOCCTL(@GDDMDR2) <----- FOCCALC/GDDM INTERFACE
INCLUDE FOCCTL(GDDMTAB2) \
INCLUDE FOCCTL(FCALC2) <----- Linkedit Control Statements
INCLUDE ADMRLIB(FSINR) <----- GDDM Stub
```



```
REPLACE GDDMDRVR          <----- Replace old entries
REPLACE NO_CLINK
INCLUDE OLDMOD(FCALC2)    <----- Existing copy of FocCalc
NAME FCALC2(R)
/*
```

## 6 MAINTENANCE

---

This chapter deals with a number of topics concerning the maintenance of FOCUS after installation.

### 6.1 FOCUS Release Numbering

---

FOCUS release numbers are in the form:

V.R

V.R represents the version and release of FOCUS.

### 6.2 The Release Signature

---

FOCUS runs in a number of operating environments; its configuration differs among them. These differences are what we mean when we speak of different versions: the MVS/TSO version or the VM/CMS version. Each of these versions can be created at each release level, sometimes with options selected at installation time. The release signature identifies the most significant of all these possible combinations.

The release banner is always displayed when FOCUS is first entered. In the MVS/TSO version it consists of the release number followed by the current date and time and your sitecode. For example:

```
FOCUS 7.0.8      mm/dd/yyyy  hh.mm.ss  sitecode
```

The release signature can be displayed on demand anytime you are within FOCUS by using the ? RELEASE command. ? RELEASE shows the release number, creation date, and sitecode of the currently running copy of FOCUS.

```
FOCUS 7.0.8      CREATED mm/dd/yyyy  sitecode
```

### 6.3 Test and Production Releases

---

A major new release such as Release 7.0 contains a large number of new features and goes through an extensive test cycle in which many FOCUS users participate at a large number of beta test sites. During this cycle, quite a few test releases are generated and distributed to the test sites. The numbering of these releases, and of FOCUS production tapes, is described below.

## 6.4 Corrections to Reported Problems

---

If you encounter a problem that requires a correction to FOCUS rather than to your own procedures, the correction will become part of the next release. In some situations, you will receive the correction to your problem as soon as it is available, regardless of when the next release is scheduled.

We call these corrections individual PTFs or reach-ahead service, and they are identified by a PTF number and an Update Level.

You will receive an individual PTF only in response to a specific problem that you yourself have reported. (The solution to problems reported by other users will reach your installation by way of the next release.) The accompanying documentation will tell you how to apply it, identify the problem being corrected, and, most importantly, state the range of releases into which the PTF may be incorporated. It is critical that you do not apply a PTF outside the indicated release range. If you do, you will probably damage the copy of FOCUS that you are trying to correct.

If you apply more than one PTF to a release, they must be applied in Update Level order. For this reason, please keep a record of all PTFs applied to each of your copies of FOCUS.

## 6.5 The Release Memos

---

FOCUS distribution tapes contain online information which summarize the new features and error corrections install notes, operational notes, and known problems included in the release.

This information may be viewed online by issuing the command:

```
>>EX READMEF
```

This is a cumulative listing of features added to each FOCUS release.

The summary of error corrections for the release are also included in the problems corrected selection. These are cumulative for each release. The ERRORS and MASTER PDSs must be allocated in order to run READMEF.

## 6.6 Release Tapes

---

All FOCUS tapes that you receive from IBI are one of two types:

- A full release distribution tape.

- An individual PTF tape.

These tapes are standard label 9-track tapes, or 3480 cartridges. Full FOCUS release distribution tapes are 6250 BPI or 1600 BPI; individual PTF tapes are 1600 BPI. You unload them by following the examples in Chapter 2.

NOTE: If you are using any of the optional interfaces listed in Chapter 1, you must reinstall these interfaces after installing a new PUT Level of FOCUS.

## 6.7 Load Libraries and Maintenance Libraries

---

All separately installed FOCUS components are delivered in paired sets of libraries: a load library, and possibly other libraries needed for execution, and a maintenance library. The maintenance library is not needed for execution, but it contains whatever is needed to install the component and maintain its load library. The contents of all these libraries may vary from release to release and it is critical that you retain the association between a load library of a given release and its corresponding maintenance library. The simplest way to do that is to follow the dataset naming conventions described in Chapter 1 of this manual.

The following table shows the standard names of all the FOCUS libraries, including the interface libraries that have their own associated installation guides. You will find that FOCUS is very easy to maintain if you retain these dataset names, with only the addition of high-level prefixes to differentiate between different releases.

<b>Component</b>	<b>Load Library Other Libraries</b>	<b>Maintenance Library</b>
The basic FOCUS system plus optional components whose installation is covered in this manual	FOCLIB.LOAD FUSELIB.LOAD ERRORS.DATA MASTER.DATA FOCEXEC.DATA ADMCFORM WINFORMS.DATA	FOCCTL.DATA
FOCCALC	MODEL.DATA	
The FOCUS Multi-Session Option	MSO.LOAD	MSO.DATA
The ADABAS Interface	ADABAS.LOAD	ADABAS.DATA
The DATACOM/DB Interface	DATACOM.LOAD	DATACOM.DATA
The DB2 Interface	FOCSQL.LOAD	FOCSQL.DATA
The IDMS Interface	IDMS.LOAD	IDMS.DATA
The IMS/DB and IMS/DC Interfaces	IMS.LOAD	IMS.DATA

The MODEL 204 Interface	FOCM204.LOAD	FOCM204.DATA
The SAP Interface	FOCSAP.LOAD	
The SYSTEM 2000 Interface	FOCS2K.LOAD	FOCS2K.DATA
The TERADATA Interface	FOCDBC.LOAD	FOCDBC.DATA
The TOTAL Interface	TOTAL.LOAD	TOTAL.DATA

For historical reasons, the maintenance library of FOCLIB.LOAD is called FOCCTL.DATA rather than FOCLIB.DATA. For all other separately installed components, the load library has the suffix LOAD and its maintenance library has the same name with the low-order suffix DATA.

## 6.8 How to Apply an Individual PTF

---

An individual PTF changes the contents of one of the standard FOCUS libraries listed above. Most of the time this is done by link editing new object code into a load module, but the correction may also affect the contents of the maintenance library itself, or of the error message library. In any event, you will receive written documentation that describes the problem being corrected, the range of FOCUS releases to which the PTF may be applied, and instructions on how to apply the PTF. The PTF will always contain all the necessary JCL needed for its application, expressed in terms of the standard FOCUS library names.

Follow the installation instruction which accompany the tape. The instructions will show you how to unload it into a partitioned dataset named PTFLIB.DATA. The JCL member will consist of one or more job steps, each of which will be the appropriate utility to update one of the standard FOCUS libraries, using the other PTF members as input or as control statements. All you will have to do is to change the standard dataset names in the JCL member to your actual dataset names, add a JOB card, and submit it for batch execution.

Each PTF also shows an Update Level number. If you have several PTFs to apply, process each one individually in ascending Update Level sequence, not in the PTF number sequence. The PTF number is merely a substitute for a name and has no intrinsic significance.

The application of individual PTFs will not be reflected in the FOCUS release signature.

## 6.9 The FOCMAP Interface

---

The FOCMAP Interface is a data interface module that makes it possible to analyze MVS and CMS load libraries through simple FOCUS queries. It maps the various elements of a load module into a hierarchical structure, described to FOCUS as file FOCMAP, and gives access to such items as subroutine names and sizes, compilation and linkage edit dates, external references, and so forth. Although it was developed to assist the IBI programming staff in the development and maintenance of FOCUS, the FOCMAP Interface can be used to analyze any load module generated by the OS linkage editor.

This interface is distributed as member FCMAPIN of FOCLIB.LOAD, and the associated MASTER file as member FOCMAP of FOCCTL.DATA. Members FOCMAP1, FOCMAP2, and FOCMAP3, also in FOCCTL.DATA, are sample procedures that use this utility. If you are interested in using it on your own load libraries, contact your local IBI representative and ask for the *FOCMAP Users Manual*.

The following example illustrates the use of the FOCMAP interface. It shows the FOCMAP1 procedure and the output it generates.

