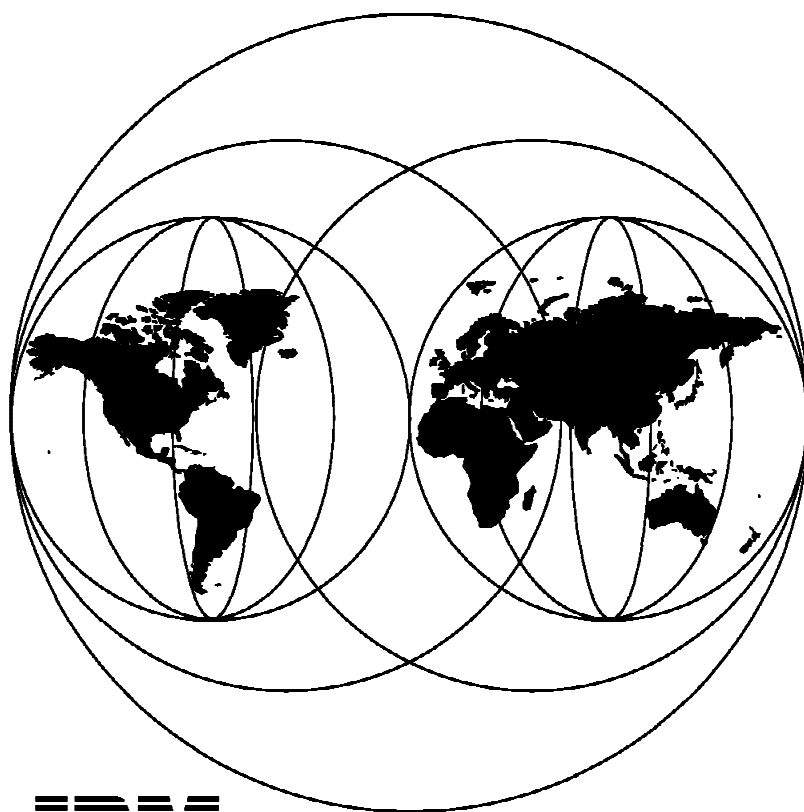


Image and Workflow Library: ImagePlus MVS/ESA Performance and Capacity Planning Benchmark

July 1997



IBM

**International Technical Support Organization
Poughkeepsie Center**



International Technical Support Organization

SG24-4975-00

**Image and Workflow Library: ImagePlus MVS/ESA
Performance and Capacity Planning Benchmark**

July 1997

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix E, "Special Notices" on page 91.

First Edition (July 1997)

This edition applies to Version 2 Release 2 of IBM SAA ImagePlus MVS/ESA, Program Numbers 5695-041 and 5695-042, for use with the MVS/ESA Operating System.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HYJ Mail Station P099
522 South Road
Poughkeepsie, New York 12601-5400

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1997. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Preface	v
The Team That Wrote This Redbook	v
Comments Welcome	vi
Chapter 1. Introduction to ImagePlus MVS/ESA	1
1.1 What is ImagePlus MVS/ESA?	1
1.1.1 Capabilities	2
1.1.2 Components	2
Chapter 2. Description of Our Scenarios	5
2.1 Purpose of This Project	5
2.2 A Typical IBM SAA ImagePlus MVS/ESA Customer	5
2.3 Our Business Scenario	5
2.4 Our TPNS Scripts	6
2.4.1 Index and Scan	6
2.4.2 Getwork and Windup Loop	7
2.4.3 Workflow Function	7
2.5 Setting Up the IPFAF Databases	7
2.5.1 The Static Tables	8
2.5.2 Folder and Object Tables	9
2.5.3 Our Use of ImagePlus Terminology	9
Chapter 3. Hardware and Software Environment	11
3.1 Hardware	11
3.1.1 Processor	11
3.1.2 DASD	11
3.2 Software	12
3.3 Setting Up DB2	12
3.4 Setting Up CICS	13
3.5 Setting Up IPFAF and IODM	13
3.5.1 Splitting IPFAF and IODM	13
3.5.2 Some Key Tables	14
3.5.3 LU 6.2 Connections and Definitions	14
3.5.4 Considerations When Displaying or Printing	16
Chapter 4. Information from the Benchmarks	17
4.1 Conclusions	17
4.1.1 Response Time Figures	17
4.1.2 CPU Utilization	18
4.2 Run 1	18
4.3 Run 2	20
4.4 Run 3	21
4.5 Caveats	22
Chapter 5. How to Use This Information	23
5.1 Indexing and Scanning	23
5.2 Getwork Windup Script	24
5.3 Workflow Functions Script	25
5.4 An Approach to Capacity Planning	25
5.4.1 Data to Gather	26
5.4.2 Applying This Data	26

5.4.3 Changes from Previous Scenario	27
5.5 Caveats	28
Chapter 6. Tuning Tips	29
6.1 DB2 Changes	29
6.1.1 Single Threading Background Transactions	29
6.1.2 Use of Packages	30
6.2 Application Design	30
6.3 Preloading the Work Detail Table	31
Appendix A. DB2 ZPARMS and RCTs	33
A.1 ZPARMS	33
A.2 Original RCT for IODM	37
A.3 Original RCT for IPFAF	39
A.4 Final RCT for IODM	49
A.5 Final RCT for IPFAF	50
A.6 DB2 Package and Plan for IODM and IPFAF	58
A.6.1 Sample: DB2 Package for IODM and IPFAF	58
A.6.2 Sample: DB2 Plan for IODM and IPFAF	62
Appendix B. CICS SIT Tables and SYSIN Definitions for IODM and IPFAF	63
B.1 IODM SIT Table	63
B.1.1 ODMD CICS Region SYSIN	64
B.1.2 ODM2 CICS Region SYSIN	64
B.2 IPFAF SIT Table	66
B.2.1 FAFD CICS Region SYSIN	68
B.2.2 FAF2 CICS Region SYSIN	68
Appendix C. IODM Exit: EKCCDDTE	69
Appendix D. Software Products and Service Levels	85
D.1 Core Sysplex Software Products Installed	85
D.1.1 Core OS/390 Elements and Features Installed	85
D.1.2 Other Core Sysplex Software Products Installed	85
D.2 Additional OS/390 Elements and Features Installed	87
D.3 Other Sysplex Software Products Installed	89
Appendix E. Special Notices	91
Appendix F. Related Publications	93
F.1 International Technical Support Organization Publications	93
F.2 Redbooks on CD-ROMs	93
F.3 Other Publications	93
How to Get ITSO Redbooks	95
How IBM Employees Can Get ITSO Redbooks	95
How Customers Can Get ITSO Redbooks	96
IBM Redbook Order Form	97
Index	99
ITSO Redbook Evaluation	101

Preface

Have you ever wondered if your ImagePlus/MVS system was about to expand and run into a bottleneck of some kind? We wondered also, so we organized a benchmark to find out using TPNS and OmegaMon. This redbook contains the details of the results. We also included some tuning tips and we tell you how to apply this information to your situation.

This redbook was written for system consultants, architects, planners, database administrators, system programmers, and technical support people who need to make capacity and performance recommendations about ImagePlus/MVS.

Relevant data from our benchmark runs is included. Some knowledge of ImagePlus/MVS, DB2, and MVS is assumed.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Poughkeepsie Center.

Charles Barnwell is a Systems Specialist in the UK. He has 7 years of experience in the ImagePlus field. He holds a degree in Physics from University College, London University. His areas of expertise include ImagePlus, MVS, and OS/2.

Jim Tau is a Senior Operational Support Specialist at the Image and Digital Library Competency Center in the U.S.A. He has been working in the image field specializing in MVS-related image products since 1992. He was formerly a member of International Technical Support Organization, Bethesda Center.

This project was sponsored by the ITSO.

Mike Ebbers is a Senior Software Engineer at the International Technical Support Organization, Poughkeepsie Center. He has worked at IBM for 23 years. He produces redbooks and workshops on workflow and imaging products.

Thanks to the following people for their invaluable contributions to this project:

Richard Howell
Midland Bank United Kingdom

Kevin Cook
IBM United Kingdom

Gary Hunter
IBM United Kingdom

And to the following departments:

IBM EDMSuite and Digital Library Worldwide Sales and Support
IBM ImagePlus Development
IBM S/390 Platform Evaluation Test

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in "ITSO Redbook Evaluation" on page 101 to the fax number shown on the form.
- Use the electronic evaluation form found on the Redbooks Web Sites:

For Internet users <http://www.redbooks.ibm.com>

For IBM Intranet users <http://w3.itso.ibm.com>

- Send us a note at the following address:

redbook@vnet.ibm.com

Chapter 1. Introduction to ImagePlus MVS/ESA

Most office information today is kept on paper. With image processing, you can convert this information into an electronic form for online processing. Document images are stored, retrieved, distributed, and processed just as paper documents are. But you enjoy increased efficiency and productivity because the time taken to retrieve documents is reduced from hours or even days to just seconds. In addition, you can achieve:

- Improved client and customer service
- Improved decision making
- Reduced clerical support and storage costs
- Improved management control
- Improved data security

By offering a fundamental reshaping of the way business documents are handled, image processing represents the next step in the evolution of office automation.

When a document is scanned into the system, it is immediately available to all the users in the system for concurrent access. You can scan logos, signatures, photographs, and other nonkeyable objects (both color and black and white) that you could not previously store online. The document image contains every element that appears on the original document, including text and graphics, signatures, notes, and drawings.

1.1 What is ImagePlus MVS/ESA?

The IBM SAA ImagePlus MVS/ESA system is a large-scale IBM system solution for converting a paper-based storage and retrieval system to a computer-based system. It brings industrial strength image processing to System/390 environments by using ImagePlus workstations. The ImagePlus system provides the components needed to add high-volume image processing to multiple areas of your enterprise. It utilizes DB2 and CICS, providing the application integrity and availability of the S/390 environment to the imaging system. An IMS version is also available.

It combines the advantages of operating in a Local Area Network (LAN) environment with the powerful processing capabilities of the System/390 mainframe environment. ImagePlus/MVS is capable of connecting over a thousand users at local and remote settings, while supporting the processing of tens of thousands of document images. This can be a competitive advantage for your enterprise.

ImagePlus/MVS protects your investment through use of strategic hardware and software. It is designed utilizing standard image architectures and protocols. It provides you with capabilities for growth, both at a central site and (with IODM/2) at remote sites in a LAN environment. ImagePlus/MVS has application programming interfaces (APIs) that allow you to have seamless integration with your line-of-business applications.

1.1.1 Capabilities

As an operational imaging system, ImagePlus/MVS has the following functions:

- Document capture process (scan a hard-copy document).
- High capacity storage (store an image of the document).
- Register the document in a folder (index the document).
- Retrieve/display/print an image of the document.
- Workflow management (route and distribute for processing).
- Document manipulation and processing.
- Document export and fax support.
- Remote and distributed processing.

It also has the following features:

- Fast response for an image display request
- Page flip time of less than one second
- The ability to compare pages and to display a full page on the window
- Automatic and manual routing of images to users
- Fast and easy-to-use ad hoc scanning facilities, and batch and high-speed capture for large volumes of documents
- Ability to print documents
- Limited editing facilities (such as adding or deleting a page)
- Faxing in and out

1.1.2 Components

The IBM SAA ImagePlus MVS/ESA system consists of the these major components:

- IBM ImagePlus software:
 - Folder Application Facility (IPFAF)
 - Object Distribution Manager (IODM)
 - Object Access Method (OAM)
- Optical Storage Subsystem Products (OSSP)
- Image workstations

1.1.2.1 ImagePlus/MVS Software

The ImagePlus/MVS software components are as follows:

- IBM SAA ImagePlus Object Distribution Manager MVS/ESA (IODM -- object routing server):

The IODM provides basic routing services for object movement. Upon receipt of requests from the Folder Application Facility, it communicates with image workstations to capture incoming documents and to forward documents for viewing, printing, or processing. Storage and retrieval requests are coordinated utilizing the Object Access Method. In a multi-site environment, SNA LU6.2 protocols are used to communicate with remote IODM nodes.

- IBM SAA ImagePlus Folder Application Facility MVS/ESA (IPFAF -- index server):
 - Application Programming Interface (IPFAF API):
 - Document Services API
 - Folder and Document Management API

- Batch API

- Folder and Workflow Application (IPFAF FWA):

IPFAF provides functions necessary to manage image-based applications in two basic categories: folder management and workflow management. Folder management allows captured images to be grouped into a logical construct called a folder, analogous to a manila file folder used for paper. The workflow management includes routing of image documents to various work queues, distribution for processing by users, tracking of work backlogs and assignments, and redirection of work in process to other users if necessary. IPFAF maintains indexes of every object in the system and authorizes all requests for storage and retrieval.

- IBM MVS/DFP Object Access Method (OAM -- object storage server):

The OAM, which provides object storage management functions, is a component of MVS/DFP (also comes with DFSMS/MVS). OAM builds on the concepts and functions of Data Facility Storage Management Subsystem (DFSMS). It uses DB2 as the database manager for object directories, for configuration knowledge of the OSSP, and for objects while they are stored on magnetic DASD. DB2 also provides logging and recovery functions to maintain the integrity of stored data. The OAM manages movement of objects between magnetic and optical storage.

- IBM SAA ImagePlus Object Distribution Manager MVS/ESA/2 (IODM/2)

- IBM SAA ImagePlus Capture Facility (ICPF)

- IBM SAA ImagePlus Intelligent Forms Facility/2 (IPFO):

- IBM SAA ImagePlus Intelligent Forms Assist/2 (IPFA)

- FaxRouter/2 High-performance Facsimile Solution Version 2.0

1.1.2.2 Optical Storage Subsystem

The Optical Storage Subsystem Products (OSSP) provide storage for a large number of objects with relatively infrequent retrieval rates. They consist of optical libraries and drives and are based on write once optical storage technology. The OSSP are controlled by the OAM component of MVS/DFP. Over 2500000 scanned images at 50 KB per page can be stored in one library unit.

Here is a set of optical storage subsystem products:

- IBM 3995 Optical Library Dataserver Products

1.1.2.3 Image Workstations

The user workstations are personal computers with one of the following software products:

- IBM ImagePlus Workstation Program/2 (IWPM for OS/2). Any display unit supported by OS/2 Presentation Manager works for ImagePlus/MVS.
- IBM ImagePlus Workstation Program V2.2 for Windows NT
- IBM ImagePlus Workstation Program/DOS for Windows (IWPM/DOS for Windows)

The workstations have two sessions. An LU2 session communicates with IPFAF through a CICS-3270 interface to convey user commands, and an LU6.2 session displays the images from IODM.

Chapter 2. Description of Our Scenarios

Here we introduce the project and the environment, and describe the benchmark scenarios we ran to simulate this environment.

2.1 Purpose of This Project

The customers we have worked with find that ImagePlus/MVS quickly becomes a critical part of their business. As the customers are increasing their use of ImagePlus/MVS and also their dependence on it, we begin to ask how much work the ImagePlus/MVS system can handle.

The purpose of this project is to take a large machine and see how many ImagePlus/MVS transactions the system can handle. We want to understand if there is a bottleneck in the foreseeable future.

The project also allows us to see how much resource the ImagePlus/MVS system uses when it is driven hard.

This chapter describes the scenarios we used to run the benchmarks.

2.2 A Typical IBM SAA ImagePlus MVS/ESA Customer

In setting up our scenarios, we considered our experience with typical ImagePlus/MVS customers. The customer will normally be a large enterprise with paper intensive processes. Two examples are insurance companies and financial institutions. We often found that the paper processes are handled at more than one site, so its ability to store centrally but distribute the paper to many different sites is an advantage.

Often there is a requirement to process the paper centrally at a head office, or distributed service centers, but be able to view the state of a customer's folder, and possibly images themselves, from different sites (call centers or branches, for example).

The customer will probably have a number of applications running on an MVS/ESA system, so IPFAF's ability to integrate with these applications (such as performing validation by checking current databases) is key to the implementation.

ImagePlus/MVS uses standard IBM architectures (SNA, DB2, CICS or IMS), so our typical customer would normally have experience with these systems and their own applications that make use of them. This chapter describes how we designed our scenarios to fit this typical customer.

2.3 Our Business Scenario

The business scenario is based on our experience of customers that use ImagePlus in a live environment. IPFAF provides a structure of tables to deliver a piece of work to the correct queue at the correct priority to ensure that the business commitments for any particular item of work are met.

2.4 Our TPNS Scripts

We have three basic TPNS scripts representing various types of transactions. These are:

- Index and scan
- Getwork and windup loop
- Workflow functions

We chose our scripts to reflect a realistic workload for a customer, and also to allow us to exercise the active IPFAF tables. We made the scripts as flexible as possible so that we could change the workload characteristics to meet our needs.

Parameters are available in the TPNS scripts to allow us to change the percentages of users performing these individual scripts, the pattern of work within the script, and the time spent waiting after finishing a script. We can also change the think time.

2.4.1 Index and Scan

This script simulates the process of indexing and scanning documents into the ImagePlus system. It can either store documents into a folder without routing the documents, or store documents into a folder and also place the document in routing for subsequent working. The first part of the script indexes the document. This is performed by an IPFAF ADOC to index the document into a folder with no routing, or an IPFAF ADOR command to index the document into a folder and also route the document for subsequent work. When this transaction is complete, IPFAF produces a TEMPID. This is a temporary document ID that the ImagePlus/MVS system uses to identify the document when scanning. The second half of the script uses this TEMPID to store the document into IODM.

This script performs an ADOR or ADOC transaction within IPFAF. It enters the command "ADOR fldr,formnum" or "ADOC fldr,formnum," choosing randomly the folderid and formnum from a list of valid ones. At the add and route window or add window, it presses Enter to generate a TEMPID. Once the TEMPID has been received, the script switches to the LU6.2 session established with IODM and stores a document. Once the store operation is complete, the script returns to the main menu by pressing PF12. After waiting a certain time, the script restarts the operation.