```
TABLE FILE FOCMAP
HEADING CENTER
" DSN"
" "
" LIST BY MEMBERS ... RUN ON &DATE AT &TOD"
" "
PRINT MSIZE AS 'SIZE,HEX'
      ISIZE AS 'SIZE,DEC'
      EPNAME AS 'ENTRY'
      LDATE AS 'CREATED , ON'
      LJDATE AS 'YYDDD'
      ALIAS OVLY RENT
BY MEMBER
END
```

WIBFOC.S7009609.FOCLIB.LOAD  
 LIST BY MEMBERS .. RUN ON 10/22/97 AT 14.24.24

MEMBER	SIZE HEX	SIZE DEC	ENTRY	CREATED ON	YD	DD	ALIAS	OVLY	RENT
-----	----	----	-----	-----	----	----	-----	----	----
ACESNK	01E040	122,944	HLIMAIN	97 OCT 21	97294		Y		
ADASIM	005308	21,256	ADABAS	97 OCT 21	97294				
ADSRV	006B60	27,488	ADSRV	97 OCT 21	97294				
ALCUTIL	003CB8	15,544	XPXPXPXP	97 OCT 09	97282				Y
AXSC	003598	13,720	AXSC	97 OCT 21	97294				
AXSP	004D88	19,848	AXSP	97 OCT 21	97294				
BANUSER	000040	64	BANUSER	97 JUL 16	97197				Y
CALLDB2	0058A0	22,688	CDRIVER	97 OCT 09	97282				
CALLIMS	0059E0	23,008	APPCIMS@	97 OCT 09	97282				Y
CALLIMS2	006E98	28,312	CALLIMS2	97 OCT 09	97282				Y
CALLPGM	0043E0	17,376	CALLPGM@	97 OCT 09	97282				Y
CALLRTR	0044A0	17,568	CALLRTR@	97 OCT 09	97282				Y
CALLUTL	001E40	7,744	CALLUTL	97 OCT 09	97282				Y
CTAB	018FA0	102,304	XPXPXPXP	97 OCT 21	97294				Y
CILINK	000650	1,616	CILINK	97 JUL 16	97197				
CIREAD	002440	9,280	CIREAD	97 OCT 09	97282				Y
CIWRT	002800	10,240	CIWRT	97 OCT 09	97282				
CKIDR	001098	4,248	CKIDR	97 JUL 16	97197				
CLEAR	000098	152	CLR3270	97 JUL 16	97197				
CLR3270	000098	152	CLR3270	97 JUL 16	97197		Y		
CLR3270T	000098	152	CLR3270	97 JUL 16	97197		Y		
CLR3270V	000098	152	CLR3270	97 JUL 16	97197		Y		
CMSDLL	000040	64	XPXPXPXP	97 JUL 16	97197				Y
CONATLAS	003338	13,112	XPXPXPXP	97 OCT 09	97282				
CONCLTTS	003AC8	15,048	CONCLTTS	97 OCT 09	97282				
CXCRYPT	00A1E8	41,448	XPXPXPXP	97 OCT 09	97282				
DQSOLF	02C930	182,576	DQSOLF@	97 OCT 21	97294				Y
DTSRV	0062A8	25,256	DTSRV@	97 OCT 09	97282				Y
DUMPCNV	00FC40	64,576	DUMPCNV	97 OCT 09	97282				
DUMPIDR	001C20	7,200	DUMPIDR	97 JUL 16	97197				
DUMPRD	0118B8	71,864	READDUMP	97 OCT 09	97282				
EDAAPI	036110	221,456	XPXPXPXP	97 OCT 09	97282				
EDAAPX	03A770	239,472	XPXPXPXP	97 AUG 01	97213				Y
EDAITST	008E38	36,408	EDAITST	97 OCT 09	97282				
EDAPRT	00CAE8	51,944	CEDAPRT	97 OCT 09	97282				
EDASAF	1807C8	1,574,856	FOCUS	97 OCT 22	97295		Y		
EDATEST	00A300	41,728	CLKTEST	97 OCT 09	97282		Y		
EDAUSRO	0064B0	25,776	RDAPASO	97 OCT 09	97282				Y
EDXFOC	0152A8	86,696	EDXFOC	97 OCT 21	97294				Y
EMGSRV	006338	25,400	XPXPXPXP	97 OCT 09	97282				Y
F\$ROOT	30F300	3,207,936	F\$ROOT	97 OCT 22	97295				Y
FCALC2	0CBC98	834,712	#FC2EP	97 OCT 09	97282				
FCMAPIN	01E828	124,968	FOCMAPIN	97 OCT 21	97294				
FMI	009D70	40,304	FMI@	97 OCT 09	97282				Y
FOC	00DE90	56,976	FOCC@	97 OCT 21	97294				
FOCDIR	001EC0	7,872	FOCDIR	97 JUL 16	97197				
FOCINI	000070	112	XPXPXPXP	97 JUL 16	97197				
FOCMIL	007A88	31,368	FOCMIL	97 JUL 16	97197				Y

WIBFOC.P7009609.FOCLIB.LOAD  
 LIST BY MEMBERS .. RUN ON 10/22/97 AT 14.24.24

MEMBER	SIZE HEX	SIZE DEC	ENTRY	CREATED ON	YYDD	ALIAS	OVLY	RENT
-----	----	----	-----	-----	----	-----	----	----
FOCMILIV	002290	8,848	FOCMILIV	97 JUL 16	97197			
FOCMILP	000E98	3,736	FOCMILP	97 JUL 16	97197			Y
FOCML3	007A88	31,368	FOCML3	97 JUL 16	97197	Y		Y
FOCSAM	0176B8	95,928	FOCSAM	97 OCT 21	97294			
FOCTRN	00E228	57,896	XPXPXPXP	97 OCT 09	97282			