We have four kinds of documents that we can store:

Document type	Document size
Tiny	4043 bytes
Small	11 892 bytes
Medium	40 184 bytes
Large	113 678 bytes

We can change the ratio of ADORs to ADOCs, and the percentages of tiny, small, medium, and large documents that are stored during the storage process. The time spent waiting after a complete script is also modifiable by parameters.

2.4.2 Getwork and Windup Loop

This script simulates an end user who is working the documents that have been placed into routing by the indexing and scanning scripts. A user typically performs an IPFAF Getwork transaction to ask for the highest priority document in the workflow system that this user is authorized to work. Having performed this, the user can do whatever business transaction is required by the item of work and, on returning to IPFAF, perform an IPFAF Windup transaction to complete the work. When performing a Windup, a user either drops the document (no further work is required the document is filed) or reroutes the document (further work is required by another user). It chooses option 1 from the main menu. When the getwork list window is shown, it sometimes chooses option 1 to display the history of the document and returns to the getwork list window by pressing PF12. It optionally chooses to display the folder contents. It does this by pressing PF3 and entering FLDR in the command line on the main menu. It returns to the main menu by pressing PF3 and chooses option 1 again to return to the getwork list window. The user then performs a windup by pressing PF10 on the getwork list window. In the windup window, the script chooses either to drop the document by entering option 7, or reroute the work by choosing option 5. When the reroute or drop is complete, the script presses PF12 twice to return to the main menu. The script then waits for an optional period of time before choosing option "1" again to restart the script.

We can modify the percentage of getworks after which the user displays the history, the percentage of time a user lists the folder contents, and the percentage of time a person does a reroute versus a drop action at windup. We can also change the time a script waits after completing a cycle.

2.4.3 Workflow Function

The workflow functions simulate a supervisor checking the status of work in a queue. The function chooses option 7 from the main menu followed by option 6 from the workflow functions window. It then randomly chooses a unit code, enters it on the window, and presses Enter to display the information. The script pages forward if other pages are available and uses PF12 twice to return to the main menu.

By using TPNS parameters, we can combine these scripts in different ways to allow us to simulate a real-life workload. See Chapter 4, "Information from the Benchmarks" on page 17 for the results.

2.5 Setting Up the IPFAF Databases

There are two areas where the IPFAF databases need to be set up. First, we need to define a routing scheme in the IPFAF static configuration tables. Second, we need to load the folder and object tables with a realistic number of documents.

We also want to be able to change these values to see if they affect resource utilization. This section describes how we can use these tables to simulate our business scenarios.

2.5.1 The Static Tables

To produce a realistic benchmark reflecting the environment of a true high end user of IPFAF, we feel it is important to customize the IPFAF databases to reflect a typical user. To do this, we set up a routing scheme to simulate the system load that a live user experiences.

The tables we customized are:

- Customer table (ENTTUCDX)
- Form Number table (ENTTTFRM)
- RLOB/Transaction Type table (EYPTWRTT)
- Unit Code table (ENTTWUNT)
- Valid Unit/RCODE Queue Definition table (EYPTWURC)
- User Profile Table (ENTTTUPR)
- User Assignment table (EYPTWEAS)

Figure 1 shows the relationship of these tables.

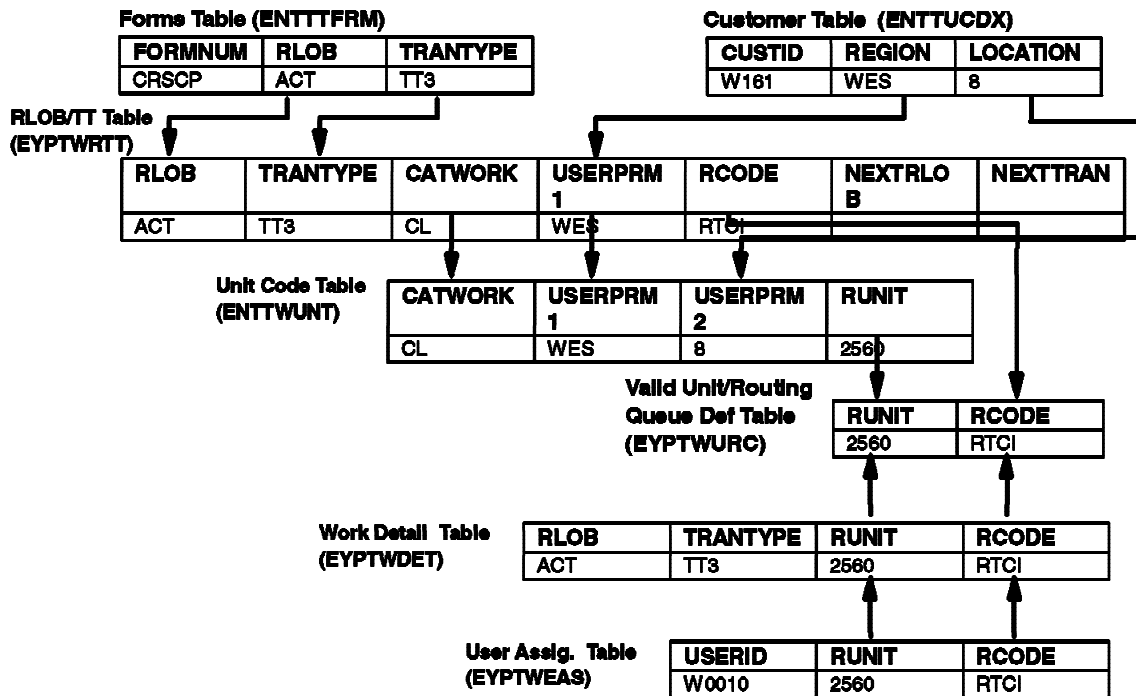


Figure 1. FWA / API Workflow Table Relationship

We started with some basic assumptions. First, we decided that we want a maximum of 1000 queues. There is a rule of thumb which states that there is one queue for each user in the system (this number may vary depending upon the application). We, therefore, decided to start with a user base of 1000.

Then, we decided to use the sample customer table provided with the sample data for IPFAF to allow us to validate our FOLDERIDs and to provide us with our line three data. We, therefore, populated the sample customer table with 100000 rows of valid folderids. These are randomly generated 12 character fields. Region and location are used for USERPRM1 and USERPRM2 and we use the same values as in the sample data (EAS and WES for USERPRM1 and 6, 7, or 8 for USERPRM2).

In our company, we chose five Routing Line of Businesses (RLOBS), and within each RLOB we have 100 Transaction Types (TTypes). So our form number table consists of 1000 randomly generated six-digit form numbers with an RLOB and TType randomly chosen from our list.

We decided to have 18 UNIT codes and 55 route codes in each of these units. To generate the RLOB/TType table, we associate an RCODE from a random selection of 55 valid RCODEs with each possible RLOB/TRANTYPE/USERPRM1 combination. The USERPRM1 field is provided by the sample Customer Data exit from the sample Customer table. The category of work is randomly chosen from three possible Categories of Work (C1, C2, or C3).

One row in the UNIT code table is generated for every possible USERPRM1, USERPRM2, and category of work combination. This gives us our eighteen possible Unit Codes ($2*3*3=18$).

EYPTWURC is generated by simply having an entry for every possible UNIT and route code combination. Eighteen unit codes times 55 route codes gives us the total possible of 990 queues.

The naming convention for our user IDs is three characters followed by three numbers. Users who have the same PRIMUNIT have the same first three characters; we then simply use subsequent numbers. We begin with 55 users per unit code. Our first unit code has users ABC001 through ABC055, and the eighteenth Unit Code has users RST001 through RST055. If we want to increase our number of users, we simply increase the number of users per unit code.

The User Assignment table describes those queues that a user is authorized to work. We chose to have users assigned to 10 route codes within their PRIMUNIT value. Thus, we have 10 entries per user for a total of 9900 entries.

2.5.2 Folder and Object Tables

Since one of our TPNS scripts is an Index and Scan scenario, we use this TPNS script to populate the Folder and Workflow tables. We decided to add 500 000 objects randomly inserted into our 100 000 possible folders. In reality, we are limited by access to machine time so we created only 50 000 objects.

2.5.3 Our Use of ImagePlus Terminology

The terms used in IPFAF workflow tables are not necessarily intuitive. This is how most of our customers use the terms within IPFAF:

Term	Our definition
FormNum	The characteristics of the document that has been scanned. This includes storage management and routing characteristics. From the columns RLOB and TranType, we determined what initial routing to perform on the document.
RLOB	Routing Line of Business. We use this to denote a business section that is dealing with this document. Examples of what these might be are branches, service centers handling life insurance, service centers handling pensions, call centers, insurance claims. In our example, we have five such business sections.

TranType	This is the type of work that needs to be performed on this document (for example, initial claim form, life insurance application, medical report). Within each RLOB, we have 100 different TranTypes.
Userprm1/2	These fields are provided by user exits within the IPFAF. Typically, they supply the characteristics of a particular customer. For example, if there are five district service centers that handle claims for a subset of customers, this field is used to ensure that a claim for a particular customer is handled by the correct service center.
Catwork	This field is simply used to normalize the tables. It is a link between the RLOB/TranType table and the Unit Code table to ensure that a particular piece of work goes to the correct unit.
RCODE	The routing code is used along with unit code to define the queue that the work is placed on. Typically, it describes a skill level of the type of user that can handle this kind of work.
Unit Code	Used with RCODE to define the queue. This is usually used to describe a group within the business. To take the example of a number of district service centers within a company, each center has its own unique unit code, but all of them might have the same Rcodes because they all handle the same work.

Chapter 3. Hardware and Software Environment

This chapter describes the environment in which the benchmarks were run, along with a description of those tasks we performed to customize the system.

3.1 Hardware

The following sections contain the hardware (processor and DASD configurations) that we used.

3.1.1 Processor

The processor we were running on is an ES/9000 9021 model 982 with 776M of central storage and 1.5GB of expanded storage. TPNS was running on another processor so the machine we were running on was dedicated to our ImagePlus software.

TPNS was running on a 9672 RX4 in LPAR mode. The machine was configured so that we had half of the RX4. Here is a picture of the configuration.

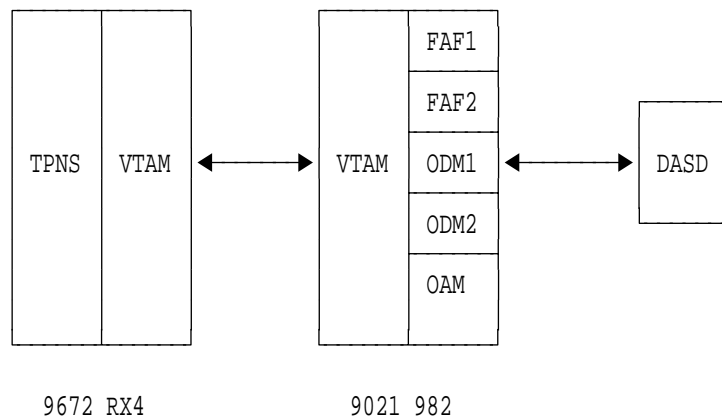


Figure 2. The Benchmark Hardware Configuration

3.1.2 DASD

All of our databases were resident on eight strings of DASD. Each string consisted of 20 volumes of 3390 spindles connected to cached 3990 controllers. See Section 3.3, "Setting Up DB2" on page 12 for a description of how we allocated our datasets across these volumes.

Note: There is no optical storage or tape subsystem involved in this benchmark.

3.2 Software

The ImagePlus components and key subsystem components used in this benchmark are:

- ImagePlus IPFAF V2.2 with Folder Work Application feature V2.2.1
- ImagePlus IODM V2.2
- CICS/ESA V4.1
- DB2 V4.1
- DFSMS/MVS V1.3.1
- LE/370 1.7
- TPNS 3.4

Additional system software can be found in Appendix D, “Software Products and Service Levels” on page 85.

3.3 Setting Up DB2

Based on our experience with customers who use ImagePlus/MVS, we had certain ideas about how we wanted to configure DB2 to provide us with the most efficient use of the DB2 subsystem. We decided, however, to start with the system as provided, so our initial setup was limited to dataset placement. The changes we made can be found in Chapter 6, “Tuning Tips” on page 29.

We had a total of 160 volumes available to us for the DB2 tables. These volumes were evenly distributed over eight strings to reduce I/O contention:

- IPFAF FWA had 16 volumes
- IPFAF API had 16 volumes
- IODM had 16 volumes
- OAM had 20 volumes (2 per OAM storage group)
- DB2 had 4 logs of 800 cylinder each

To control the placements of the DB2 datasets, we had each DB2 object (table, index, and so on) reside in its own DB2 storage group. We determined which volumes each STOGROUP was able to use. We ensured that, for any table, any indexes were on a different volume from the data. We ensured as much as possible that any active table was on a pack of its own.

The buffer pool settings were initially:

BP0	2000 4K blocks
BP1	2000 4K blocks
BP32K	500 4K blocks

Generally, we used BP0 for our data and BP1 for the indices. BP32K was used for the OAM 32K tables.

We made some changes to ZPARMS to reflect the situation at a known ImagePlus/MVS site. We did this so that we did not experience problems with these parameters later. The ZPARMS settings we used are in Section A.1, “ZPARMS” on page 33.

The RCT that we started our tests with are in Section A.2, “Original RCT for IODM” on page 37. Discussion of changes that we made to the RCT are in Chapter 6, “Tuning Tips” on page 29.

3.4 Setting Up CICS

The CICS system we used had previously been used to support only a small number of users. We needed to change this to support up to 1000 users.

For the LU2 sessions, we simply changed CICS to support auto-install so that any number of 3270s could be active. For the LU6.2 sessions, we defined all 1000 users that might be logging on. The reason for this is that the IODM to IWPM/2 communications is initiated by the host. If there is no CICS definition for the workstation, we cannot perform the OINT transaction. We have experience with customers who do use auto-install for their LU6.2 communication. However, so that the connection gets defined to CICS, it has been necessary to write an OS/2 program that uses APPC to initiate the conversation. Once the workstation has been auto-installed, IODM can initialize the workstation.

We found that defining this large number of workstations caused CICS to take some time to initialize, and it also used a large amount of storage. Since we had two IODMs, we could allocate 500 workstation definitions to each IODM (even LU numbers in one region and odd numbers in the other). This alleviated our initialization delay.

We did not spend much time configuring CICS. Some benefit might be gained by tuning CICS to better support ImagePlus/MVS. The SITs that we used for our CICS regions can be found in Appendix B, "CICS SIT Tables and SYSIN Definitions for IODM and IPFAF" on page 63.

3.5 Setting Up IPFAF and IODM

More than one large customer has had to split their IPFAF or IODM (or both) regions into multiple CICS regions to overcome restrictions on how much processor capacity one CICS region can use. We questioned whether we needed to split IPFAF and IODM, and came to the conclusion that IPFAF used more than one processor in our complex, and that IODM needed close to one processor. We, therefore, decided to use two IPFAF regions and two IODM regions, both accessing exactly the same tables.

3.5.1 Splitting IPFAF and IODM

To run two IPFAF systems and two IODM systems all accessing the same DB2 tables, we needed to update the IPFAF API and the IODM configuration tables. FWA is unaffected since all communication with IODM is performed by the API.

For the API, the affected table is the symbolic name table (EYPTSYMB). This table contains a list of all the IODMs that the IPFAF region can talk to, plus cross references between the logical name (symbolic name) and the true CICS ID for the link. We therefore added a new IODM to this table. For the format of the table, look at the *IPFAF API V2R2 System Programmer's Guide, SC31-7524*.

IODM contains three tables that are affected by this change:

- Front-end Application System Configuration table (IDFSIDTB), in which we needed to tell IODM that more than one IPFAF might be talking to it.
- System Configuration table (IDOSIDTB) that describes to each IODM its own characteristics.

- System Parameter table (IDPARMTB) that tells IODM what parameters to use.

We added the new IPFAF to IDFSIDTB, the new IODM to IDOSIDTB, and the new IODM to IDPARMTB. For the format of these tables, see the *ImagePlus IODM V2R2 System Programmer's Guide, SC31-7535*.