FOCUS	1807C8	1,574,856	FOCUS	97 OCT 22	97295			
FOCUTL	0121E8	74,216	XPXPXPXP	97 OCT 21	97294			Y
FOCWIN	0158D0	88,272	FWMAIN	97 OCT 09	97282			Y
FOCXDC	000008	8	FOCXDC	97 JUL 16	97197			
FSASLIB	00EBA0	60,320	FSASLIB	97 OCT 09	97282			Y
FSIO	002CE8	11,496	XPXPXPXP	97 OCT 09	97282			
FSROOT	30F300	3,207,936	F\$ROOT	97 OCT 22	97295	Y		Y
FSTEST	000008	8	FSTEST	97 JUL 16	97197			
FSXAIO	005980	22,912	XAIO	97 OCT 09	97282			
FTPC	00E738	59,192	XPXPXPXP	97 OCT 09	97282			Y
FXCMEM	007DD0	32,208	XPXPXPXP	97 OCT 09	97282			Y
GENFBLCK	000460	1,120	GENFBLCK	97 JUL 16	97197			
GENFCTL	000DC8	3,528	GENFCTL	97 JUL 16	97197			
GETLOGID	0003C0	960	GETLOGID	97 JUL 16	97197	Y		
GETPRFID	0003C0	960	GETPRFID	97 JUL 16	97197	Y		
GETSECID	0003C0	960	GETSECID	97 JUL 16	97197			
GKCOMP	026690	157,328	GKBASE	97 OCT 09	97282			
GKID3	0127E0	75,744	GKBASE	97 OCT 09	97282			
GKINF	013558	79,192	GKINF	97 OCT 09	97282			
GKMAV	002E48	11,848	GKMAV@	97 OCT 09	97282			Y
GKSMODE	0104C8	66,760	GKSMODE@	97 OCT 21	97294			
GKUNBIT	000070	112	GKUNBIT	97 JUL 16	97197			Y
GKUTIL	005958	22,872	GKUTIL@	97 OCT 09	97282			Y
GNTINT	02CF70	184,176	XPXPXPXP	97 OCT 21	97294			Y
GRMDBA	0082C8	33,480	XPXPXPXP	97 OCT 09	97282			Y
GRMDEF	004F08	20,232	XPXPXPXP	97 OCT 09	97282			Y
GRMMAS	00AD70	44,400	XPXPXPXP	97 OCT 09	97282			Y
GRMMNT	00F6A8	63,144	XPXPXPXP	97 OCT 09	97282			Y
HERMES	005520	21,792	XPXPXPXP	97 OCT 09	97282			Y
HFFUNC	00D0C0	53,440	XPXPXPXP	97 OCT 09	97282			Y
HFINST	000130	304	HFINST0	97 JUL 16	97197			Y
HLFUNC	0075E8	30,184	XPXPXPXP	97 OCT 09	97282			Y
HLI	0E16D8	923,352	\$HLIB	97 OCT 22	97295			
HLIFOCUS	001E80	7,808	FOCUS	97 OCT 09	97282			
HLIKX	005AB0	23,216	HLIKX	97 OCT 21	97294			
HLISECUR	000088	136	HLISECUR	97 JUL 16	97197			
HLISNK	01E040	122,944	HLIMAIN	97 OCT 21	97294			
HLISST	0E2FD0	929,744	\$HLIB	97 OCT 22	97295			
HLISU	00F7D0	63,440	HLISEN	97 OCT 21	97294			
HLIX	009698	38,552	HLIX@	97 OCT 09	97282			
HRMITCP	009B38	39,736	XPXPXPXP	97 OCT 09	97282			
HRMTCP	008260	33,376	XPXPXPXP	97 OCT 09	97282			
HRMXTSK	009B58	39,768	XPXPXPXP	97 OCT 09	97282			Y



WIBFOC.P7009609.FOCLIB.LOAD  
 LIST BY MEMBERS .. RUN ON 10/22/97 AT 14.24.24

MEMBER	SIZE HEX	SIZE DEC	ENTRY	CREATED ON	YYDD	ALIAS	OVLY	RENT
-----	----	----	-----	-----	----	-----	----	----
HRM0	00FF68	65,384	XPXPXPXP	97 OCT 09	97282			
HRM2	00D6E0	55,008	XPXPXPXP	97 OCT 09	97282			
HRM62	00E318	58,136	XPXPXPXP	97 OCT 09	97282			Y
HRPGATE	003F78	16,248	CHRPGATE	97 OCT 09	97282			
HTIX	008B30	35,632	HTIX@	97 OCT 09	97282			Y
H2O	00A270	41,584	HOROADR@	97 OCT 09	97282			Y
IBICOMMN	001620	5,664	IBICOMMN	97 JUL 16	97197			
IBICPUID	000EB8	3,768	IBICPUID	97 SEP 02	97245			Y
IBILINK	003538	13,624	IBILINK	97 JUL 16	97197			
IBITRC	005718	22,296	XPXPXPXP	97 OCT 22	97295			
IMAN	01BC70	113,776	IMAN@	97 OCT 21	97294			Y
IMSRV	0086E0	34,528	IMSRV@	97 OCT 09	97282			Y
INTCBS	0076A8	30,376	XPXPXPXP	97 OCT 09	97282			Y
INTERP	0143C8	82,888	INTERP	97 OCT 09	97282			
INTERPAB	000150	336	INTERPAB	97 JUL 16	97197			
KKDO	011B20	72,480	XPXPXPXP	97 OCT 09	97282			
LETH	019540	103,744	LETCTX	97 OCT 22	97295			
LKTEST	00A300	41,728	CLKTEST	97 OCT 09	97282			
MANMSO	07D880	514,176	\$MANMSO	97 JUL 16	97197			
MASPAR	020FD0	135,120	XPXPXPXP	97 OCT 22	97295			Y
MFP	01CBD8	117,720	XPXPXPXP	97 OCT 09	97282			Y
MNTCLI	018D00	101,632	XPXPXPXP	97 OCT 09	97282			Y
MNTDRV	002F00	12,032	MNTDRV@	97 OCT 09	97282			Y
MNTIPR	096AB0	617,136	XPXPXPXP	97 OCT 21	97294			Y
MNTRT1	007160	29,024	MNTRT1@	97 OCT 09	97282			Y
MODMNT	00A6E8	42,728	MDMAIN	97 OCT 09	97282			
MSCTL	03A2C8	238,280	SSCTL	97 OCT 21	97294	Y		
MSDEB	005778	22,392	DEBIN	97 OCT 09	97282			Y
MSDUMP	0060F0	24,816	SSVXME	97 OCT 09	97282			
MSETUP	006308	25,352	MSETUP	97 OCT 09	97282			
MSINIT	002038	8,248	SSVXME	97 OCT 09	97282			
MSKILL	00B540	46,400	MSKILL0	97 OCT 21	97294			
MSODEB	000158	344	MSODEB	90 FEB 02	90033			
MSODLL	008490	33,936	XPXPXPXP	97 OCT 09	97282			Y
MSOINFO	000540	1,344	MSOINFO	97 JUL 16	97197			
MSOMAPS	000008	8	MSOMAPS	97 JUL 16	97197			
MSOTRAC	0001E0	480	MSOTRAC	97 JUL 16	97197			
MSSMFDUM	000008	8	MSSMFDUM	97 JUL 16	97197			Y
MSSMFXT	000150	336	MSSMFXT	97 JUL 16	97197			Y
MSUSER	012B28	76,584	SSVXME	97 OCT 21	97294			
MSXGET	005018	20,504	SSVXME	97 OCT 09	97282			
MSXPUT	0063E0	25,568	SSVXME	97 OCT 09	97282			
MVSDLL	007CE8	31,976	XPXPXPXP	97 OCT 09	97282			Y
NLSP	011288	70,280	XPXPXPXP	97 OCT 09	97282			Y
NLSSYS	007B28	31,528	XPXPXPXP	97 OCT 21	97294			
NMDFAKE	000060	96	NMDFAKE	97 AUG 01	97213			Y
NMDIN	01B8B0	112,816	NMDIN@	97 OCT 21	97294			Y
NMDLINK	000060	96	NMDLINK	97 AUG 01	97213	Y		Y

WIBFOC.P7009609.FOCLIB.LOAD  
 LIST BY MEMBERS .. RUN ON 10/22/97 AT 14.24.24

MEMBER	SIZE HEX	SIZE DEC	ENTRY	CREATED ON	YD	ALIAS	OVLY	RENT
-----	----	----	-----	-----	----	-----	----	----
NMDPAS2	004AF8	19,192	NPI2CALL	97 OCT 09	97282			
NTDM	0062C8	25,288	XPXPXPXP	97 OCT 09	97282			Y
NTDUMP	00B2B8	45,752	XPXPXPXP	97 OCT 09	97282			Y
NTHR	012760	75,616	XPXPXPXP	97 OCT 21	97294			Y
NWH	012BB8	76,728	NWCTX	97 OCT 21	97294			Y
ODIN	00FF90	65,424	XPXPXPXP	97 OCT 09	97282			Y
ODINIO	004760	18,272	XPXPXPXP	97 OCT 09	97282			Y
OLDHRM	021EC0	138,944	XPXPXPXP	97 OCT 09	97282			
OMHERMES	004F28	20,264	EDHRM@	97 JUL 16	97197			Y
OMITCP	00AB50	43,856	OMITCP@	97 OCT 09	97282			Y
OMLU62	00D6D8	55,000	OMLU62@	97 OCT 09	97282			Y
OMMEM	003CF0	15,600	XPXPXPXP	97 OCT 09	97282			Y
OMSOCK	00B950	47,440	SOCKSTUB	97 OCT 09	97282			
OMTCP	00B950	47,440	SOCKSTUB	97 OCT 09	97282	Y		
OPSINIT	0092F0	37,616	STARTUP	97 OCT 09	97282			
OSL	013DC0	81,344	XPXPXPXP	97 OCT 09	97282			Y
OTB	009258	37,464	STARTUP	97 OCT 09	97282			
OXAT	000430	1,072	OXAT#	97 JUL 16	97197			
OXSS	00EEC8	61,128	OXSS	97 OCT 21	97294			Y
PARSER	014350	82,768	XPXPXPXP	97 OCT 09	97282			Y
PTFTABLE	013910	80,144	PTFTABLE	97 JUL 16	97197			
PTFTUPD	002470	9,328	PTFTUPD	97 OCT 09	97282			
PTFVR	000580	1,408	PTFVR	97 JUL 16	97197			
PTREE	007610	30,224	XPXPXPXP	97 OCT 09	97282			Y
QADBSIN	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QADBSINX	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QCVTIN	0074E8	29,928	XPXPXPXP	97 OCT 21	97294			Y
QDATACOM	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QDSET	00EFE0	61,408	QDSET@	97 OCT 21	97294			Y
QFMI	005918	22,808	QFMI@	97 OCT 09	97282			Y
QFOCQMF	004CA0	19,616	QFOCQMF@	97 OCT 09	97282			
QFOCSAM	0222D8	139,992	QVSAMX@	97 OCT 21	97294	Y		Y
QFQDBE	029588	169,352	XPXPXPXP	97 OCT 09	97282			Y
QFSP000	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QHLLX	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QIDMSR	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QIMAN	00C228	49,704	QIMAN@	97 OCT 09	97282			Y