3.5.2 Some Key Tables

Here is a partial listing of the key tables we used in this configuration.

```
-- FRONT-END APPLICATION SYSTEM CONFIGURATION TABLE
  SELECT * FROM EKADB2.IDFSIDTB;
-----+-----+-----+-----+-----+-----+-----+-----+
SSYSID  SCICSID  FMSSTAT  FMSTYPE  ODMIFLV  MODDATE   MODTIME  MODOPRI
-----+-----+-----+-----+-----+-----+-----+-----+
FAFD    FAFD     S         C         0220     1995-11-27 13.08.47  IBMUSRE
FAF2    FAF2     S         C         0220     1997-01-28 11.05.15  IMAGE4

-- SYSTEM CONFIGURATION TABLE
  SELECT * FROM EKADB2.IDOSIDTB;
-----+-----+-----+-----+-----+-----+-----+-----+
TSYSID  TCICSID  QUESTAT  SENQSTAT  MAXQTASK  ATIQSIZE  OMDSFITLV  ODMT
-----+-----+-----+-----+-----+-----+-----+-----+
ODMD    ODM2     2         2           5           0 0220       H
ODM2    ODM2     2         2           5           0 0220       H

-- SYSTEM PARAMETER TABLE
  SELECT * FROM EKADB2.IDPARMTB;
-----+-----+-----+-----+-----+-----+-----+-----+
OSYSID  OCICSID  STRELOWF  GUESLOWF  SECEXITF  CHKDITG  CBUFSIZE  SCRBLA
-----+-----+-----+-----+-----+-----+-----+-----+
ODMD    ODM2     N          N          N          N          31581
ODM2    ODM2     N          N          N          N          31581

-- SYMBOLIC NAME TABLE
  SELECT * FROM EYPDB2.EYPTSYMB;
-----+-----+-----+-----+-----+-----+-----+-----+
SYMBCNT  SYMBVAL
-----+-----+-----+-----+-----+-----+-----+-----+
          2  ODM2ODM2    1ODM2ODM2    1
```

3.5.3 LU 6.2 Connections and Definitions

The LU 6.2 sessions and connections are defined to each CICS region. Here are the definitions we used in this benchmark.

```
*
* IODM region LU 6.2 definitions
*
*****
* Define connection and session to FAFD region
*****
*
DEFINE CONN(FAFD)          GROUP(ODMLU62)
      NETNAME(CICSAFAD)    ACCESSMETHOD(VTAM)
      PROTOCOL(APPC)       SINGLESESS(NO)
      DATASTREAM(USER)    RECORDFORMAT(U)
      AUTOCONNECT(YES)     INSERVICE(YES)
```

```

ATTACHSEC(LOCAL)
*
DEFINE SESS(FAFD)          GROUP(ODMLU62)
CONNECTION(FAFD)         MODENAME(ISCLU62)
PROTOCOL(APPC)           MAXIMUM(8,4)
SENDSIZE(4096)           RECEIVESIZE(4096)
OPERSECURITY(1)
SESSPRIORITY(0)
AUTOCONNECT(YES)        BUILDCHAIN(YES)
USERAREALEN(0)          IOAREALEN(0)
RELREQ(NO)              DISCREQ(NO)
*
*
*****
* Define connection and session to FAF2 region *
*****
*
DEFINE CONN(FAF2)         GROUP(ODMLU62)
NETNAME(CICSFAF2)       ACCESSMETHOD(VTAM)
PROTOCOL(APPC)          SINGLESESS(NO)
DATASTREAM(USER)        RECORDFORMAT(U)
AUTOCONNECT(YES)        INSERVICE(YES)
ATTACHSEC(LOCAL)
*
DEFINE SESS(FAF2)         GROUP(ODMLU62)
CONNECTION(FAF2)        MODENAME(ISCLU62)
PROTOCOL(APPC)           MAXIMUM(8,4)
SENDSIZE(4096)           RECEIVESIZE(4096)
OPERSECURITY(1)
SESSPRIORITY(0)
AUTOCONNECT(YES)        BUILDCHAIN(YES)
USERAREALEN(0)          IOAREALEN(0)
RELREQ(NO)              DISCREQ(NO)
*
* IPFAF region LU 6.2 definitions
*
*****
* Define connection and session to ODM2 region *
*****
*
DEFINE CONN(ODMD)        GROUP(FAFLU62)
NETNAME(CICSODMD)       ACCESSMETHOD(VTAM)
PROTOCOL(APPC)          SINGLESESS(NO)
DATASTREAM(USER)        RECORDFORMAT(U)
AUTOCONNECT(Y)          INSERVICE(YES)
ATTACHSEC(LOCAL)
*
DEFINE SESS(ODMD)        GROUP(FAFLU62)
CONNECTION(ODMD)        MODENAME(ISCLU62)
PROTOCOL(APPC)           MAXIMUM(8,4)
SENDSIZE(4096)           RECEIVESIZE(4096)
SESSPRIORITY(0)
AUTOCONNECT(YES)        BUILDCHAIN(YES)
USERAREALEN(0)          IOAREALEN(0)
RELREQ(NO)              DISCREQ(NO)
*
*****
* Define connection and session to ODM2 region *
*****

```

```

*
DEFINE CONN(ODM2)          GROUP(FAFLU62)
      NETNAME(CICSODM2)    ACCESSMETHOD(VTAM)
      PROTOCOL(APPC)       SINGLESESS(NO)
      DATASTREAM(USER)    RECORDFORMAT(U)
      AUTOCONNECT(Y)       INSERVICE(YES)
      ATTACHSEC(LOCAL)

*
DEFINE SESS(ODM2)          GROUP(FAFLU62)
      CONNECTION(ODM2)     MODENAME(ISCLU62)
      PROTOCOL(APPC)       MAXIMUM(8,4)
      SENDSIZE(4096)       RECEIVESIZE(4096)
      SESSPRIORITY(0)
      AUTOCONNECT(YES)     BUILDCHAIN(YES)
      USERAREALEN(0)       IOAREALEN(0)
      RELREQ(NO)           DISCREQ(NO)

```

3.5.4 Considerations When Displaying or Printing

When IPFAF displays a document, there are two possible IODMs. One IODM owns the workstation, whose responsibility it is to send the image to IWPM/2. The other IODM owns the object and retrieves it from OAM.

In a display operation, IPFAF sends the display request to the workstation-owning IODM, which looks at which IODM owns the object. If it is the owner, it simply retrieves the object and sends it to the workstation. However, if the object is owned by another IODM, the workstation-owning IODM communicates with the object-owning IODM, asking it to retrieve the object. When it receives the object, it sends it to the workstation.

We wanted to avoid the communication between the IODMs for three reasons. First, the transaction figures that we gathered were inflated by a number of cross-IODM transactions. Second, the resource utilization included extra transactions that might not actually occur in a customer system. Third, the process of setting up the cross-region communication while still using one database causes restrictions on naming conventions of your symbolic names.

We therefore had to decide how to allow the use of two IODMs. Our decision was this: since we are only using one OAM, whichever IODM IPFAF thinks owns the object does so. This is because, in fact, both IODMs can access the object. We, therefore, decided to use the Disperse Display exit (EKCCDDTE) to change the data stream passed to IODM to identify the object-owning IODM as whichever IODM the transaction happens to be running in. When only one OAM is being used, this is a perfectly satisfactory solution. The code we used for EKCCDDTE is in Appendix C, "IODM Exit: EKCCDDTE" on page 69.

Chapter 4. Information from the Benchmarks

This chapter contains the results of our benchmarks. We describe three runs that capture the information we were looking for. Run 1 exemplified the current environment of a known large customer. Run 2 was to see the result of increasing the number of indexing and scanning operations. Run 3 was to test the effect of increasing the number of queues per user.

See Chapter 5, "How to Use This Information" on page 23 for a more detailed description of the contents of the scripts for each run.

4.1 Conclusions

Our major conclusion from these runs was that DB2 was never the limiting factor in the system. We ran out of processor resource before DB2 seemed to be under stress. With a well-tuned DB2 system on the processor we were running on, the indication was that significantly more than the 100 ImagePlus/MVS transactions per second we achieved were within the capabilities of the system.

We also showed how sensitive the system is to the application design. A different transaction mix causes different resource utilizations. More important, the design of the system (number of queues per user) can significantly affect the processor utilization of an ImagePlus/MVS system. See Chapter 5, "How to Use This Information" on page 23 for more on interpreting the results.

Note: These results are specific to our environment and need to be used with extreme care when extrapolating to a live environment. See Section 4.5, "Caveats" on page 22 for elements of this test that may differ from your situation.

4.1.1 Response Time Figures

For each run, we gathered response time statistics using TPNS. We used these response times to ensure that we were providing a reasonable service level agreement in a live production environment.

The response times as measured by TPNS include all LU2 and LU6.2 transactions.

The response times may not reflect true user responses since, although TPNS was running on a different machine, it was a machine in the same sysplex so we experienced no network delays. It is, however, a reasonable guide to response times.

The LU6.2 responses were also not subject to network delays. There is also a secondary concern in that it measures only the response involved in sending or receiving the image. For scanning, the scan time is not involved. It is just the time from initiating a store document to the time the successfully stored response is received at the workstation. For a display, no processing time is allowed for in the response time. For a small image, the perceived response time includes the overhead of decompressing the image and displaying it at the workstation. For a large document, perceived user response may actually be lower than those quoted since IWPM/2 displays the first page when it receives it.

This means that the first page may actually be displayed before the entire object is received, giving user perceived response actually faster than TPNS measures.

4.1.2 CPU Utilization

The figures we quote from RMF give the percentage of CPU used by the various different components of ImagePlus/MVS. The processor we were running on was an ES/9000 9021 model 982. This complex has a total of eight processors. The percentage figures given by RMF are a percentage of a single processor, so the numbers quoted come to more than 100% CPU utilization.

To interpret these numbers and relate them to your processor, you need to use the *Large System Performance Reference, SC28-1187*, which quotes the internal throughput ratios for IBM and other manufacturers' systems.

Note that the CPU figures we quote are simply for the subsystems that ImagePlus uses. In any MVS system, there is an overhead of processor that is used by systems functions. This overhead is termed a "Capture Ratio." To understand how much CPU the entire system would use, you need to factor in your customer's capture ratio. In our case the capture ratio varied for the different runs, but was generally 75%. It is worth checking with the capacity planners at your customer before using a figure of 75%.

4.2 Run 1

The purpose of this run was to drive the system hard in a work delivery intensive environment. The profile we set out to simulate is a system that is used primarily for delivering work to users who process that work quickly and then drop the work and ask for more work. We, therefore, set our TPNS parameters to have more users in the getwork/windup script than in the other two.

The parameters we set in TPNS were:

- 75% of our users were getwork/windup users.
- 20% of our users were Indexers/Scanners.
- 5% of the users were supervisors checking queues.
- 20% of the time that users did a getwork, they looked at the history of the document.
- 40% of the time that users did a getwork, they listed the contents for that folder.
- 90% of the time the getwork users did a windup, they dropped the document. They rerouted the other 10%.
- The time waited after the end of the script before restarting was:
 - Getwork users - 30 Seconds
 - Indexers - 90 seconds
 - Supervisors - 10 seconds

The tables below show the number of CICS transactions that we ran during a half hour period and the CPU utilization that those transactions resulted in. For more information on how these transaction relate to actual end user interactions, please see Chapter 5, "How to Use This Information" on page 23.

<i>Table 1. Transaction Profile</i>		
Transaction ID	Number Performed during Half-Hour Period	Number Performed per Second
\$AO0	86	0.05
\$AR0	9408	5.23
\$GT0	4703	2.61
\$GW0	44 633	24.80
\$LC0	10 558	5.87
\$MN0	32 774	18.21
\$QD0	2606	1.45
\$WU0	42 145	23.41
DISD	18 429	10.24
ER00	4747	2.64
STAH	4746	2.64
STOR	4747	2.64
TMPC	4742	2.63
Totals	184 324	102.40

The objective of this run was to see how many transactions we were able to drive through an ImagePlus system. The only part of the system that constrained us was the amount of available CPU. We thought it significant that over 100 CICS transaction could be run through the system. We observed no bottlenecks during this run. Had we had more CPU available, we feel we could have run significantly more transactions.

<i>Table 2. CPU Utilization by Component</i>	
Component	%CPU Time
IPFAF 1	162.3
IPFAF 2	158.0
IODM 1	46.2
IODM 2	43.9
DB2	40.7
Total	451.10

The figures above show only the components used by ImagePlus. You must add your customer's capture ratio to give the true amount of CPU required by the entire system including MVS overhead. As mentioned above, our capture ratio was approximately 75% (that is we only captured 75% of the overall system utilization; the rest is system overhead).

Note: The percentage figures given by RMF are a percentage of a single processor, so the numbers quoted come to more than 100% CPU utilization.

<i>Table 3. Response Times Measured by TPNS</i>		
Average Response Time	Maximum Response Time	90th Percentile
0.11	6.00	0.30

We used the response time statistics from TPNS to show that we were not actually hitting any bottlenecks. With a 90th percentile figure of three tenths (.3) of a second, we feel that with a correctly tuned system you should be able to guarantee subsecond responses from the IPFAF system. Note that the actual time to decompress and view an image may be higher than this.

4.3 Run 2

For this run, we decided to have a higher percentage of indexers. This gave us a higher number of inserts into the object, event, and work detail tables at the same time that the getwork script is performing updates and deletes.

To increase the number of indexers, we reduced the time that indexers waited after finishing the script before restarting. We reduced the number from 90 seconds to 30 seconds.

Table 4. Transaction Profile

Transaction ID	Number Performed during Half-Hour Period	Number Performed per Second
\$AO0	173	0.10
\$AR0	15 725	8.74
\$GT0	7840	4.36
\$GW0	37 023	20.57
\$LC0	8783	4.88
\$MN0	33 551	18.64
\$QD0	3608	2.00
\$WU0	34 728	19.29
DISD	15 248	8.47
ER00	7927	4.40
STAH	7951	4.42
STOR	7927	4.40
TMPC	7929	4.41
Totals	188 413	104.67

Table 5. CPU Utilization by Component

Component	%CPU Time
IPFAF 1	147.10
IPFAF 2	152.60
IODM 1	44.90
IODM 2	44.40
DB2	38.90
Total	427.90

Table 6. Response Times Measured by TPNS

Average Response Time	Maximum Response Time	90th Percentile
0.10	3.50	0.30

The figures that we measured here confirmed our view that the Getwork transaction is the most heavy user of CPU within the IPFAF system. However, if your users mainly use the ImagePlus system as an electronic file-cabinet rather than a work delivery system then you will find that the CPU requirements are lower than for a heavily workflow-biased system.

4.4 Run 3

We noticed during earlier runs that the resource utilization of the system seemed to be sensitive to the number of queues each user has. To test this theory, we increased the number of queues each user is assigned to 55 (from 24 in Run 1). All other parameters were left the same.

Table 7. Transaction Profile

Transaction ID	Number Performed during Half-Hour Period	Number Performed per Second
\$A00	82	0.05
\$AR0	7381	4.10
\$GT0	3687	2.05
\$GW0	38415	21.34
\$LC0	8952	4.97
\$MN0	28454	15.81
\$QD0	2464	1.37
\$WU0	35902	19.95
DISD	15631	8.68
ER00	3729	2.07
STAH	3729	2.07
STOR	3729	2.07
TMPC	3736	2.08
Totals	155891	86.61

To our surprise, the number of transactions per section was lower, yet the CPU utilization was higher (see Table 8). As you read, from the first two runs we came to the conclusion that Getwork is the transaction that requires the most CPU. This is because of the number of DB2 accesses it performs. In this run, we increased the number of queues that the Getwork transaction needs to search to define what piece of work is the highest priority for this user. Thus we increased the amount of DB2 accesses and therefore the amount of CPU required. See section 6.2, "Application Design" on page 30 for additional details.

Table 8 (Page 1 of 2). CPU Utilization by Component

Component	%CPU Time
IPFAF 1	213.1
IPFAF 2	205.1
IODM 1	38.8
Total	530.20

<i>Table 8 (Page 2 of 2). CPU Utilization by Component</i>	
Component	%CPU Time
IODM 2	36.4
DB2	36.4
Total	530.20

<i>Table 9. Response Times Measured by TPNS (in Seconds)</i>		
Average Response Time	Maximum Response Time	90th Percentile
0.18	4.70	0.50

Our conclusions from these runs are described in the next chapter.

4.5 Caveats

There are specific elements that make this different from the workload that an ImagePlus/MVS customer may experience. These include:

- We were working with small tables. The work detail table contained 50 000 rows, which may be similar to a live situation. However, the object table also contained only 50 000 records, which is unusually low.

This has many effects on the way the system performs. With a large number of transactions accessing small tables, we can expect more contention as multiple transactions try to access the same part of the tables. On the other hand, scanning a small object table may take less resource than scanning a large table, and it may be that in our situation, all entries were in buffer pools, so we avoided I/O that is necessary on a larger table.

- We did not use optical or tape storage.

We often find that the limitation on throughput in an ImagePlus/MVS system is how many mounts per hour an optical storage or tape subsystem can handle. We recommend that any optical or tape subsystem be configured to be able to easily handle the number of requests that it is likely to encounter.

In our experience, it is difficult to estimate the number of mounts per hour that are requested from secondary or tertiary storage. An analysis of the number of accesses to filed documents does not reflect the actual requests to secondary storage. Once users have online access to this data, they continue to make requests for documents that have long been archived. We recommend that you expect significantly more requests than estimated from analyzing the current processes. You should also ensure that the solution you implement is flexible so that you can change the system to service these requests.

- Our workload profile was specific. We made some runs here to show how a change in workload can significantly affect the resource utilization of the system. Consider this when making any estimates based on these figures.

Chapter 5. How to Use This Information

This book documents our results from running certain transaction profiles. A live situation undoubtedly produces different profiles. This chapter describes the transaction flows for the TPNS scripts that we ran and points out some differences in the resource utilization for the different scripts. This should give some assistance in relating these figures to a real life situation. Note that capacity planning for any application is a process that needs to be undertaken with great care. Capacity is easy to underestimate, with resulting problems later on.

We avoid using "number of users" as a measurement, since it is not necessarily an accurate way to determine the likely resource utilization of a given application. The figures from Run 3 show how an apparently innocent change (the number of queues a user is assigned to) can have a dramatic effect on CPU utilization, independent of the number of users logged on. In the field, we have seen that predicting capacity based on the number of users can cause a significant underestimation of resources required.

We therefore recommend that you determine the workload mix that your application will generate. Although this cannot guarantee perfect planning, it provides a better estimate than working with the number of users.

To see what transactions the end user interactions actually generate, please see Chapter 2, "Description of Our Scenarios" on page 5.

5.1 Indexing and Scanning

A single TPNS script calls a number of actual CICS transactions. The following table shows which CICS transactions were performed for each indexing and scanning script we ran.

Script Entry	IPFAF CICS Transactions	IODM CICS Transactions	Description
ADOR from Main Menu	\$MNO		Reads the window and determines which transaction needs to be performed. Calls Customer Data Exit to determine validity of folderid. Determines routing info. Puts up Add and Route window.
Press Enter on the Add and Route window.	\$AR0		Reads the Add and Route window. Rechecks routing, validates fields such as date as well as validating the RLOB/TRANTYPE combination. performs DEFOBJ API that starts IODM transaction and updates Scan Pending table with information about the TEMPID.
		STAH	Updates the Store Authorization table to inform IODM of pending scan.

Script Entry	IPFAF CICS Transactions	IODM CICS Transactions	Description
Start scanning document at workstation.		TMPC	Checks the Store Authorization table to determine this is a valid TEMPID, passes relevant information to the workstation.
Store the document.		STOR	Stores the document in OAM, deletes entry from the Store Authorization table, starts the register transaction (ER00) in IPFAF.
	ER00		Updates the Object and Event tables with the information from Scan Pending table. Starts the \$GT0 transaction to inform FWA that the document has been scanned.
	\$GT0		Inserts a row in the Work Detail table for the recently scanned document. Updates the event table with the relevant information.