QIMS	010DB0	69,040	QIMS@	97 OCT 21	97294			Y
QIMSX	032580	206,208	QIMSX@	97 OCT 21	97294			Y
QLEGACY	00A968	43,368	QLEGACY@	97 OCT 09	97282			Y
QMFINT	00CFA0	53,152	XPXPXPXP	97 OCT 21	97294			Y
QMFSS	005A80	23,168	XPXPXPXP	97 OCT 09	97282			Y
QMFSS	005BF0	23,536	QSENTRY	97 OCT 09	97282			Y
QM204I	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QM204IN	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QNMDIN	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QOPSYS	036230	221,744	XPXPXPXP	97 OCT 21	97294			Y
QQDSET	005418	21,528	QQDSET@	97 OCT 09	97282			Y

WIBFOC.P7009609.FOCLIB.LOAD  
 LIST BY MEMBERS .. RUN ON 10/22/97 AT 14.24.24

MEMBER	SIZE HEX	SIZE DEC	ENTRY	CREATED ON	YYDD	ALIAS	OVLY	RENT
-----	----	----	-----	-----	----	-----	----	----
QRCAT	0047E0	18,400	QRCAT@	97 OCT 09	97282			Y
QSQLDS	033150	209,232	QSQLDS@	97 OCT 21	97294			Y
QSTRAC	007578	30,072	QSTRAC	97 OCT 09	97282			Y
QSWDB2	013CD0	81,104	QSWDB2@	97 OCT 21	97294			Y
QS2K	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QTOTAL	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QTOTIN	00A968	43,368	QLEGACY@	97 OCT 09	97282	Y		Y
QVSAMX	0222D8	139,992	QVSAMX@	97 OCT 21	97294			Y
RCAT	00E088	57,480	RCAT@	97 OCT 21	97294			Y
RDAAPP	009F40	40,768	RDAAPP	97 OCT 09	97282			
RMH	006758	26,456	RMH@	97 AUG 01	97213			Y
RTH	016BD0	93,136	RTH@	97 OCT 09	97282			Y
SASLIB	06FAD8	457,432	XPXPXPXP	97 OCT 09	97282			Y
SASLIBTR	0186C0	100,032	XPXPXPXP	97 OCT 09	97282			Y
SCHERMES	00BDD8	48,600	HOROADR@	97 OCT 09	97282			
SQLDS	028250	164,432	SQLDS@	97 OCT 21	97294			Y
SQLFOC	0734F0	472,304	SQLHAND@	97 OCT 09	97282			Y
SQLGET	008158	33,112	QSQLGET@	97 OCT 09	97282			Y
SQSRV	00AAB8	43,704	SQSRV1@	97 OCT 21	97294			
SRVINFO	000540	1,344	MSOINFO	97 JUL 16	97197	Y		
SSABSMRT	00BAF8	47,864	SSABSMRT	97 OCT 09	97282			
SSCON	006890	26,768	SSCON	97 OCT 09	97282			Y
SSCONS	02A5B0	173,488	XPXPXPXP	97 OCT 09	97282			
SSCONSEC	000028	40	SSCONSEC	97 JUL 16	97197			Y
SSCRENT	006890	26,768	SSCRENT	97 OCT 09	97282	Y		Y
SSCTL	03A2C8	238,280	SSCTL	97 OCT 21	97294			
SSDDFEOM	0008B0	2,224	SSDDFEOM	97 JUL 16	97197			Y
SSDRDA	009FE0	40,928	SSDRDA	97 OCT 09	97282			Y
SSDWPK	002D08	11,528	DSNLHLI1	97 JUL 16	97197			Y
SSFOC	00BF00	48,896	SSFOC0	97 OCT 09	97282			Y
SSFOCTSO	00BF00	48,896	SSFOCTSO	97 OCT 09	97282	Y		Y
SSGET	0041F0	16,880	SSVX	97 OCT 09	97282			
SSOPER	00A240	41,536	SSVX	97 OCT 09	97282			
SSPUT	003978	14,712	SSVX	97 OCT 09	97282			
SSSOCK	001D48	7,496	SSSOCK@	97 OCT 09	97282			
SSSVCS	003F50	16,208	SSSVCS	97 OCT 09	97282			Y
SSTIME	002FF8	12,280	SSTIME	97 OCT 09	97282			
SUBCSAMD	002B08	11,016	SUBCSAMD	97 SEP 05	97248			
SUBMITX	0007A0	1,952	SUBMITX	97 JUL 16	97197			
SUBPCRMD	000800	2,048	SUBPCRMD	97 JUL 16	97197			
SUBPERMD	000118	280	SUBPERMD	97 JUL 16	97197			
SUBSYSI	0020E8	8,424	SUBSYSI	97 SEP 23	97266			
SUBVECMD	003CA0	15,520	SUBVECMD	97 OCT 21	97294			
SUSI	000780	1,920	SUSI	97 JUL 16	97197			
SVCTEST	000888	2,184	SVCTEST	97 JUL 16	97197			
TSCMFN	007168	29,032	XPXPXPXP	97 OCT 09	97282			
TSCNCP	0060B8	24,760	XPXPXPXP	97 OCT 09	97282			
TSCOM	00A6D0	42,704	CTSCOM	97 OCT 09	97282			

WIBFOC.P7009609.FOCLIB.LOAD  
 LIST BY MEMBERS .. RUN ON 10/22/97 AT 14.24.24

MEMBER	SIZE HEX	SIZE DEC	ENTRY	CREATED ON	YYDDD	ALIAS	OVLY	RENT
-----	----	----	-----	-----	-----	-----	----	----
TSCOML	00A6D0	42,704	TSCOML	97 OCT 09	97282	Y		
TSCOML2	018650	99,920	CTSCML2	97 OCT 09	97282			
TSCOM3	008760	34,656	TSCOM3A	97 OCT 09	97282			Y
TSDB2	00A6D0	42,704	TSDB2	97 OCT 09	97282	Y		
TSDB2L2	018650	99,920	CTSCML2	97 OCT 09	97282	Y		
TSDB2XF	017160	94,560	TSDB2XF	97 OCT 21	97294	Y		
TSEDA	00A6D0	42,704	TSEDA	97 OCT 09	97282	Y		
TSEDAL2	018650	99,920	TSEDAL2	97 OCT 09	97282	Y		
TSEDAXF	017160	94,560	CTSEDAXF	97 OCT 21	97294			
TSFCFN	009FD8	40,920	XPXPXPXP	97 OCT 09	97282			
TSGU	0112A0	70,304	TSGU@	97 OCT 09	97282			
TSGU0	0112A0	70,304	TSGU0	97 OCT 09	97282	Y		
TSHLFN	005350	21,328	XPXPXPXP	97 OCT 09	97282			
TSKILL	004570	17,776	CTSKIL	97 OCT 09	97282			
TS3GL	0038C8	14,536	XPXPXPXP	97 OCT 09	97282			Y
UFAAPI	006238	25,144	XPXPXPXP	97 OCT 09	97282			Y
UTL	01C490	115,856	XPXPXPXP	97 OCT 09	97282			Y
VRPBU0	001AC0	6,848	GETMEMM	97 JUL 16	97197			
VSAMX	01BEA0	114,336	VSAMX@	97 OCT 21	97294			Y
VVISAM	0009C0	2,496	VVISAM	97 JUL 16	97197			
VVSET	00C630	50,736	VVSET@	97 OCT 21	97294			Y
WEBGRP	00AEE0	44,768	WEBGRP@	97 OCT 22	97295			
WFINIT	027E08	163,336	XPXPXPXP	97 OCT 09	97282			Y
WPGEN	02BC50	179,280	XPXPXPXP	97 OCT 09	97282			
WPNT	041BB0	269,232	PMAIN	97 OCT 09	97282			
WTINIT	02C180	180,608	XPXPXPXP	97 OCT 09	97282			Y
XMI	0074D8	29,912	XMI	97 OCT 21	97294			
XMIKIL	0079B8	31,160	XMIKIL	97 OCT 21	97294			
XSBEQ	0031D0	12,752	XSCODE	97 OCT 09	97282			Y
XVSET	009060	36,960	XVSET@	97 OCT 21	97294			Y
XVSRV	0073F8	29,688	XVSRV@	97 OCT 21	97294			Y
X2XBIO	002290	8,848	X2XBIO	97 JUL 16	97197	Y		
X2XBSRC	002290	8,848	X2XBSRC	97 JUL 16	97197	Y		
ZAPCNTL	000128	296	ZAPCNTL	97 JUL 16	97197			

## A Common Mistakes During FOCUS Installation

---

Problem 1	After successfully installing FOCUS, executing your TSOALLOC CLIST to test the installation results in a message that TSOALLOC has failed due to catalog errors.
Cause	SYSIN and SYSPRINT are not allocated in the logon procedure you used to log onto TSO.
Action	Allocate SYSIN and SYSPRINT before running TSOALLOC CLIST
Problem 2	When you attempt to test use of the FOCUS SVC in zapped modules, you get the message:  "FD10000 - SVCnD1(nnn) HAS A PROBLEM"
Cause	<ol style="list-style-type: none"><li>1. SVC was not a type 3 or 4.</li><li>2. SVC was not enabled.</li><li>3. SVC was not put in SVC table and assembled.</li><li>4. System was not IPLed after the SVC was installed.</li></ol>
Action	<ol style="list-style-type: none"><li>1. Make sure SVC is type 3 or 4, enabled, and properly assembled into the SVC table.</li><li>2. Make sure your system has been IPLed after the FOCUS SVC was installed in the SVC table.</li></ol>
Problem 3	On some systems you cannot allocate files through ISPF that have RECFM=U.
Cause	Your system has been set up to restrict you from doing this within ISPF.
Action	Allocate FOCLIB.LOAD and FUSELIB.LOAD with a batch job. Remember not to specify an LRECL when allocating files with RECFM=U.
Problem 4	0C1 abend occurs when starting the sink machine.
Cause	Use of the old FOCUS SVC from Release 6.0 or a subsequent release.
Action	Reinstall the SVC from a Release 7.0 or later tape, making sure it is link-edited with AMODE=31, RMODE=24.

## B Sample JCL for the Central Database Job

---