From this table, we can see that one TPNS script results in seven actual CICS transactions. However, if we need to know how many documents we have actually stored, then the count of IODM STOR transactions will give us that figure. Using the figures from Run 1 (Table 1 on page 19) we can see from the number of STOR transactions that we have actually stored 4,747 documents in the half hour period of the run.

5.2 Getwork Windup Script

These are the figures for a getwork windup loop. See section 5.2, "Getwork Windup Script" for an explanation of a getwork.

Script Entry	IPFAF CICS Transactions	IODM CICS Transactions	Description
Option 1 from Main Menu	\$MNO		Reads the window and determines which transaction needs to be performed.
	\$GW0		Determine the highest priority document for this user.
		DISP	Display the document for the user.
Press PF10 on the Getwork List window.	\$GW0		Reads the window to determine what to do. Calls \$WU0 to perform Windup.
	\$WU0		Writes the Windup window to ask the user what to do.
Choose option 7 from the windup window.	\$WU0		Performs the drop.
Press PF12 to return to main menu.	\$WU0		Display the main menu window.

One getwork windup window also results in seven actual CICS transactions. If we again tie this to the statistics we gathered from run 1 (Table 1 on page 19) we can see that each getwork script performed by a user actually results in two

\$GW0 trasactions. Since we performed 44,633 \$GW0 transactions in our half our period the users actually performed 22,316 Getwork scripts.

Please note that in a live system this ratio may change. For example, if a user performs a history view from the Getwork List screen then an extra three \$GW0 transactions will be performed. If a user reroutes a document, then extra \$WU0 transactions will be performed. Before using the above figures in a live situation, ensure that the users perform Getwork Windup as described in our script (see Chapter 2, "Description of Our Scenarios" on page 5 for details).

5.3 Workflow Functions Script

With an image system that provides a work delivery function such as ImagePlus/MVS, the role of the supervisor changes dramatically. When large quantities of paper arrive in a office, the supervisors will normally spend much time sorting the paper and handing out work to the processing clerks. With an ImagePlus/MVS system, this work has already been done by the indexing process.

The focus of the supervisor's role therefore changes to ensure that outstanding work is being performed in a timely fashion. To do this, the supervisor can use the workflow screens of IPFAF to inspect the queues of work that is outstanding. This is what this script simulates.

Table 12. CICS Transactions for Each Workflow Functions Script

Script Entry	IPFAF CICS Transactions	IODM CICS Transactions	Description
Option 7 from Main Menu	\$MNO		Reads the window, determines which function is to be performed, writes the Workflow Functions window.
Choose option 6	\$MNO		Reads the window, displays the Queue Information Selection window.
Enter a unit code and press Enter.	\$QD0		Finds the work in that queue and displays the Queue Information window.
Press PF3 to return to main menu.	\$QD0		Display the main menu window.

These figures show us that for each queue display there are actually 2 \$QD0 transactions performed. Referring back to section Table 1 on page 19, since there were 2,464 \$QD0 transactions, the script was actually performed 1,232 times.

5.4 An Approach to Capacity Planning

As we stated earlier in the chapter, we do not recommend using the number of users as a guide to capacity planning. In this section, we describe what data we feel is important to gather to produce an initial capacity plan and how to use that data.

As with all capacity planning, it is necessary to monitor the live system once it is running to ensure that assumptions that were made during the original estimates

are true, and that factors unique to your implementation are not affecting the amount of resource required.

5.4.1 Data to Gather

When creating your plan, we suggest you start with data from a high level design of your image system, such as:

- Number of documents to be stored into the system
- Percentage of these documents that are routed in the workflow
- Number of reroutes each document will experience on average
- Number of queues
- Number of queues each user has assigned on average
- When is indexing and scanning performed

From this data, you can begin your capacity plan.

Hint: Our experience with customers, as well as with this benchmark, shows that the most resource intensive transaction is the Getwork. One approach we have successfully used in the past is to estimate the number of Getworks that are performed in the system. See section 5.2, “Getwork Windup Script” on page 24 for an explanation of a Getwork. This figure gives you a good start on an initial capacity plan.

5.4.2 Applying This Data

Let's look at an example of how to apply the data you have gathered. We compare a sample system to our benchmark. Consider a proposed system that has the following characteristics:

- 50 000 documents are stored per day.
- 60% of all documents are placed in routing, the remaining documents are stored and not worked.
- 66% of all documents are worked once, and then dropped. The rest of the documents are worked twice on average.
- On average, each user has 24 queues to work.
- Most indexing and scanning is performed outside the working day.

From the preceding figures, we can see that stored documents cause 30 000 business getworks per day. Ten thousand of these documents are rerouted once, giving another 10 000 business getworks per day. Our total number of getworks per day is, therefore, 40 000 per day.

If the working day in this environment is eight hours, we expect 5000 business getworks per hour. We need to decide if there is a peak in the workload (the processor resource we need must cover our peak hour). Let us assume that the peak hour is double the normal workload. The peak hour, therefore, generates 10 000 business getworks.

From the figures from Table 11 on page 24, we can see that each business transaction causes two CICS \$GW0 transactions. We can, therefore, expect 20 000 CICS \$GW0 transaction during peak hour. From Table 1 on page 19, we can see that we performed 44 633 \$GW0 transactions in a half-hour period, (that is, 89 000 \$GW0 transactions in an hour).

Our proposed system of 20000 transactions per hour is approximately one quarter of the number of transactions we performed in the benchmark. Therefore, we estimate that the Getwork transactions place a load of 110% of one processor on a 9021 model 982. (Remember that there are multiple processors in a sysplex). If we apply a capture ratio of 75% to this figure then the overall system load will be 147% of one processor on a 9021 model 982. For a different system, you can use the *Large System Performance Reference, SC28-1187* to determine the effect on the processor you are running on.

During our first run, there were other transactions being performed (Add and Route, List Folder Contents, and Supervisor Functions). Although our scenario was biased toward an intensive Getwork situation, there are a number of these other transactions that are generated by the workload. Unless your environment is significantly different from ours (for example, a store and retrieve environment where no Getworks are performed at all), the figures we have generated should be close enough for a working capacity plan.

5.4.3 Changes from Previous Scenario

If your scenario is significantly different from ours, you may need to apply some factors to modify your estimates.

5.4.3.1 More Indexing and Scanning

Consider an environment where scanning is performed during the day. In Run 2, we increased the number of Add and Route Scripts we performed. This gives us data to estimate how much processor resource the add and route script uses.

In Run 1, we performed 44633 \$GW0 transactions in a half-hour period. In Run 2, we performed 37023 \$GW0 transactions. The CPU utilization for Run 1 was 451%, and in Run 2 was 423%. We can normalize this data to see the result if we had performed the same number of Getworks in Run 2 as we did in Run 1. 423 divided by 37023 and multiplied by 44633 gives 510% CPU utilization. If we used the same workload profile, we would expect 451% utilization. The extra 60% utilization is, therefore, caused by the extra Add and Route transactions that we performed. In run 1, we performed 9408 \$AR0 transactions, while in Run 2 we performed 15725. Since we have normalized the CPU utilization figures, we must do the same for the number of \$AR0 transactions. So if we had been running at 510% CPU utilization, we could, therefore, perform $15725 / 37023 * 44633 = 1895$ \$AR0 transactions.

From the preceding figures, we can say that the extra 60% CPU utilization was caused by 18957 - 9408 \$AR0 transactions (that is, 9549 \$AR0 transactions). From Table 10 on page 23, the number of \$AR0 transactions for each Add and Route script is actually a one-to-one ratio. This means that 9549 Add and Routes in a half-hour period (4775 Add and Routes per hour) requires 60% of one processor in a 9021-982 complex.

Let's apply these figures to our proposed system.

We are indexing and scanning 50000 documents a day. If this is performed over our 8-hour day, on average we index and store 6250 documents per hour. Allowing for a peak hour of double the average rate, we find that we index and scan 12500 documents per hour. Since Run 1 already includes some Indexing and Scanning, we can say that 4000 (one quarter of the actual measured transactions in our run multiplied by two for our peak hour) is already included in the figure of 110% CPU utilization. We need to calculate the processor

resource required for the extra Indexing and Scanning operations, which is $12500 - 4000 = 8500$. We must multiply the 60% by $8500 / 4775 = 1.7$ which gives a figure of 107% CPU utilization.

In the scenario where we are scanning during the day and our peak hour is twice the average, we require an additional 107% of one processor on a 9021 model 982. Applying our capture ratio of 75% gives us an overall figure of 142.6%.

5.4.3.2 Different Number of Queues Per User

Run 3 shows us that if we increase the number of queues per user, the extra work that the Getwork transaction has to perform to determine the highest priority piece of work for a user increases the amount of processor resource required. Let's do some calculations to see the affect of this.

Consider the number of transactions per second we performed in runs 1 and 3. Run 1 performed 102.4 transactions per second and required 451% processor utilization (that is, $451 / 102.4 = 4.4\%$ processor utilization per transaction per second).

Run 3 performed only 86.6 transactions per second but required 530% processor utilization. These figures give $530 / 86.6 = 6.1\%$ processor utilization per transaction per second.

Let's apply this information to our sample system. If we change the number of queues per user from 24 queues to 55, we expect our processor utilization to increase from 110%. To calculate what the increase is, take the ratio of processor resource per transaction per second for Run 3 versus Run 1 ($6.1 / 4.4 = 1.4$). We must multiply our 110% utilization by 1.4 to give us the resource we need. This gives a total utilization of 154% if we increase the number of queues from 24 to 55. Applying our capture ratio of 75% gives an overall processor requirement of 205.3% processor utilization.

If we decrease the number of queues per user, we expect the processor resource to decrease in proportion.

5.5 Caveats

The calculations we performed previously are based on the small number of runs we documented in Chapter 4, "Information from the Benchmarks" on page 17. We recommend that figures are monitored carefully on a live system during actual rollout to calculate the amount of processor resource you need in your environment.

The capture ration we are using was based on our experience. This varied across our runs, so you should take care to determine the correct figure to use in your environment.

Chapter 6. Tuning Tips

This chapter describes changes we made to the system that we either found necessary or thought might improve the resource utilization. Wherever possible, we quantify the benefits.

6.1 DB2 Changes

Here are some changes we made to the DB2 environment.

6.1.1 Single Threading Background Transactions

We found that single threading background transactions were not necessary, despite our previous experience with customers.

Our experience in live situations is that when a large number of transactions are accessing the active tables, timeouts can occur (particularly with indexes). This is a problem if the transaction that times out is one of the background register transactions (ER00 and \$GT0). See Chapter 5, "How to Use This Information" on page 23 for more information on these transactions. The result is that an index scan operation has apparently worked successfully, but the document does not appear in the object table or work detail table.

Customers have created DB2 reports to warn them of this situation. They also try to eliminate this problem by single threading ER00. The sample RCT entry for ER00 also suggests this. This is the sample RCT entry from the *IPFAF API V2R2 System Programmer's Guide, SC31-7524*.

```
ER00      DSNCRCT TYPE=ENTRY,                                X
          PLAN=EYPODF00,                                  PLAN NAME          X
          THRDM=1,                                       MAX NBR IN A THREAD SET X
          THRDA=1,                                       NBR OF ACTIVE THREAD X
          THRDS=0,                                       NBR ACT THRDS AT   X
          TWAIT=YES,                                     TRAN TO WAIT FOR THREAD X
          AUTH=(EYPCICS),                                AUTHORIZATION GROUP ID X
          TXID=ER00                                     TRANSACTION ID
```

Specifying THRDM=1 means that this transaction can use only one thread. The TWAIT=YES parameter means that the transaction waits for an available thread rather than going to the pool.

We originally started our runs with this RCT entry for ER00. See Section A.3, "Original RCT for IPFAF" on page 39. However, we found that we were limited in the number of scans per minute we could perform. If we tried to perform 600 scans/minute, this was 300 scans/minute for each of the IPFAF systems. To achieve this rate, we needed to perform 300 ER00 transaction per minute, or five ER00 transactions per second. Although five transactions per second is not high, when they are single threaded, each ER00 must have an elapsed time of .2 of a second for us to maintain the transaction rate. We wanted to scan more documents per minute than these limitations gave us, so we chose not to single thread ER00.

We decided to increase the number of threads available to ER00, and also to specify TWAIT=POOL so that the transaction went to the pool of threads if none were available. This is the RCT entry we used for the rest of our runs.

ER00	DSNCRCT TYPE=ENTRY,		X
	PLAN=TPAODF00,	PLAN NAME	X
	THRDM=20,	MAX NBR IN A THREAD SET	X
	THRDA=20,	NBR OF ACTIVE THREAD	X
	THRDS=10,	NBR ACT THRDS AT	X
	TWAIT=POOL,	TRAN GO TO POOL FOR THRD	X
	AUTH=(ENTCICS),	AUTHORIZATION GROUP ID	X
	TXID=ER00	TRANSACTION ID	

Despite allowing several concurrent ER00 transactions to run, we never hit problems with ER00 timeouts.

The FWA "GOODTRAN" transaction \$GT0 has a similar recommendation in the manual. We chose to allow more than one thread for this transaction.

6.1.2 Use of Packages

Much of the resource used by DB2 is used when opening and closing threads. We wanted to avoid this as much as possible, and to maximize the reuse of threads. To do this, we bound each DBRM into its own package, and then bound all packages into one plan so we only had one plan for our CICS regions. The bind jobs for creating this plan can be found in A.6, "DB2 Package and Plan for IODM and IPFAF" on page 58.

We also changed the RCT so that we only had one entry for all of our transactions to use the same threads rather than have threads of their own. The RCT we used can be found in A.4, "Final RCT for IODM" on page 49. We also specified the PURGEC parameter RELEAS DEALLOCATE rather than RELEASE COMMIT to keep threads open longer.

To decrease the amount of resource used for authorization checking, we granted public authorization to our plan; this also allowed us to specify CACHESIZE=0 since no cache is required for authorization checking.

To prevent lock contention, we changed the IODM and IPFAF tablespaces from LOCKSIZE=ANY to LOCKSIZE=PAGE and change the plans from REPEATABLE READ to CURSOR STABILITY.

The overall effect of this was a significant reduction in CPU utilization and a decrease in lock contention. During the periods of the runs we documented in Chapter 4, "Information from the Benchmarks" on page 17, we experienced no timeouts or deadlocks throughout any of the runs.

6.2 Application Design

The runs we performed showed us the importance of application design. Although we were unable to test the effects of all possible combinations, the most obvious effect was the change in number of queues per user. In Run 3, we ran exactly the same TPNS profile as Run 1, but we were only able to run at approximately 86 transactions/second. In Run 1, we ran at over 102 transactions

per second. While performing more transactions, we used less processor resource.

The only change between Run 1 and Run 3 was to increase the number of queues each user has assigned from 24 queues to 55 queues.

If we look at the way that IPFAF performs a getwork, the reason for this becomes clear. When a user performs a getwork, the first thing IPFAF does is to fetch the first row for that user from the User Assignment table (EYPTWEAS). It then retrieves rows from the Work Detail Table (EYPTWDET) to determine the highest priority item in that queue. When it has done this, it returns to the User Assignment table to find the next queue. If we have 24 queues, these SQL statements are performed 24 times; if we have 55, they are performed 55 times.

Our recommendation is to reassess the business design of your system. Wherever possible, keep the number of queues per user as small as possible.

There may be other aspects of design that affect the system. It is worth reviewing all of the application designing that a customer has done. Think through the effects on system resource, and test any changes before making them.

6.3 Preloading the Work Detail Table

Provided with the IPFAF API is a job to preload the Work Detail table EYPTWDET. This program is called EYPLWDET. The reason for this job is to reduce contention on one of the indexes on EYPTWDET. We found that with DB2 Version 4 the contention on indexes has been reduced considerably. We therefore found it unnecessary to run this preload program. In fact for one run we did actually run the preload job before we performed a run. Although the figures from this run are not published here, we did find that we used more processor resource, and were unable to run as many transaction after running the preload job.

Our recommendation for DB2 version 4 users would therefore be to use type 2 indexes, and to investigate the possibility of skipping the preload job on the Work Detail table.

Appendix A. DB2 ZPARMS and RCTs

A.1 ZPARMS

These were the ZPARM values that we used. The changes that we made are described in the comments at the beginning of the listing.

```
/******  
/* JOB NAME = DSNTIJUZ *  
/* * *  
/* DESCRIPTIVE NAME = INSTALLATION JOB STREAM *  
/* * *  
/* LICENSED MATERIALS - PROPERTY OF IBM *  
/* 5695-DB2 *  
/* (C) COPYRIGHT 1982, 1995 IBM CORP. ALL RIGHTS RESERVED. *  
/* * *  
/* STATUS = VERSION 4 *  
/* * *  
/* FUNCTION = DSNZPARM AND DSNHDECP UPDATES *  
/* * *  
/* PSEUDOCODE = *  
/* DSNTIZA STEP ASSEMBLE DSN6... MACROS, CREATE DSNZPARM *  
/* DSNTIZL STEP LINK EDIT DSNZPARM *  
/* DSNTLOG STEP UPDATE PASSWORDS *  
/* DSNTIZP STEP ASSEMBLE DSNHDECP DATA-ONLY LOAD MODULE *  
/* DSNTIZQ STEP LINK EDIT DSNHDECP LOAD MODULE *  
/* DSNTIMQ STEP SMP/E PROCESSING FOR DSNHDECP *  
/* * *  
/* Changes made for ImagePlus to reflect known working system *  
/* 02/26/97 ADF CHANGED NUMLKTS=1000 TO NUMLKTS=5000, *  
/* CHANGED NUMLKUS=10000 TO NUMLKUS=50000 *  
/* CHANGED SRTPOOL=876 TO SRTPOOL=4000 *  
/* CHANGED INBUFF=28 TO INBUFF=60 *  
/* CHANGED OUTBUFF=400 TO OUTBUFF=4000 *  
/* CHANGED WRTHRSH=20 TO WRTHRSH=256 *  
/* CHANGED MAXRBLK=4384 TO MAXRBLK=60000 *  
/* 03/06/97 CHANGED DLDFREQ=5 TO DLDFREQ=0 *  
/* CHANGED PRIQTY=85 TO PRIQTY=500 *  
/* CHANGED SECQTY=5 TO SECQTY=100 *  
/* CHANGED LOGLOAD=50000 TO LOGLOAD=500000 *  
/* Following changes made because move to packages put extra *  
/* load on EDMPOOL *  
/* 03/07/97 CHANGED EDMPOOL=34625 TO EDMPOOL=64625 *  
/* 03/08/97 CHANGED EDMPOOL=64625 TO EDMPOOL=86625 *  
/* CHANGED PRIQTY=500 TO PRIQTY=700 *  
/******  
//DSNTIZA EXEC PGM=ASMA90,PARM='OBJECT,NODECK,NODBCS'  
//SYSLIB DD DISP=SHR,  
// DSN=DB2410.SDSNMACS  
// DD DISP=SHR,  
// DSN=SYS1.MACLIB  
//SYSLIN DD DSN=&&LOADSET(DSNTILMC),DISP=(NEW,PASS),  
// UNIT=SYSDA,  
// SPACE=(800,(50,50,2)),DCB=(BLKSIZE=800)  
//SYSPRINT DD SYSOUT=*  
//SYSUDUMP DD SYSOUT=*
```

```

//SYSUT1 DD UNIT=SYSDA,SPACE=(800,(50,50),,,ROUND)
//SYSUT2 DD UNIT=SYSDA,SPACE=(800,(50,50),,,ROUND)
//SYSUT3 DD UNIT=SYSDA,SPACE=(800,(50,50),,,ROUND)
//SYSIN DD *
      DSN6ENV MVS=XA
      DSN6SPRM RESTART, X
              ALL, X
              ABEXP=YES, X
              ABIND=YES, X
              AUTH=YES, X
              AUTHCACH=1024, X
              BINDNV=BINDADD, X
              CATALOG=DSNDB1G, X
              CHGDC=NO, X
              DECDIV3=NO, X
              DEFIXTP=2, X
              DEFLTID=IBMUSER, X
              DSMAX=7000, X
              EDMPOOL=86625, X
              EDPROP=NO, X
              HOPAUTH=YES, X
              IRLMAUT=YES, X
              IRLMPC=BD1IRLM, X
              IRLMSID=IRD1, X
              IRLMRWT=60, X
              IRLMSWT=300, X
              MAXRBLK=60000, X
              NUMLKTS=5000, X
              NUMLKUS=50000, X
              RECALL=YES, X
              RECALLD=120, X
              RGFCOLID=DSNRGCOL, X
              RGFDBNAM=DSNRGFDB, X
              RGFDEDPL=NO, X
              RGFDEFLT=ACCEPT, X
              RGFESCP=, X
              RGFFULLQ=YES, X
              RGFINSTL=NO, X
              RGFNMORT=DSN_REGISTER_OBJT, X
              RGFNMprt=DSN_REGISTER_APPL, X
              RRLOCK=NO, X
              SEQCACH=BYPASS, X
              SITETYP=LOCALSITE, X
              SRTPOOL=4000, X
              SYSADM=ANGELA, X
              SYSADM2=IBMUSER, X
              SYSOPR1=SYSOPR, X
              SYSOPR2=IBMUSR1, X
              UTIMOUT=6
      DSN6ARVP ALCUNIT=CYL, X
              ARCWRTC=(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15), X
              ARCWTOR=NO, X
              ARCPFX1=DB2LOG.DBD1.ARCLG1, X
              ARCPFX2=DB2LOG.DBD1.ARCLG2, X
              ARCRETN=60, X
              BLKSIZE=28672, X
              CATALOG=YES, X
              COMPACT=NO, X
              PRIQTY=700, X

```

```

PROTECT=NO, X
QUIESCE=5, X
SECQTY=100, X
TSTAMP=YES, X
UNIT=3390, X
UNIT2=3390
DSN6LOGP DEALLCT=(0), X
INBUFF=60, X
MAXARCH=500, X
MAXRTU=2, X
OUTBUFF=4000, X
TWOACTV=YES, X
TWOARCH=NO, X
WRTHRS=256
DSN6SYSP AUDITST=NO, X
CONDBAT=500, X
CTHREAD=500, X
DLDFREQ=0, X
IDBACK=500, X
IDFORE=500, X
LOGLOAD=500000, X
MAXDBAT=500, X
MON=YES, X
MONSIZE=8192, X
PCLOSET=5, X
PCLOSEN=5, X
RLF=NO, X
RLFTBL=37, X
RLFERR=NOLIMIT, X
RLFAUTH=SYSIBM, X
ROUTCDE=(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15), X
SMFACCT=(1), X
SMFSTAT=YES, X
STATIME=15, X
STORMXAB=0, X
STORPROC=, X
STORTIME=180, X
TRACSTR=NO, X
TRACTBL=16
DSN6FAC DDF=NO, X
CMTSTAT=ACTIVE, X
IDTHTOIN=0, X
RESYNC=2, X
RLFERRD=NOLIMIT
DSN6GRP DSHARE=YES, X
GRPNAME=DSNDB1G, X
MEMBNAME=DBD1

END
/*
/*****
/* LINK EDIT THE NEW DSNZPARM MEMBER. PUT LOAD MODULE IN SDSNEXIT. *
/*****
//DSNTIZL EXEC PGM=IEWL,PARM='LIST,XREF,LET,RENT',
// COND=(4,LT)
//ADSNLOAD DD DISP=SHR,
// DSN=DB2410.SDSNLOAD
/** DD DISP=SHR,
/** DSN=DB2410.ADSNLOAD
//SYSPUNCH DD DSN=&&LOADSET(DSNTILMC),DISP=(OLD,DELETE)

```

```
//SYSLMOD DD DISP=SHR,  
//          DSN=DB2410.SDSNEXIT  
//SYSPRINT DD SYSOUT=*  
//SYSUDUMP DD SYSOUT=*  
//SYSUT1   DD UNIT=SYSDA,SPACE=(1024,(50,50))  
//SYSLIN   DD *  
           INCLUDE SYSPUNCH(DSNTILMC)  
           INCLUDE ADSNLOAD(DSNZPARM)  
           INCLUDE ADSNLOAD(DSNAA)  
           ORDER DSNAA  
           ENTRY  DSNZMSTR  
           NAME   DSD1PRMC(R)  
/*
```

A.2 Original RCT for IODM

This is the original RCT that we started with for IODM.

```
***** 00001000
* * 00002000
* SAMPLE CICS RCT FOR OBJECT DISTRIBUTION MANAGER VERSION 2.2 * 00003000
* * 00004000
* NOTE - X NEEDS TO BE CHANGED TO THE CORRECT SUFFIX * 00005000
* NOTE - XXXX NEEDS TO BE CHANGED TO THE CORRECT DB2 SUBSYSTEM NAME * 00006000
* * 00007000
* * 00007900
***** 00008000
      DSNCRCT TYPE=INIT,SUFFIX=1,SUBID=DBD1,THRDMAX=100 00009004
      DSNCRCT TYPE=POOL,THRDA=80,THRDM=80,TWAIT=YES 00009102
* **** CHANGE THE DISP TO DISD FOR USING THE EKCCDDTE EXIT ***** 00009800
*      DSNCRCT TYPE=ENTRY,TXID=DISP,PLAN=TPOPDISP, X 00009900
*      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X 00010000
*      THRDM=3,THRDA=3,THRDS=3 00010104
      DSNCRCT TYPE=ENTRY,TXID=DISD,PLAN=TPOPDISP, X00010200
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X00010300
      THRDM=3,THRDA=3,THRDS=3 00010404
* ***** 00010500
      DSNCRCT TYPE=ENTRY,TXID=STAH,PLAN=TPOPSTAH, X00010600
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X00010700
      THRDM=3,THRDA=3,THRDS=3 00010804
      DSNCRCT TYPE=ENTRY,TXID=STOR,PLAN=TPOPSTOR, X00010900
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X00011000
      THRDM=3,THRDA=3,THRDS=3 00011104
      DSNCRCT TYPE=ENTRY,TXID=(PRNT,RPRT),PLAN=TPOPPRNT, X00011200
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X00011300
      THRDM=1,THRDA=1,THRDS=1 00011400
      DSNCRCT TYPE=ENTRY,TXID=TMPC,PLAN=TPOPTMPC, X00011500
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X00011600
      THRDM=3,THRDA=3,THRDS=3 00011704
      DSNCRCT TYPE=ENTRY,TXID=(PRTW,RPRW),PLAN=TPOPPRTW, X00011800
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X00011900
      THRDM=1,THRDA=1,THRDS=0 00012000
      DSNCRCT TYPE=ENTRY,TXID=OMWS,PLAN=TPOPOMWS, X00012100
      ROLBE=NO,TWAIT=YES,AUTH=(EKCCICS), X00012200
      THRDM=1,THRDA=1,THRDS=0 00012300
      DSNCRCT TYPE=ENTRY,TXID=(PREF,DPRF),PLAN=TPOPPREF, X00012400
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS), X00012500
      THRDM=1,THRDA=1,THRDS=0 00012600
      DSNCRCT TYPE=ENTRY,TXID=(DELE,CHGE,DELM),PLAN=TPOPDELE, X00012700
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS) 00012800
      DSNCRCT TYPE=ENTRY,TXID=MODA,PLAN=TPOPMODA, X00012900
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS) 00013000
      DSNCRCT TYPE=ENTRY,TXID=OIUP,PLAN=TPOPOIUP, X00013100
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS) 00013300
      DSNCRCT TYPE=ENTRY,TXID=COBJ,PLAN=TPOPCOBJ, X00013400
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS) 00013500
      DSNCRCT TYPE=ENTRY,TXID=(QQBK,ORBW),PLAN=CBRIDBS, X00013600
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS) 00013700
      DSNCRCT TYPE=ENTRY,TXID=ORBK,PLAN=TPOPORBK, X00013800
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS) 00013900
      DSNCRCT TYPE=ENTRY,TXID=OQSV,PLAN=TPOPOQSV, X00014000
      ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS) 00014100
      DSNCRCT TYPE=ENTRY,TXID=(OQPM,OQPC),PLAN=TPOPOAQC, X00014200
```

```

        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00014300
    DSNCRCT TYPE=ENTRY, TXID=DSCT, PLAN=TPOPSCT,                            X00014400
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00014500
*****
** RCT ENTRIES FOR OBJECT DISTRIBUTION MANAGER V2.1                        *    00014700
*****
        DSNCRCT TYPE=ENTRY, TXID=OTBL, PLAN=TPOPOTBL,                       X00014900
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00015000
        DSNCRCT TYPE=ENTRY, TXID=ASND, PLAN=TPOPASND,                       X00015100
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00015200
*****
** RCT ENTRIES FOR OBJECT DISTRIBUTION MANAGER V2.1.1                    *    00015400
*****
        DSNCRCT TYPE=ENTRY, TXID=SSTO, PLAN=TPOPSSTO,                       X00015600
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS) ,                             X00015700
        THRDM=3, THRDA=3, THRDS=0                                           00015800
        DSNCRCT TYPE=ENTRY, TXID=SRET, PLAN=TPOPSRET,                       X00015900
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS) ,                             X00016000
        THRDM=3, THRDA=3, THRDS=0                                           00016100
*****
** RCT ENTRIES FOR OBJECT DISTRIBUTION MANAGER V2.2                        *    00016300
*****
        DSNCRCT TYPE=ENTRY, TXID=OSCN, PLAN=TPOPOSCN,                       X00016500
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00016600
        DSNCRCT TYPE=ENTRY, TXID=BEGB, PLAN=TPOPBEGB,                       X00016700
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00016800
        DSNCRCT TYPE=ENTRY, TXID=SLST, PLAN=TPOPSLST,                       X00017000
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00017100
        DSNCRCT TYPE=ENTRY, TXID=OINT, PLAN=TPOPOINT,                       X00017200
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00017300
        DSNCRCT TYPE=ENTRY, TXID=OWMS, PLAN=TPOPOWMS,                       X00017400
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00017500
*
*=====
** NEW RCT ENTRY FOR DB2 RETRY MODULE                                     ** 00017800
*=====
        DSNCRCT TYPE=ENTRY, TXID=ORET, PLAN=TPOPORET,                       X00018000
        ROLBE=NO, TWAIT=POOL, AUTH=( EKCCICS)                                00019000
*=====
*
        DSNCRCT TYPE=FINAL                                                    00020100
        END                                                                    00020200
        END                                                                    00020300

```



```

*
EM00      DSNCRCT TYPE=ENTRY,                                00600003
          PLAN=TPAOMD00,                                    X00610003
          THRDM=1,                                         PLAN NAME          X00620003
          THRDA=1,                                         MAX NBR IN A THREAD SET X00630003
          THRDS=0,                                         NBR OF ACTIVE THREAD  X00640003
          TWAIT=YES,                                       NBR ACT THRDS AT     X00650003
          AUTH=(ENTCICS),                                   TRAN TO WAIT FOR THREAD X00660003
          TXID=EM00,                                       AUTHORIZATION GROUP ID X00670003
          TXID=EM00,                                       TRANSACTION ID       00680003
*
*
*****
* NOTE - THE FOLLOWING IS AN EM01 RCT MACRO ENTRY.          * 00710003
* THIS IS FOR TABLE SET 01.                               * 00720003
* THE PLAN NAME CAN BE MODIFIED.                           * 00730003
*****
*
EM01      DSNCRCT TYPE=ENTRY,                                X00760003
          PLAN=TPAOMD01,                                    PLAN NAME          X00770003
          THRDM=1,                                         MAX NBR IN A THREAD SET X00780003
          THRDA=1,                                         NBR OF ACTIVE THREAD  X00790003
          THRDS=0,                                         NBR ACT THRDS AT     X00800003
          TWAIT=YES,                                       TRAN TO WAIT FOR THREAD X00810003
          AUTH=(ENTCICS),                                   AUTHORIZATION GROUP ID X00820003
          TXID=EM01,                                       TRANSACTION ID       00830003
*
*
*****
* NOTE - THE FOLLOWING IS AN ER00 RCT MACRO ENTRY.          * 01760003
* THIS IS FOR TABLE SET 00.                               * 01770003
* THE PLAN NAME CAN BE MODIFIED.                           * 01780003
*****
*
ER00      DSNCRCT TYPE=ENTRY,                                X01810003
          PLAN=TPAODF00,                                    PLAN NAME          X01820003
          THRDM=1,                                         MAX NBR IN A THREAD SET X01830003
          THRDA=1,                                         NBR OF ACTIVE THREAD  X01840003
          THRDS=1,                                         NBR ACT THRDS AT     X01850003
          TWAIT=YES,                                       TRAN TO WAIT FOR THREAD X01860003
          AUTH=(ENTCICS),                                   AUTHORIZATION GROUP ID X01870003
          TXID=ER00,                                       TRANSACTION ID       01880003
*
*
*****
* NOTE - THE FOLLOWING IS AN ER01 RCT MACRO ENTRY.          * 01910003
* THIS IS FOR TABLE SET 01.                               * 01920003
* THE PLAN NAME CAN BE MODIFIED.                           * 01930003
*****
*
ER01      DSNCRCT TYPE=ENTRY,                                X01960003
          PLAN=TPAODF01,                                    PLAN NAME          X01970003
          THRDM=1,                                         MAX NBR IN A THREAD SET X01980003
          THRDA=1,                                         NBR OF ACTIVE THREAD  X01990003
          THRDS=0,                                         NBR ACT THRDS AT     X02000003
          TWAIT=YES,                                       TRAN TO WAIT FOR THREAD X02010003
          AUTH=(ENTCICS),                                   AUTHORIZATION GROUP ID X02020003
          TXID=ER01,                                       TRANSACTION ID       02030003
*
*
*****
* NOTE - THE FOLLOWING IS AN EYDM RCT MACRO ENTRY.          * 02960003
* THIS IS FOR ALL TABLE SETS.                             * 02970003
* THE PLAN NAME CAN BE MODIFIED.                           * 02980003

```

```

***** 02990003
*
EYDM      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=EYDM              TRANSACTION ID
*
***** 03000003
***** 03010003
* NOTE - THE FOLLOWING IS AN EYDT RCT MACRO ENTRY.
* THIS IS FOR ALL TABLE SETS.
* THE PLAN NAME CAN BE MODIFIED.
***** 03100003
*
EYDT      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=EYDT              TRANSACTION ID
*
***** 03110003
***** 03120003
***** 03130003
***** 03140003
*
EYRT      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=EYRT              TRANSACTION ID
*
***** 03150003
***** 03160003
***** 03170003
***** 03180003
***** 03190003
***** 03200003
***** 03210003
***** 03220003
***** 03230003
*
EYRT      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=EYRT              TRANSACTION ID
*
***** 03240003
***** 03250003
* NOTE - THE FOLLOWING IS AN EYRT RCT MACRO ENTRY.
* THIS IS FOR ALL TABLE SETS.
* THE PLAN NAME CAN BE MODIFIED.
***** 03260003
***** 03270003
***** 03280003
***** 03290003
*
EYRT      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=EYRT              TRANSACTION ID
*
***** 03300003
***** 03310003
***** 03320003
***** 03330003
***** 03340003
***** 03350003
***** 03360003
***** 03370003
***** 03380003
*
EYRT      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=EYRT              TRANSACTION ID
*
***** 03390003
***** 03400003
* NOTE - THE FOLLOWING IS AN UFIM RCT MACRO ENTRY.
* THIS IS FOR ALL TABLE SETS.
* THE PLAN NAME CAN BE MODIFIED.
***** 03410003
***** 03420003
***** 03430003
***** 03440003
*
UFIM      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=UFIM              TRANSACTION ID
*
***** 03450003
***** 03460003
***** 03470003
***** 03480003
***** 03490003
***** 03500003
***** 03510003
***** 03520003
***** 03530003
*
EYRT      DSNCRCT TYPE=ENTRY,
          PLAN=TPAIODM,          PLAN NAME
          THRDM=1,              MAX NBR IN A THREAD SET
          THRDA=1,              NBR OF ACTIVE THREAD
          THRDS=0,              NBR ACT THRDS AT
          TWAIT=YES,            TRAN TO WAIT FOR THREAD
          AUTH=(ENTCICS),        AUTHORIZATION GROUP ID
          TXID=EYRT              TRANSACTION ID
*
***** 03540003
***** 03550003
* NOTE - THE FOLLOWING IS AN UFPF RCT MACRO ENTRY.
* THIS IS FOR ALL TABLE SETS.
***** 03560003
***** 03570003