```

//HLISNK      EXEC  PGM=HLISNK,PARM='parameters - see below'
//STEPLIB    DD    DSN=prefix.FOCLIB.LOAD,DISP=SHR
//FOCLIB     DD    DSN=prefix.FOCLIB.LOAD,DISP=SHR
//ERRORS     DD    DSN=prefix.ERRORS.DATA,DISP=SHR
//*
//*          FOCUSU is the Communication dataset
//*
//FOCSU      DD    DSN=FOCSU.DATA,DISP=SHR
//*
//*          FOCUSU is a dummy FOCUS database created during
//*          the SU installation
//*
//FOCUSSU    DD    DSN=FOCUSSU.FOCUS,DISP=SHR
//*
//*          FOCSSORT is needed if the COMBINE/SU
//*          feature is used
//*
//FOCSORT    DD  SPACE=(TRK,(1,1)),UNIT=SYSDA
//*
//*          Sink error log
//*
//HLIERROR   DD    SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//*
//*          Optional log file (can be a dataset or printer)
//*
//HLIPRINT   DD    SYSOUT=*,DCB=(RECFM=FB,LRECL=88,BLKSIZE=8800)
//*         OR  DD    SYSOUT=*,DCB=(LRECL=133,RECFM=FBA,BLKSIZE=13300)
//*
//*          MASTER contains the file description of all
//*          databases used in SU mode
//*          NOTE only datasets with identical LRECLs
//*          and RECFMs should be concatenated
//*
//MASTER     DD    DSN=prefix.MASTER.DATA,DISP=SHR,
//           DD    DSN=prefix.FOCCTL.DATA,DISP=SHR
//*
//*
//FOCEXEC    DD    DSN=prefix.FOCEXEC.DATA,DISP=SHR
//*
//*
//*          Here are the actual centrally controlled FOCUS
//*          databases allocated with DISP=SHR so multi-threaded
//*          SU reporting facility can be used.
//*
//EMPLOYEE   DD    DSN=prefix.EMP02.FOCUS,DISP=SHR
//EDUCFILE    DD    DSN=prefix.EDC05.FOCUS,DISP=SHR
//JOBFILE     DD    DSN=prefix.JOB15.FOCUS.DISP=SHR

```

.

etc.

This job must not be allowed to time out. Its virtual storage requirements are discussed in Chapter 1. See also the *Simultaneous Usage Reference Manual, TSO Version*, for optional parameters for this job.

## C Subsystem Error Messages

---

The following messages are issued by SUBSYSI. They are written to the MVS system console by a WTO.

These messages are written directly by SUBSYSI; they are not read from ERRORS.DATA. There is no national language support for messages issued by SUBSYSI.

**SUBSYS001 UNABLE TO CREATE SUBSYSTEM**

An error has been encountered while creating the subsystem. Contact your IBI support representative.

**SUBSYS002 CREATED SUBSYSTEM NOT FOUND**

The subsystem search has failed even though the subsystem has just been created. Contact your IBI support representative.

**SUBSYS003 UNABLE TO LOAD MODULES TO CSA**

Error during LOAD of modules to CSA. Refer to the accompanying CSVxxxx messages for further information.

**SUBSYS004 INVALID PARAMETERS PASSED**

An invalid parameter has been passed to SUBSYSI. Valid parameters are: START, RESTART, REPLACE, STOP, and REMOVE.

**SUBSYS005 SUBSYSTEM XXXX NOT FOUND. UNABLE TO STOP**

The specified subsystem cannot be located; the requested operation will not be performed.

**SUBSYS006 SUBSYSTEM XXXX NOT FOUND. UNABLE TO REPLACE**

The specified subsystem cannot be located; the requested operation will not be performed.

**SUBSYS007 SUBSYSTEM XXXX NOT FOUND. UNABLE TO REMOVE**

The specified subsystem cannot be located; the requested operation will not be performed.

**SUBSYS008 SUBSYSTEM XXXX NOT FOUND. UNABLE TO INFO**

The specified subsystem cannot be located; the requested operation will not be performed.

**SUBSYS009 UNABLE TO ACTIVATE PC ROUTINE**

A bad return code was received during activation of the PC routines. Contact your IBI support representative.

**SUBSYS010 UNABLE TO RUN NON-AUTHORIZED**

An attempt was made to run SUBSYSI from an unauthorized library. Copy SUBSYSI and the associated modules to an APF-authorized library and try again.

**SUBSYS100 SUBSYSTEM XXXX SUCCESSFULLY CREATED**

The subsystem control blocks have been created and linked to the system control blocks.

**SUBSYS101 SUBSYSTEM XXXX SUCCESSFULLY INITIALIZED**

The subsystem has been initialized and all control blocks have been filled in.



- SUBSYS102 SUBSYSTEM XXXX DOES NOT EXIST. START MODE ASSUMED**  
SUBSYSI was called with 'RESTART', but it has not been initialized. 'START' will be assumed.
- SUBSYS103 SUBSYSTEM XXXX ALREADY EXISTS. RESTART MODE ASSUMED**  
SUBSYSI was called with 'START' but it was already initialized. 'RESTART' will be assumed.
- SUBSYS104 SUBSYSTEM XXXX ENVIRONMENT REPLACED**  
SUBSYSI was called with 'REPLACE' and the operation succeeded.
- SUBSYS105 SUBSYSTEM XXXX STOPPED**  
The subsystem is logically stopped. The CSA modules are still loaded but are not active.
- SUBSYS106 SUBSYSTEM XXXX INTERFACE REMOVED**  
The subsystem has been stopped, and modules have been removed from the CSA. The SSCT and the PC routine still exist, but are unused.
- SUBSYS107 SUBSYSTEM XXXX COUNTER ERROR OCCURRED**  
The subsystem has been stopped, but one or more users are still using it. A 30-second wait to allow all activity to stop has not fixed this.
- This message generally indicates that the subsystem was stopped while one or more IBI servers were still running. If this is not the case, contact your IBI support representative.

## D Subsystem Sample JCL and Zaps

---

The following members of FOCCTL.DATA contain sample JCL and zaps for your use. They are reproduced here for reference.

Member	Contents
SUBSYSZP	Zap to activate the IBI Subsystem
SUBSYSNM	Zap to change the subsystem name
SUBSYSCP	JCL to copy SUBSYSI and related modules into a separate library
SUBSYSIJ	JCL to run SUBSYSI
SUBSYSIH	JCL to run SUBSYSI and initialize HiperBUDGET
SUBSYSP	Sample cataloged procedure for running SUBSYSI as a started task

*Table 1: Sample JCL and zaps in FOCCTL.DATA*

## D.1 SUBSYSZP: Zap to Activate the IBI Subsystem

---