```

```

* THE PLAN NAME CAN BE MODIFIED. * 03580003
***** 03590003
* 03600003
UFPF DSNCRCT TYPE=ENTRY, X03610003
      PLAN=TPAIODM, PLAN NAME X03620003
      THRDM=1, MAX NBR IN A THREAD SET X03630003
      THRDA=1, NBR OF ACTIVE THREAD X03640003
      THRDS=0, NBR ACT THRDS AT X03650003
      TWAIT=YES, TRAN TO WAIT FOR THREAD X03660003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X03670003
      TXID=UFPF TRANSACTION ID 03680003
* 03690003
***** 03700003
* NOTE - THE FOLLOWING IS AN UFST RCT MACRO ENTRY. * 03710003
* THIS IS FOR ALL TABLE SETS. * 03720003
* THE PLAN NAME CAN BE MODIFIED. * 03730003
***** 03740003
* 03750003
UFST DSNCRCT TYPE=ENTRY, X03760003
      PLAN=TPAODF00, PLAN NAME X03770003
      THRDM=1, MAX NBR IN A THREAD SET X03780003
      THRDA=1, NBR OF ACTIVE THREAD X03790003
      THRDS=0, NBR ACT THRDS AT X03800003
      TWAIT=YES, TRAN TO WAIT FOR THREAD X03810003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X03820003
      TXID=UFST TRANSACTION ID 03830003
* 03840003
* 03850003
***** 03860003
* NOTE - THE FOLLOWING IS A UFRE RCT MACRO ENTRY. * 03870003
* THE PLAN NAME CAN BE MODIFIED. * 03880003
* THIS IS ONLY NECESSARY WHEN MIGRATING FROM IPFAF V1.2 TO * 03890003
* IPFAF V2.2. * 03900003
* THE PLAN NAME CAN BE MODIFIED. * 03910003
***** 03920003
* 03930003
UFRE DSNCRCT TYPE=ENTRY, X03940003
      PLAN=TPAZIMIG, PLAN NAME X03950003
      THRDM=1, MAX NBR IN A THREAD SET X03960003
      THRDA=1, NBR OF ACTIVE THREAD X03970003
      THRDS=0, NBR ACT THRDS AT X03980003
      TWAIT=YES, TRAN TO WAIT FOR THREAD X03990003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X04000003
      TXID=UFRE TRANSACTION ID 04010003
* 04020003
* 04030003
***** 04040003
* NOTE - THE FOLLOWING IS A VER1 RCT MACRO ENTRY. * 04050003
* THIS IS FOR THE API VERIFICATION TRANSACTION VER1. * 04060003
* THE PLAN NAME CAN BE MODIFIED. * 04070003
***** 04080003
* 04090003
VER1 DSNCRCT TYPE=ENTRY, X04100003
      PLAN=TPAVER1, PLAN NAME X04110003
      THRDM=1, MAX NBR IN A THREAD SET X04120003
      THRDA=1, NBR OF ACTIVE THREAD X04130003
      THRDS=0, NBR ACT THRDS AT X04140003
      TWAIT=YES, TRAN TO WAIT FOR THREAD X04150003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X04160003

```

	TXID=VER1	TRANSACTION ID	04170003
*			04180003
*			04190003
*****			04200003
*	THE FOLLOWING 4 ENTRIES ARE NEW FOR FAF API V2.2		* 04210003
*****			04220003
*			04230003
*****			04240003
*	NOTE - THE FOLLOWING IS AN PSCH RCT MACRO ENTRY.		* 04250003
*	THIS IS FOR ALL TABLE SETS.		* 04260003
*	THE PLAN NAME CAN BE MODIFIED.		* 04270003
*****			04280003
*			04290003
PSCH	DSNCRCT TYPE=ENTRY,		X04300003
	PLAN=TPAAPSCH,	PLAN NAME	X04310003
	THRDM=3,	MAX NBR IN A THREAD SET	X04320003
	THRDA=3,	NBR OF ACTIVE THREAD	X04330003
	THRDS=0,	NBR ACT THRDS AT	X04340003
	TWAIT=YES,	TRAN TO WAIT FOR THREAD	X04350003
	ROLBE=NO,	DO NOT ROLLBACK IF DEADLOCK	X04360003
	AUTH=(ENTCICS),	AUTHORIZATION GROUP ID	X04370003
	TXID=(PSCH,EYPP)	TRANSACTION ID	04380003
*			04390003
*			04400003
*****			04410003
*	NOTE - THE FOLLOWING IS AN SNPA RCT MACRO ENTRY.		* 04420003
*	THIS IS FOR ALL TABLE SETS.		* 04430003
*	THE PLAN NAME CAN BE MODIFIED.		* 04440003
*	YOU MAY WANT TO CONSIDER CHANGING THE VALUE OF THE THRDM PARAMETER		* 04450003
*	TO ALLOW CONCURRENT TRANSACTIONS.		* 04460003
*****			04470003
*			04480003
SNPA	DSNCRCT TYPE=ENTRY,		X04490003
	PLAN=TPAASNPA,	PLAN NAME	X04500003
	THRDM=3,	MAX NBR IN A THREAD SET	X04510003
	THRDA=3,	NBR OF ACTIVE THREAD	X04520003
	THRDS=0,	NBR ACT THRDS AT	X04530003
	TWAIT=YES,	TRAN TO WAIT FOR THREAD	X04540003
	ROLBE=NO,	DO NOT ROLLBACK IF DEADLOCK	X04550003
	AUTH=(ENTCICS),	AUTHORIZATION GROUP ID	X04560003
	TXID=SNPA	TRANSACTION ID	04570003
*			04580003
*			04590003
*****			04600003
*	NOTE - THE FOLLOWING IS AN SNPF RCT MACRO ENTRY.		* 04610003
*	THIS IS FOR ALL TABLE SETS.		* 04620003
*	THE PLAN NAME CAN BE MODIFIED.		* 04630003
*	YOU MAY WANT TO CONSIDER CHANGING THE VALUE OF THE THRDM PARAMETER		* 04640003
*	TO ALLOW CONCURRENT TRANSACTIONS.		* 04650003
*****			04660003
*			04670003
SNPF	DSNCRCT TYPE=ENTRY,		X04680003
	PLAN=TPAASNPF,	PLAN NAME	X04690003
	THRDM=3,	MAX NBR IN A THREAD SET	X04700003
	THRDA=3,	NBR OF ACTIVE THREAD	X04710003
	THRDS=0,	NBR ACT THRDS AT	X04720003
	TWAIT=YES,	TRAN TO WAIT FOR THREAD	X04730003
	ROLBE=NO,	DO NOT ROLLBACK IF DEADLOCK	X04740003
	AUTH=(ENTCICS),	AUTHORIZATION GROUP ID	X04750003

```

          TXID=SNPF                TRANSACTION ID                04760003
*
*
*
*****
* NOTE - THE FOLLOWING IS AN EYSE RCT MACRO ENTRY.                * 04800003
* THIS IS FOR ALL TABLE SETS.                                    * 04810003
* THE PLAN NAME CAN BE MODIFIED.                                  * 04820003
* YOU MAY WANT TO CONSIDER CHANGING THE VALUE OF THE THRDM PARAMETER * 04830003
* TO ALLOW CONCURRENT TRANSACTIONS.                              * 04840003
*****
*
*
EYSE      DSNCRCT TYPE=ENTRY,                X04870003
          PLAN=TPAIODM2,                    PLAN NAME                X04880003
          THRDM=3,                          MAX NBR IN A THREAD SET X04890003
          THRDA=3,                          NBR OF ACTIVE THREAD    X04900003
          THRDS=0,                          NBR ACT THRDS AT       X04910003
          TWAIT=YES,                        TRAN TO WAIT FOR THREAD X04920003
          ROLBE=NO,                         DO NOT ROLLBACK IF DEADLOCK X04930003
          AUTH=(ENTCICS),                   AUTHORIZATION GROUP ID  X04940003
          TXID=EYSE                          TRANSACTION ID           04950003
*
*
*
*****
*
*
* SAMPLE CICS RCT FOR IPFAF
*
* NOTE- ENTCICS IS THE CICS REGION NAME
*
* 05/07/93  APAR PN40013 - ADDED ENTRY
*            FOR NEW DBCS TRANSACTION
*            (IBLK).
*
* 09/21/93  APAR PN45991 - ADDED FAST PATH
*            ENTRIES: FOLDER CONTENTS FOR
*            MODIFICATION (FLDM), FOLDER
*            CONTENTS FOR PAGE PAGE CHANGES
*            (FLDP) AND DELETED DOCUMENTS
*            (FLDD).
*
*
*****
          DSNCRCT TYPE=ENTRY, TXID=(SGON), TWAIT=POOL,                X0000000
          THRDM=3,                    MAX NBR IN A THREAD SET    X04700003
          THRDA=3,                    NBR OF ACTIVE THREAD      X04710003
          THRDS=3,                    NBR ACT THRDS AT         X04720003
          AUTH=(ENTCICS), PLAN=TPFSGON                                0000000
          DSNCRCT TYPE=ENTRY, TXID=(FAF),                              X0000000
          THRDM=1, THRDA=1, AUTH=(ENTCICS), PLAN=TPFFAF              0000000
          DSNCRCT TYPE=ENTRY, TXID=(IFAF),                             X0000000
          THRDM=1, THRDA=1, AUTH=(ENTCICS), PLAN=TPFIFAF            0000000
          DSNCRCT TYPE=ENTRY, TXID=(SGOF),                             X0000000
          THRDM=1, THRDA=1, AUTH=(ENTCICS), PLAN=TPFSGOF            0000000
          DSNCRCT TYPE=ENTRY, TXID=(AFLD),                             X0000000
          THRDM=1, THRDA=1, AUTH=(ENTCICS), PLAN=TPFAFLD            0000000
          DSNCRCT TYPE=ENTRY, TXID=(UFLD),                             X0000000
          THRDM=1, THRDA=1, AUTH=(ENTCICS), PLAN=TPFUFLD            0000000
          DSNCRCT TYPE=ENTRY, TXID=(LFLD),                             X0000000
          THRDM=1, THRDA=1, AUTH=(ENTCICS), PLAN=TPFLFLD            0000000
          DSNCRCT TYPE=ENTRY, TXID=(FNOT),                             X0000000
          THRDM=1, THRDA=1, AUTH=(ENTCICS), PLAN=TPFFNOT            0000000
          DSNCRCT TYPE=ENTRY, TXID=(ANOT),                             X0000000

```



```

DSNCRCT TYPE=ENTRY, TXID=( $M01 ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCM01 0000000
***** 0000000
* IPFAF : RESOURCE CONTROL TABLE ENTRIES * 0000000
* WORKFLOW RELATED * 0000000
***** 0000000
DSNCRCT TYPE=ENTRY, TXID=( ADOR ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFADOR 0000000
DSNCRCT TYPE=ENTRY, TXID=( GETW ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFGETW 0000000
DSNCRCT TYPE=ENTRY, TXID=( WDUK ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFWDUP 0000000
DSNCRCT TYPE=ENTRY, TXID=( ASGN ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFASGN 0000000
DSNCRCT TYPE=ENTRY, TXID=( HOLD ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFHOLD 0000000
DSNCRCT TYPE=ENTRY, TXID=( ROUT ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFROUT 0000000
DSNCRCT TYPE=ENTRY, TXID=( QUE ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFQUE 0000000
***** TABLE SET 00 ***** 0000000
DSNCRCT TYPE=ENTRY, X00760003
      PLAN=TPFCAR0, PLAN NAME X00770003
      THRDM=10, MAX NBR IN A THREAD SET X00780003
      THRDA=10, NBR OF ACTIVE THREAD X00790003
      THRDS=10, NBR ACT THRDS AT X00800003
      TWAIT=POOL, TRAN TO WAIT FOR THREAD X00810003
      AUTH=( ENTCICS ), AUTHORIZATION GROUP ID X00820003
      TXID=$AR0 TRANSACTION ID 00830003
DSNCRCT TYPE=ENTRY, TXID=( $AD0 ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCAD0 0000000
DSNCRCT TYPE=ENTRY, TXID=( $MD0 ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCMD0 0000000
DSNCRCT TYPE=ENTRY, TXID=( $GW0 ), X0000000
      THRDM=10, MAX NBR IN A THREAD SET X00780003
      THRDA=10, NBR OF ACTIVE THREAD X00790003
      THRDS=10, NBR ACT THRDS AT X00800003
      TWAIT=POOL, TRAN TO WAIT FOR THREAD X00810003
      AUTH=( ENTCICS ), PLAN=TPFCGW0 0000000
DSNCRCT TYPE=ENTRY, TXID=( $WU0 ), X0000000
      THRDM=10, MAX NBR IN A THREAD SET X00780003
      THRDA=10, NBR OF ACTIVE THREAD X00790003
      THRDS=10, NBR ACT THRDS AT X00800003
      TWAIT=POOL, TRAN TO WAIT FOR THREAD X00810003
      AUTH=( ENTCICS ), PLAN=TPFCWU0 0000000
DSNCRCT TYPE=ENTRY, TXID=( $QD0 ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCQD0 0000000
DSNCRCT TYPE=ENTRY, TXID=( $DD0 ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCDD0 0000000
DSNCRCT TYPE=ENTRY, TXID=( $UA0 ), X0000000
      THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCUA0 0000000
DSNCRCT TYPE=ENTRY, TXID=( $GT0 ), X0000000
      THRDM=10, MAX NBR IN A THREAD SET X00780003
      THRDA=10, NBR OF ACTIVE THREAD X00790003
      THRDS=10, NBR ACT THRDS AT X00800003
      TWAIT=POOL, TRAN TO WAIT FOR THREAD X00810003
      AUTH=( ENTCICS ), PLAN=TPFCGT0 0000000
***** TABLE SET 01 ***** 0000000
DSNCRCT TYPE=ENTRY, TXID=( $AR1 ), X0000000

```

```

        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCAR1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $AD1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCAD1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $MD1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCMD1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $GW1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCGW1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $WU1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCWU1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $QD1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCQD1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $DD1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCDD1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $UA1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCUA1           0000000
DSNCRCT TYPE=ENTRY,TXID=( $GT1 ),                          X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCGT1           0000000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES                  *
*   PREFETCH RELATED                                       *
*****
        DSNCRCT TYPE=ENTRY,TXID=( FAFM ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFFAFM           0000000
***** TABLE SET 00 *****
        DSNCRCT TYPE=ENTRY,TXID=( $PL0 ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCPL0           0000000
        DSNCRCT TYPE=ENTRY,TXID=( $PN0 ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCPN0           0000000
***** TABLE SET 01 *****
        DSNCRCT TYPE=ENTRY,TXID=( $PL1 ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCPL1           0000000
        DSNCRCT TYPE=ENTRY,TXID=( $PN1 ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCPN1           0000000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES                  *
*   HELP RELATED                                           *
*****
        DSNCRCT TYPE=ENTRY,TXID=( $HP ),                     X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCHP           0000000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES                  *
*   LOB DRIVER RELATED                                     *
*****
        DSNCRCT TYPE=ENTRY,TXID=( LOB ),                     X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFL0B           0000000
        DSNCRCT TYPE=ENTRY,TXID=( LOB2 ),                   X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFLB2           0000000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES                  *
*   INDEXING WORKSTATION INTERFACE RELATED                 *
*****
        DSNCRCT TYPE=ENTRY,TXID=( AUTH ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFAUTH           0000000
        DSNCRCT TYPE=ENTRY,TXID=( DFLT ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFDFLT           0000000
        DSNCRCT TYPE=ENTRY,TXID=( INDX ),                    X0000000
        THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFINDX           0000000
        DSNCRCT TYPE=ENTRY,TXID=( IBLK ),                    X0000000

```

```

                                THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFIBLK
*****
*   ON LINE AUTO INDEXING PLAN
*****
ATX0      DSNCRCT TYPE=ENTRY, TXID=( ATX0 ),
                                THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=AUTOIX0
ATX1      DSNCRCT TYPE=ENTRY, TXID=( ATX1 ),
                                THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=AUTOIX1
                                DSNCRCT TYPE=FINAL
                                END

```

```

0000000
0000000
0000000
0000000
X0000000
0000000
X0000000
0000000
04980003
04990003

```

A.4 Final RCT for IODM

This is the final RCT used in IODM. It reflects the changes we made for Chapter 6, "Tuning Tips" on page 29.

```
DSNCRCT TYPE=INIT,SUFFIX=1,SUBID=DBD1,THRDMAX=100          X00001000
PURGEC=0,55                                                00002000
DSNCRCT TYPE=COMD,AUTH=(SIGNID),                          X00003000
THRDA=1,THRDM=1,TWAIT=YES                                00004000
DSNCRCT TYPE=POOL,THRDA=48,THRDM=48,TWAIT=YES            00005000
* *****                                                    00006000
DSNCRCT TYPE=ENTRY,PLAN=TPNSO,                            X00007000
ROLBE=NO,TWAIT=POOL,AUTH=(EKCCICS),                      X00008000
THRDM=50,THRDA=50,THRDS=50,                             X00009000
TXID=(DISD,STAH,STOR,PRNT,RPRT,TMPC,PRTW,               X00010000
RPRW,OMWS,PREF,DPRF,DELE,CHGE,DELM,MODA,                X00010100
OIUP,COBJ,ORBK,ORBW,OQSV,OAQM,OAQC,DSCT,                X00010200
OTBL,ASND,SSTO,SRET,OSCN,BEGB,SLST,OINT,                 X00010300
OWMS,ORET)                                               00010400
*=====*                                                00010500
*                                                           00010600
DSNCRCT TYPE=FINAL                                       00010700
END                                                       00010800
```