```

//*** Your JOB card
//*
//*-----*
//*
//* This zap must be applied to enable use of the IBI Subsystem *
//* instead of the IBI-supplied SVC. *
//* *
//* Affected products: *
//*      SU, MSO, SmartMode, HiperFOCUS, IMS BMP *
//* *
//* To reactivate the SVC, change the REP statement to *
//*      "REP 00 0Axx" *
//* where "xx" is the (hex) SVC number to use. *
//* *
//*-----*
//*
//ZAP      EXEC PGM=AMASPZAP
//SYSPRINT DD  SYSOUT=*
//SYSLIB   DD  DISP=SHR,DSN=prefix.FOCLIB.LOAD
//SYSIN    DD  *
            NAME MVSDLL  IBISVCEX
            REP  00  0000
//*
//

```

## D.2 SUBSYSNM: Zap to Change the Subsystem Name

---

```

//*** Your JOB card
//*
//*-----*
//*
//* This zap may be applied to change the default subsystem name *
//* used by the IBI Subsystem. *
//*
//* If you have defined the IBI Subsystem in SYS1.PARMLIB(IEFSSNxx), *
//* be sure to update the definition with the new subsystem name. *
//*
//* As this job updates one of the members copied by SUBSYSCP, *
//* SUBSYSCP must be run after this JOB. *
//*
//* Affected products: *
//*      SU, MSO, SmartMode, HiperFOCUS, IMS BMP *
//*
//*-----*
//*
//ZAP      EXEC PGM=AMASPZAP
//SYSPRINT DD  SYSOUT=*
//SYSLIB   DD  DISP=SHR,DSN=prefix.FOCLIB.LOAD
//SYSIN    DD  *
           NAME MVSDLL  SUBSYSN
           REP  00  C9C2C9E2          <-- 'IBIS'
           NAME SUBSYSI  SUBSYSN
           REP  00  C9C2C9E2          <-- 'IBIS'
//*
//

```

## D.3 SUBSYSCP: JCL to Copy SUBSYSI Into a Separate Library

---

```

//*** Your JOB card
//*
//*-----*
//*
//* This job can be used to move the IBI Subsystem modules from
//* FOCLIB.LOAD to a secured library. The target library must be
//* APF-authorized.
//*
//* If the zap provided in SUBSYSNM has been run, ensure that this
//* job is re-run.
//*
//* Remove all members of SSLIB prior to running this.
//*
//* It is recommended that you delete the modules from FOCLIB.LOAD
//* after this job is run.
//*
//*-----*
//*
//COPY      EXEC PGM=IEBCOPY
//FOCLIB    DD   DISP=SHR,DSN=prefix.FOCLIB.LOAD
//SSLIB     DD   DISP=OLD,DSN=prefix.SUBSYS.LOAD
//SYSIN     DD   *
            COPYMOD INDD=FOCLIB,OUTDD=SSLIB
            S M=(SUBSYSI,SUBCSAMD,SUBPCRMD,SUBPERMD,SUBVECMD)
//*
//

```

## D.4 SUBSYSIJ: JCL to run SUBSYSI

---

```

    /*** Your JOB card
    /**
    /**-----*
    /**
    /** Sample JCL to run SUBSYSI, used for starting and controlling *
    /** the IBI Subsystem. (Additional controls are available from *
    /** the system console; refer to the appropriate product *
    /** documentation for details.) To run SUBSYSI as a started task, *
    /** refer to member SUBSYSP. *
    /** *
    /** SUBSYSI must be run from an APF-authorized library. *
    /** *
    /** Update the PARM field to select the desired operation. *
    /** Valid parameters are: START, RESTART, REPLACE, STOP, REMOVE *
    /** *
    /**-----*
    /**
    /**SUBSYSI EXEC PGM=SUBSYSI,PARM=START
    /**STEPLIB DD DISP=SHR,DSN=prefix.SUBSYS.LOAD
    /**
  
```

## D.5 SUBSYSIH: JCL to Run SUBSYSI and Initialize HiperBUDGET

---

```

    /*** Your JOB card
    /**
    /**-----*
    /**
    /**
    /** Sample JCL for starting SUBSYSI and initializing HiperBUDGET.
    /**
    /** Refer to the product documentation for further information on
    /** SUBSYSI and on HiperBUDGET.
    /**
    /** The sample SYSIN input sets no limit on any Hiper resources,
    /** and then displays the current settings.
    /**
    /**-----*
    /**
    /**SUBSYSI EXEC PGM=SUBSYSI,PARM=START
    /**STEPLIB DD DISP=SHR,DSN=prefix.SUBSYS.LOAD
    /**SYSIN DD *
    SET MVSLIM=-1,SERVLIM=-1,TCBLIM=-1,FILELIM=-1
    DISPLAY HIPERBUDGET
    /**
  
```

## D.6 SUBSYSP: Sample Cataloged Procedure

---

```
/*-----*
/*
/* SUBSYSP: Sample cataloged procedure for running SUBSYSI as a  *
/* started task. To run SUBSYSI as a batch job,                *
/* refer to member SUBSYSI.                                     *
/*                                                              *
/*-----*
/*
/*SUBSYSI PROC OPT=START
/*
/*SUBSYSI EXEC PGM=SUBSYSI,PARM=&OPT
/*STEPLIB DD DISP=SHR,DSN=prefix.SUBSYS.LOAD
/*
/*-----*
```



# Reader Comments

---

---

In an ongoing effort to produce effective documentation, the Documentation Services staff at Information Builders welcomes any opinion you can offer regarding this manual.

Please use this form to relay suggestions for improving this publication or to alert us to corrections. Identify specific pages where applicable. Send comments to:

Corporate Publications  
Attn: Manager of Documentation Services  
Information Builders  
Two Penn Plaza  
New York, NY 10121-2898

or FAX this page to (212) 967-6406, or call Sara Elam at (212) 736-4433 x3207.

Name: \_\_\_\_\_

Company: \_\_\_\_\_

Address: \_\_\_\_\_

Telephone: \_\_\_\_\_ Date: \_\_\_\_\_

Comments:

# Reader Comments

---

**Information Builders, Two Penn Plaza, New York, NY 10121-2898**

**(212) 736-4433**

FOCUS for IBM Mainframe

DN1000994.1097

MVS/TSO Installation Guide Release 7.0