A.5 Final RCT for IPFAF

This is the RCT for IPFAF after the recommendations made in Chapter 6, "Tuning Tips" on page 29 had been made.

```

*****
*
* SAMPLE CICS RCT FOR FAF API V2.2
*
* NOTE - XX NEEDS TO BE CHANGED TO THE CORRECT SUFFIX
*       ?DB2SUB? NEEDS TO BE CHANGED TO THE DB2
*       SUBSYSTEM ID
*       EYPCICS IS CURRENTLY THE AUTHID. CHANGE IT
*       ACCORDING TO YOUR SYSTEM.
*****
*
*       TITLE 'DSNCRCT - FOR API'
*
*       PRINT NOGEN
*****
* APPLICATIONS
* NOTE:
* MAKE SURE (SUM OF THRDA < THRDMAx 2).
* TO PROTECT THREADS FOR REUSE: THRDS > 0 AND THRDA > 0.
* TO QUEUE TRANS/TRAN GROUPS: THRDS > 0 AND TWAIT=YES.
* CUSTOMIZE THE THRDM PARAMETER TO ALLOW FOR CONCURRENT
* TRANSACTIONS.
* CHANGE TWAIT=POOL IF DESIRED.
*****
*
*       DSNCRCT TYPE=INIT,
*           SNAP=Y,
*           SUBID=DBD1,
*           SUFFIX=30,
*           ROLBI=YES,
*           TWAITI=POOL,
*           THRDMAx=100
*
*           THREAD SUBTASK FAILURE OUTPUT CLX
*           DB2 SUBSYSTEM ID
*           SUFFIX ID
*           SUFFIX ID
*           SUFFIX ID
*           MAX NUMBER OF THREADS
*
*       DSNCRCT TYPE=COMD,
*           ROLBE=NO,
*           THRDM=1,
*           THRDA=1,
*           THRDS=1,
*           TWAIT=YES
*
*           ROLLBACK AT DEADLOCK
*           ROLLBACK AT DEADLOCK
*           ROLLBACK AT DEADLOCK
*           ROLLBACK AT DEADLOCK
*           OVERFLOW TO POOL
*
*       DSNCRCT TYPE=POOL,
*           PLAN=DSN8CC31,
*           THRDM=27,
*           THRDA=27,
*           TWAIT=YES
*
*           PLAN NAME FOR DB2 IVP
*           MAX NUMBER IN A THREAD SET
*           NUMBER OF ACTIVE THREAD
*           WAIT FOR THREAD
*****
*
* API RCT ENTRIES
*****

```

```

*
***** 00540003
***** 00550003
* NOTE - THE FOLLOWING IS AN EM00 RCT MACRO ENTRY. * 00560003
* THIS IS FOR TABLE SET 00. * 00570003
* THE PLAN NAME CAN BE MODIFIED. * 00580003
***** 00590003
* 00600003
EM00 DSNCRCT TYPE=ENTRY, X00610003
      PLAN=TPAOMD00, PLAN NAME X00620003
      THRDM=1, MAX NBR IN A THREAD SET X00630003
      THRDA=1, NBR OF ACTIVE THREAD X00640003
      THRDS=1, NBR ACT THRDS AT X00650003
      TWAIT=YES, TRAN TO WAIT FOR THREAD X00660003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X00670003
      TXID=EM00 TRANSACTION ID 00680003
* 00690003
***** 00700003
* NOTE - THE FOLLOWING IS AN EM01 RCT MACRO ENTRY. * 00710003
* THIS IS FOR TABLE SET 01. * 00720003
* THE PLAN NAME CAN BE MODIFIED. * 00730003
***** 00740003
* 00750003
EM01 DSNCRCT TYPE=ENTRY, X00760003
      PLAN=TPAOMD01, PLAN NAME X00770003
      THRDM=1, MAX NBR IN A THREAD SET X00780003
      THRDA=1, NBR OF ACTIVE THREAD X00790003
      THRDS=1, NBR ACT THRDS AT X00800003
      TWAIT=YES, TRAN TO WAIT FOR THREAD X00810003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X00820003
      TXID=EM01 TRANSACTION ID 00830003
* 01740003
***** 01750003
* NOTE - THE FOLLOWING IS AN ER00 RCT MACRO ENTRY. * 01760003
* THIS IS FOR TABLE SET 00. * 01770003
* THE PLAN NAME CAN BE MODIFIED. * 01780003
***** 01790003
* 01800003
ER00 DSNCRCT TYPE=ENTRY, X01810003
      PLAN=TPAODF00, PLAN NAME X01820003
      THRDM=20, MAX NBR IN A THREAD SET X01830003
      THRDA=20, NBR OF ACTIVE THREAD X01840003
      THRDS=10, NBR ACT THRDS AT X01850003
      TWAIT=POOL, TRAN TO WAIT FOR THREAD X01860003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X01870003
      TXID=ER00 TRANSACTION ID 01880003
* 01890003
***** 01900003
* NOTE - THE FOLLOWING IS AN ER01 RCT MACRO ENTRY. * 01910003
* THIS IS FOR TABLE SET 01. * 01920003
* THE PLAN NAME CAN BE MODIFIED. * 01930003
***** 01940003
* 01950003
ER01 DSNCRCT TYPE=ENTRY, X01960003
      PLAN=TPAODF01, PLAN NAME X01970003
      THRDM=1, MAX NBR IN A THREAD SET X01980003
      THRDA=1, NBR OF ACTIVE THREAD X01990003
      THRDS=0, NBR ACT THRDS AT X02000003
      TWAIT=YES, TRAN TO WAIT FOR THREAD X02010003
      AUTH=(ENTCICS), AUTHORIZATION GROUP ID X02020003

```

```

          TXID=ER01                TRANSACTION ID                02030003
*
*
*****
* NOTE - THE FOLLOWING IS AN EYDM RCT MACRO ENTRY.                * 02940003
* THIS IS FOR ALL TABLE SETS.                                    * 02950003
* THE PLAN NAME CAN BE MODIFIED.                                  * 02960003
*****                                                            * 02970003
*
*
EYDM      DSNCRCT TYPE=ENTRY,                                       X03010003
          PLAN=TPAIODM,           PLAN NAME                        X03020003
          THRDM=1,                MAX NBR IN A THREAD SET        X03030003
          THRDA=1,                NBR OF ACTIVE THREAD        X03040003
          THRDS=0,                NBR ACT THRDS AT           X03050003
          TWAIT=YES,              TRAN TO WAIT FOR THREAD    X03060003
          AUTH=(ENTCICS),          AUTHORIZATION GROUP ID    X03070003
          TXID=EYDM                TRANSACTION ID              03080003
*
*
*****
* NOTE - THE FOLLOWING IS AN EYDT RCT MACRO ENTRY.                * 03090003
* THIS IS FOR ALL TABLE SETS.                                    * 03100003
* THE PLAN NAME CAN BE MODIFIED.                                  * 03110003
*****                                                            * 03120003
*
*
EYDT      DSNCRCT TYPE=ENTRY,                                       X03130003
          PLAN=TPAIODM,           PLAN NAME                        X03140003
          THRDM=1,                MAX NBR IN A THREAD SET        X03150003
          THRDA=1,                NBR OF ACTIVE THREAD        X03160003
          THRDS=0,                NBR ACT THRDS AT           X03170003
          TWAIT=YES,              TRAN TO WAIT FOR THREAD    X03180003
          AUTH=(ENTCICS),          AUTHORIZATION GROUP ID    X03190003
          TXID=EYDT                TRANSACTION ID              03200003
*
*
*****
* NOTE - THE FOLLOWING IS AN EYRT RCT MACRO ENTRY.                * 03210003
* THIS IS FOR ALL TABLE SETS.                                    * 03220003
* THE PLAN NAME CAN BE MODIFIED.                                  * 03230003
*****                                                            * 03240003
*
*
EYRT      DSNCRCT TYPE=ENTRY,                                       X03250003
          PLAN=TPAIODM,           PLAN NAME                        X03260003
          THRDM=1,                MAX NBR IN A THREAD SET        X03270003
          THRDA=1,                NBR OF ACTIVE THREAD        X03280003
          THRDS=0,                NBR ACT THRDS AT           X03290003
          TWAIT=YES,              TRAN TO WAIT FOR THREAD    X03300003
          AUTH=(ENTCICS),          AUTHORIZATION GROUP ID    X03310003
          TXID=EYRT                TRANSACTION ID              03320003
*
*
*****
* NOTE - THE FOLLOWING IS AN UFIM RCT MACRO ENTRY.                * 03330003
* THIS IS FOR ALL TABLE SETS.                                    * 03340003
* THE PLAN NAME CAN BE MODIFIED.                                  * 03350003
*****                                                            * 03360003
*
*
UFIM      DSNCRCT TYPE=ENTRY,                                       X03370003
          PLAN=TPAIODM,           PLAN NAME                        X03380003
          THRDM=1,                MAX NBR IN A THREAD SET        X03390003
          THRDA=1,                NBR OF ACTIVE THREAD        X03400003
          THRDS=0,                NBR ACT THRDS AT           X03410003
          TWAIT=YES,              TRAN TO WAIT FOR THREAD    X03420003
          TXID=UFIM                TRANSACTION ID              03430003
*
*
*****
* NOTE - THE FOLLOWING IS AN UFIM RCT MACRO ENTRY.                * 03440003
* THIS IS FOR ALL TABLE SETS.                                    * 03450003
* THE PLAN NAME CAN BE MODIFIED.                                  * 03460003
*****                                                            * 03470003
*
*
UFIM      DSNCRCT TYPE=ENTRY,                                       X03480003
          PLAN=TPAIODM,           PLAN NAME                        X03490003
          THRDM=1,                MAX NBR IN A THREAD SET        X03500003
          THRDA=1,                NBR OF ACTIVE THREAD        X03510003
          THRDS=0,                NBR ACT THRDS AT           X03520003
          TWAIT=YES,              TRAN TO WAIT FOR THREAD    X03530003
          TXID=UFIM                TRANSACTION ID              03540003
*
*
*****

```

```

AUTH=(ENTCICS),          AUTHORIZATION GROUP ID      X03520003
TXID=UFIM                TRANSACTION ID                03530003
*
*
*****
* NOTE - THE FOLLOWING IS AN UFPF RCT MACRO ENTRY.          * 03560003
* THIS IS FOR ALL TABLE SETS.                             * 03570003
* THE PLAN NAME CAN BE MODIFIED.                           * 03580003
*****
*
UFPF      DSNCRCT TYPE=ENTRY,                                X03610003
          PLAN=TPAIODM,          PLAN NAME                X03620003
          THRDM=1,              MAX NBR IN A THREAD SET   X03630003
          THRDA=1,              NBR OF ACTIVE THREAD      X03640003
          THRDS=0,              NBR ACT THRDS AT          X03650003
          TWAIT=YES,            TRAN TO WAIT FOR THREAD    X03660003
          AUTH=(ENTCICS),       AUTHORIZATION GROUP ID   X03670003
          TXID=UFPF            TRANSACTION ID                03680003
*
*
*****
* NOTE - THE FOLLOWING IS AN UFST RCT MACRO ENTRY.          * 03710003
* THIS IS FOR ALL TABLE SETS.                             * 03720003
* THE PLAN NAME CAN BE MODIFIED.                           * 03730003
*****
*
UFST      DSNCRCT TYPE=ENTRY,                                X03760003
          PLAN=TPAODF00,       PLAN NAME                X03770003
          THRDM=1,              MAX NBR IN A THREAD SET   X03780003
          THRDA=1,              NBR OF ACTIVE THREAD      X03790003
          THRDS=0,              NBR ACT THRDS AT          X03800003
          TWAIT=YES,            TRAN TO WAIT FOR THREAD    X03810003
          AUTH=(ENTCICS),       AUTHORIZATION GROUP ID   X03820003
          TXID=UFST            TRANSACTION ID                03830003
*
*
*****
* NOTE - THE FOLLOWING IS A UFRE RCT MACRO ENTRY.          * 03870003
* THE PLAN NAME CAN BE MODIFIED.                           * 03880003
* THIS IS ONLY NECESSARY WHEN MIGRATING FROM IPFAF V1.2 TO * 03890003
* IPFAF V2.2.                                               * 03900003
* THE PLAN NAME CAN BE MODIFIED.                           * 03910003
*****
*
UFRE      DSNCRCT TYPE=ENTRY,                                X03940003
          PLAN=TPAZIMIG,       PLAN NAME                X03950003
          THRDM=1,              MAX NBR IN A THREAD SET   X03960003
          THRDA=1,              NBR OF ACTIVE THREAD      X03970003
          THRDS=0,              NBR ACT THRDS AT          X03980003
          TWAIT=YES,            TRAN TO WAIT FOR THREAD    X03990003
          AUTH=(ENTCICS),       AUTHORIZATION GROUP ID   X04000003
          TXID=UFRE            TRANSACTION ID                04010003
*
*
*****
* NOTE - THE FOLLOWING IS A VER1 RCT MACRO ENTRY.          * 04050003
* THIS IS FOR THE API VERIFICATION TRANSACTION VER1.       * 04060003
* THE PLAN NAME CAN BE MODIFIED.                           * 04070003
*****
*
VER1      DSNCRCT TYPE=ENTRY,                                X04100003

```

```

PLAN=TPAVER1,          PLAN NAME          X04110003
THRDM=1,              MAX NBR IN A THREAD SET X04120003
THRDA=1,              NBR OF ACTIVE THREAD    X04130003
THRDS=0,              NBR ACT THRDS AT       X04140003
TWAIT=YES,            TRAN TO WAIT FOR THREAD X04150003
AUTH=(ENTCICS),       AUTHORIZATION GROUP ID  X04160003
TXID=VER1             TRANSACTION ID          04170003
*                                                              04180003
*                                                              04190003
*****                                                              04200003
*   THE FOLLOWING 4 ENTRIES ARE NEW FOR FAF API V2.2          * 04210003
*****                                                              04220003
*                                                              04230003
*****                                                              04240003
* NOTE - THE FOLLOWING IS AN PSCH RCT MACRO ENTRY.          * 04250003
* THIS IS FOR ALL TABLE SETS.                              * 04260003
* THE PLAN NAME CAN BE MODIFIED.                            * 04270003
*****                                                              04280003
*                                                              04290003
PSCH   DSNCRCT TYPE=ENTRY, X04300003
        PLAN=TPAAPSCH,      PLAN NAME          X04310003
        THRDM=3,           MAX NBR IN A THREAD SET X04320003
        THRDA=3,           NBR OF ACTIVE THREAD    X04330003
        THRDS=0,           NBR ACT THRDS AT       X04340003
        TWAIT=YES,         TRAN TO WAIT FOR THREAD X04350003
        ROLBE=NO,          DO NOT ROLLBACK IF DEADLOCK X04360003
        AUTH=(ENTCICS),     AUTHORIZATION GROUP ID  X04370003
        TXID=(PSCH,EYPP)   TRANSACTION ID          04380003
*                                                              04390003
*                                                              04400003
*****                                                              04410003
* NOTE - THE FOLLOWING IS AN SNPA RCT MACRO ENTRY.          * 04420003
* THIS IS FOR ALL TABLE SETS.                              * 04430003
* THE PLAN NAME CAN BE MODIFIED.                            * 04440003
* YOU MAY WANT TO CONSIDER CHANGING THE VALUE OF THE THRDM * 04450003
*   TO ALLOW CONCURRENT TRANSACTIONS.                       * 04460003
*****                                                              04470003
*                                                              04480003
SNPA   DSNCRCT TYPE=ENTRY, X04490003
        PLAN=TPAASNPA,     PLAN NAME          X04500003
        THRDM=3,           MAX NBR IN A THREAD SET X04510003
        THRDA=3,           NBR OF ACTIVE THREAD    X04520003
        THRDS=0,           NBR ACT THRDS AT       X04530003
        TWAIT=YES,         TRAN TO WAIT FOR THREAD X04540003
        ROLBE=NO,          DO NOT ROLLBACK IF DEADLOCK X04550003
        AUTH=(ENTCICS),     AUTHORIZATION GROUP ID  X04560003
        TXID=SNPA          TRANSACTION ID          04570003
*                                                              04580003
*                                                              04590003
*****                                                              04600003
* NOTE - THE FOLLOWING IS AN SNPF RCT MACRO ENTRY.          * 04610003
* THIS IS FOR ALL TABLE SETS.                              * 04620003
* THE PLAN NAME CAN BE MODIFIED.                            * 04630003
* YOU MAY WANT TO CONSIDER CHANGING THE VALUE OF THE THRDM * 04640003
*   TO ALLOW CONCURRENT TRANSACTIONS.                       * 04650003
*****                                                              04660003
*                                                              04670003
SNPF   DSNCRCT TYPE=ENTRY, X04680003
        PLAN=TPAASNPF,     PLAN NAME          X04690003

```

THRDM=3,	MAX NBR IN A THREAD SET	X04700003
THRDA=3,	NBR OF ACTIVE THREAD	X04710003
THRDS=0,	NBR ACT THRDS AT	X04720003
TWAIT=YES,	TRAN TO WAIT FOR THREAD	X04730003
ROLBE=NO,	DO NOT ROLLBACK IF DEADLOCK	X04740003
AUTH=(ENTCICS),	AUTHORIZATION GROUP ID	X04750003
TXID=SNPF	TRANSACTION ID	04760003
*		04770003
*		04780003
*****		04790003
* NOTE - THE FOLLOWING IS AN EYSE RCT MACRO ENTRY.		* 04800003
* THIS IS FOR ALL TABLE SETS.		* 04810003
* THE PLAN NAME CAN BE MODIFIED.		* 04820003
* YOU MAY WANT TO CONSIDER CHANGING THE VALUE OF THE THRDM PARAMETER		* 04830003
* TO ALLOW CONCURRENT TRANSACTIONS.		* 04840003
*****		04850003
*		04860003
EYSE DSNCRCT TYPE=ENTRY,		X04870003
PLAN=TPAIODM2,	PLAN NAME	X04880003
THRDM=3,	MAX NBR IN A THREAD SET	X04890003
THRDA=3,	NBR OF ACTIVE THREAD	X04900003
THRDS=0,	NBR ACT THRDS AT	X04910003
TWAIT=YES,	TRAN TO WAIT FOR THREAD	X04920003
ROLBE=NO,	DO NOT ROLLBACK IF DEADLOCK	X04930003
AUTH=(ENTCICS),	AUTHORIZATION GROUP ID	X04940003
TXID=EYSE	TRANSACTION ID	04950003
*		05215000
DSNCRCT TYPE=ENTRY,PLAN=TPNSO,		X05210400
ROLBE=NO,AUTH=(ENTCICS),		X05210500
THRDM=60,THRDA=80,THRDS=80,		X05210600
TXID=(SGON,FAF,IFAF,SGOF,AFLD,UFLD,LFLD,FNOT,ANOT,		X05210900
FLDR,FLDM,FLDP,FLDD,ADOC,\$MNO,\$LFO,\$LCO,\$OCO,\$LNO,		X05211000
\$MFO,\$AOO,\$MEO,\$MWO,\$MOO,ADOR,GETW,WDUP,ASGN,HOLD,		X05212000
ROUT,QUE,\$ARO,\$ADO,\$MDO,\$GWO,\$WUO,\$QDO,\$DDO,\$UAO,		X05213000
\$GTO,FAFM,\$PLO,\$PNO,\$HP,LOB,LOB2,AUTH,DFLT,INDX)		05214000
*		05215000
DSNCRCT TYPE=ENTRY,TXID=(\$MN1),		X05690000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCMN1		05700000
DSNCRCT TYPE=ENTRY,TXID=(\$LF1),		X05710000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCLF1		05720000
DSNCRCT TYPE=ENTRY,TXID=(\$LC1),		X05730000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCLC1		05740000
DSNCRCT TYPE=ENTRY,TXID=(\$OC1),		X05750000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCOC1		05760000
DSNCRCT TYPE=ENTRY,TXID=(\$LN1),		X05770000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCLN1		05780000
DSNCRCT TYPE=ENTRY,TXID=(\$MF1),		X05790000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCMF1		05800000
DSNCRCT TYPE=ENTRY,TXID=(\$AO1),		X05810000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCAO1		05820000
DSNCRCT TYPE=ENTRY,TXID=(\$ME1),		X05830000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCME1		05840000
DSNCRCT TYPE=ENTRY,TXID=(\$MW1),		X05850000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCMW1		05860000
DSNCRCT TYPE=ENTRY,TXID=(\$MO1),		X05870000
THRDM=1,THRDA=1,AUTH=(ENTCICS),PLAN=TPFCMO1		05880000
*****		05890000
* IPFAF : RESOURCE CONTROL TABLE ENTRIES	*	05900000
* WORKFLOW RELATED	*	05910000

```

*****
*      DSNCRCT TYPE=ENTRY, TXID= ( ADOR ),                X 05930000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFADOR 05940000
*      DSNCRCT TYPE=ENTRY, TXID= ( GETW ),                X 05950000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFGETW 05960000
*      DSNCRCT TYPE=ENTRY, TXID= ( W DUP ),                X 05970000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFWDUP 05980000
*      DSNCRCT TYPE=ENTRY, TXID= ( ASGN ),                X 05990000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFASGN 06000000
*      DSNCRCT TYPE=ENTRY, TXID= ( HOLD ),                X 06010000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFHOLD 06020000
*      DSNCRCT TYPE=ENTRY, TXID= ( ROUT ),                X 06030000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFROUT 06040000
*      DSNCRCT TYPE=ENTRY, TXID= ( QUE ),                 X 06050000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFQUE 06060000
***** TABLE SET 00 *****
*      DSNCRCT TYPE=ENTRY, TXID= ( $AR0 ),                X 06080000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCAR0 06090000
*      DSNCRCT TYPE=ENTRY, TXID= ( $AD0 ),                X 06100000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCAD0 06110000
*      DSNCRCT TYPE=ENTRY, TXID= ( $MD0 ),                X 06120000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCMD0 06130000
*      DSNCRCT TYPE=ENTRY, TXID= ( $GW0 ),                X 06140000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCGW0 06150000
*      DSNCRCT TYPE=ENTRY, TXID= ( $WU0 ),                X 06160000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCWU0 06170000
*      DSNCRCT TYPE=ENTRY, TXID= ( $QD0 ),                X 06180000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCQD0 06190000
*      DSNCRCT TYPE=ENTRY, TXID= ( $DD0 ),                X 06200000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCDD0 06210000
*      DSNCRCT TYPE=ENTRY, TXID= ( $UA0 ),                X 06220000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCUA0 06230000
*      DSNCRCT TYPE=ENTRY, TXID= ( $GT0 ),                X 06240000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCGT0 06250000
***** TABLE SET 01 *****
*      DSNCRCT TYPE=ENTRY, TXID= ( $AR1 ),                X06270000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCAR1 06280000
*      DSNCRCT TYPE=ENTRY, TXID= ( $AD1 ),                X06290000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCAD1 06300000
*      DSNCRCT TYPE=ENTRY, TXID= ( $MD1 ),                X06310000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCMD1 06320000
*      DSNCRCT TYPE=ENTRY, TXID= ( $GW1 ),                X06330000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCGW1 06340000
*      DSNCRCT TYPE=ENTRY, TXID= ( $WU1 ),                X06350000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCWU1 06360000
*      DSNCRCT TYPE=ENTRY, TXID= ( $QD1 ),                X06370000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCQD1 06380000
*      DSNCRCT TYPE=ENTRY, TXID= ( $DD1 ),                X06390000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCDD1 06400000
*      DSNCRCT TYPE=ENTRY, TXID= ( $UA1 ),                X06410000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCUA1 06420000
*      DSNCRCT TYPE=ENTRY, TXID= ( $GT1 ),                X06430000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFCGT1 06440000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES                * 06450000
*      PREFETCH RELATED                                  * 06470000
*****
*      DSNCRCT TYPE=ENTRY, TXID= ( FAFM ),                X 06490000
*          THRDM=1, THRDA=1, AUTH= ( ENTCICS ), PLAN=TPFFAFM 06500000

```

```

***** TABLE SET 00 *****
*      DSNCRCT TYPE=ENTRY, TXID=( $PLO ),           X 06520000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCPLO           06530000
*      DSNCRCT TYPE=ENTRY, TXID=( $PN0 ),           X 06540000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCPN0           06550000
***** TABLE SET 01 *****
*      DSNCRCT TYPE=ENTRY, TXID=( $PL1 ),           X06570000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCPL1           06580000
*      DSNCRCT TYPE=ENTRY, TXID=( $PN1 ),           X06590000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCPN1           06600000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES           *           06620000
*      HELP RELATED                                 *           06630000
*****
*      DSNCRCT TYPE=ENTRY, TXID=( $HP ),           X 06650000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFCHP           06660000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES           *           06680000
*      LOB DRIVER RELATED                           *           06690000
*****
*      DSNCRCT TYPE=ENTRY, TXID=( LOB ),           X 06710000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFLOB           06720000
*      DSNCRCT TYPE=ENTRY, TXID=( LOB2 ),          X 06730000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFLB2           06740000
*****
* IPFAF : RESOURCE CONTROL TABLE ENTRIES           *           06760000
*      INDEXING WORKSTATION INTERFACE RELATED       *           06770000
*****
*      DSNCRCT TYPE=ENTRY, TXID=( AUTH ),           X 06790000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFAUTH           06800000
*      DSNCRCT TYPE=ENTRY, TXID=( DFLT ),          X 06810000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFDFLT           06820000
*      DSNCRCT TYPE=ENTRY, TXID=( INDX ),          X 06830000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=TPFINDX           06840000
* Note IBLK could get in the block entry - I didn't have room       06850000
IBLK  DSNCRCT TYPE=ENTRY,                           X06860000
      PLAN=TPFIBLK,                                PLAN NAME           X06870000
      THRDM=1,                                     MAX NBR IN A THREAD SET X06880000
      THRDA=0,                                     NBR OF ACTIVE THREAD   X06890000
      THRDS=0,                                     NBR ACT THRDS AT      X06900000
      AUTH=( ENTCICS ),                            AUTHORIZATION GROUP ID X06910000
      TXID=IBLK                                    TRANSACTION ID         06920000
*****
* ON LINE AUTO INDEXING PLAN                           *           06930000
*****
*      DSNCRCT TYPE=ENTRY, TXID=( ATX0 ),           X06960000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=AUTOIX0           06970000
*      DSNCRCT TYPE=ENTRY, TXID=( ATX1 ),           X06980000
*          THRDM=1, THRDA=1, AUTH=( ENTCICS ), PLAN=AUTOIX1           06990000
*      DSNCRCT TYPE=FINAL                           07000000
*      END                                           07010000

```

A.6 DB2 Package and Plan for IODM and IPFAF

This is the final DB2 packages and plan for IODM and IPFAF after the recommendations made in Chapter 6, "Tuning Tips" on page 29 had been made.

A.6.1 Sample: DB2 Package for IODM and IPFAF

```
//STEP1 EXEC PGM=IKJEFT01,DYNAMNBR=20,REGION=1024K 00010900
//DBRMLIB DD DSN=IMAGE2.API220.SEYPDBR1,DISP=SHR, 00011006
// VOL=SER=D83AE8,UNIT=SYSDA 00011106
// DD DSN=IMAGE2.API220.SEYPDBR3,DISP=SHR, 00011206
// VOL=SER=D83AE8,UNIT=SYSDA 00011306
// DD DSN=IMAGE2.FWA221.SENTDBR1,DISP=SHR, 00011206
// VOL=SER=D83AE8,UNIT=SYSDA 00011306
// DD DSN=IMAGE2.ODM220.SEKCDBR1,DISP=SHR, 00011206
// VOL=SER=D83AE8,UNIT=SYSDA 00011306
// DD DSN=SYS1.CBRDBRM,DISP=SHR 00011206
// DD DSN=IMAGES.COMMON.DBRMLIB,DISP=SHR 00011206
//SYSTSPRT DD SYSOUT=* 00011400
//SYSPRINT DD SYSOUT=* 00011500
//SYSUDUMP DD SYSOUT=* 00011600
//SYSTSIN DD * 00011700
DSN SYSTEM(DBD1) 00011806
BIND PACKAGE(TPNS) MEMBER(CBRHCAUD) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHCLDL) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHCLLU) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHCLMK) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHCLSG) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHCOMT) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHDDIR) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHDOBJ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHRLBK) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHROBJ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSBCC) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSBCN) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSBCT) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSBKV) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSOBJ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSPCC) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSPCN) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSPCT) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSPDT) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHSVOL) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHTBSV) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHUNX ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHUPDT) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHUPDV) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHUVOL) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRHWOBJ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS00) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS05) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS10) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS15) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS20) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS25) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS30) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS35) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(CBRISS40) EXPLAIN(YES) ENABLE(CICS)
```



```

BIND PACKAGE(TPNS) MEMBER(EYPOFN01) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPOF200) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPOF201) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPOMD00) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPOMD01) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPOMV00) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPOMV01) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPORD00) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPORD01) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPORT00) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPORT01) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWAD ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWDR ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFL00) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFL01) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFN ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFN1 ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFN41) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFN42) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFN51) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWFN52) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWGT ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPWRD ) EXPLAIN(YES) ENABLE(CICS)
BIND PACKAGE(TPNS) MEMBER(EYPZCIO ) EXPLAIN(YES) ENABLE(CICS)

```

```

RUN PROGRAM(DSNTIAD) PLAN(DSNTIA41) LIB('DB2410.RUNLIB.LOAD') 00018306
END 00018400
//SYSIN DD * 00018500
GRANT EXECUTE ON PLAN TPNSO TO PUBLIC; 00018606
/* 00027000

```

A.6.2 Sample: DB2 Plan for IODM and IPFAF

```

//STEP1 EXEC PGM=IKJEFT01,DYNAMNBR=20,REGION=1024K 00010900
//SYSTSPRT DD SYSOUT=* 00011400
//SYSPRINT DD SYSOUT=* 00011500
//SYSUDUMP DD SYSOUT=* 00011600
//SYSTSIN DD * 00011700
DSN SYSTEM(DBD1) 00011806
BIND PLAN(TPNSO) OWNER(TPNS) QUALIFIER(TPNS) - 00011900
PKLIST(TPNS.*) - 00012000
ACTION(REPLACE) ISOLATION(CS) RETAIN CACHESIZE(0) - 00012100
ACQUIRE(USE) RELEASE(DEALLOCATE) VALIDATE(BIND) EXPLAIN(YES) - 00012200
CURRENTDATA(NO) 00012200
RUN PROGRAM(DSNTIAD) PLAN(DSNTIA41) LIB('DB2410.RUNLIB.LOAD') 00018306
END 00018400
//SYSIN DD * 00018500
GRANT EXECUTE ON PLAN TPNSO TO PUBLIC; 00018606
/* 00027000

```

Appendix B. CICS SIT Tables and SYSIN Definitions for IODM and IPFAF

B.1 IODM SIT Table

The same SIT table is used both of IODM CICS regions. The overrides to each IODM CICS regions are shown in Section B.1.1, "ODMD CICS Region SYSIN" on page 64 and Section B.1.2, "ODM2 CICS Region SYSIN" on page 64.

```

***** 00010017
* 00020017
* SAMPLE CICS SIT FOR OBJECT DISTRIBUTION MANAGER V2.2 00030017
* 00040017
* NOTE - XX NEEDS TO BE CHANGED TO THE CORRECT SUFFIX 00050017
* 00060017
* SOME PARAMETERS HAVE IODM SPECIFIC VALUES. THESE ARE THE ONES 00060017
* MARKED WITH '***' BEFORE THE COMMENTS. 00060017
* 00060017
* 00060017
***** 00070017
SIT10 TITLE 'DFHSIT10 - CICS SYSTEM INITIALIZATION TABLE' 00080017
* 00090000
          DFHSIT TYPE=CSECT, X05000000
              AKPFREQ=200,          ACTIVITY KEYPOINTING X07000000
              APPLID=CICSODMD,      APPLICATION NAME OF CICS SYSTEMX09000000
              BMS=(FULL,COLD,ALIGN,DDS), BASIC MAPPING SUPPORT X12000000
              CICSVC=218,          TYPE 2 SVC(SPARE) X14000000
              DATFORM=MMDDYY,      EXTERNAL DATE DISPLAY X16000000
              DBP=1$,              DYNAMIC BACKOUT PROGRAM X17000000
              DBUFSZ=1024,         BUFFER SIZE FOR DYN BACKOUT X18000000
              DCT=90,              *** TD AND IODM DESTINATIONS X20000000
              DIP=NO,              NO BATCH DATA INTERCHANGE X21000000
              DLI=NO,              NO DL/I SUPPORT X22000000
              DUMP=YES,            FULL DUMP X22000000
              DUMPDS=AUTO,         SWITH DUMPS AUTOMATICALLY X22000000
              FCT=NO,              *** VSAM FOR IODM BATCH PROCESSING X28000000
              FLDSEP=' ',          ALLOW LESS THAN 4 CHAR TRANIDS X28000000
              FLDSTRT='\',         FREE FROM INPUT FOR BUILD-IN X28000000
              GMTEXT='YOU ARE CONNECTED TO OBJ DIST MGR CICS', X28010008
              GRPLIST=ODMLIST,     CSD GROUP LIST NAME X32000009
              HPO=YES,             HIGH PERFORMANCE OPTION X32000009
              ICP=COLD,            INTERVAL CONTROL PGM X32000009
              ICV=100,             INTERVAL CONTROL EXIT TIME-MS X32000009
              ICVR=20000,          RUNAWAY TASK TIME X33000000
              ICVTS=100,          TERMINAL SCAN DELAY X35000000
              ISC=YES,              *** FOR REMOTE IODM AND FRONT-END X38000000
              JCT=90,              *** IODM JOURNALING X39010000
              LGNMSG=YES,          VTAM LOGON DATA AVAILABLE X39010000
              MCT=90,             MONITOR CONTROL PGM X44000000
              MSGLVL=1,           PRINT START-UP MSGS X44000000
              MXT=50,             MAX NO. OF ALL CONCURRENT TASKSX45000000
              PGCHAIN=X/,         BMS COMMAND - PAGE CHAINING X52000000
              PGCOPY=C/,          - PAGE COPY X53000000
              PGPURGE=T/,         - PAGE PURGE X54000000
              PGRET=P/,           - PAGE RETRIEVAL X55000000

```

```

PLTPI=I9,          *** LCB & MSG TABLE LOAD AT INIT  X
PLTSD=S9,          *** SHUT DOWN IODM                      X
PRINT=PA1,         REQUEST KEY FOR 3270 PRINTOUT    X60000000
RAPOOL=3,          FIXED RPL'S FIXED              X61000000
RENTPGM=NOPROTECT, USE CICS-KEY STORAGE           X
SEC=NO,            NO RACF SUPPORT                 X26000000
SECPRFX=NO,        NO PREFIX THE RESOURCE NAMES  X26000000
SPOOL=YES,         JES SPOOL SUPPORT             X63000000
SRBSVC=215,        X65000000
SRT=1$,            DEFAULT SRT                    X65000000
START=AUTO,        COLD OR WARM START            X66000000
STGPROT=NO,        NO STORAGE PROTECTION         X
SUFFIX=90,         *** SUFFIX FOR SIT                X
SYSIDNT=ODMD,     *** CICS SYSTEM ID                X70000000
TCP=YES,           TERMINAL CONTROL PROGRAM      X72000007
TCT=NO,           *** VTAM AND BTAM TERMINALS     X74000000
TD=(6,3),         BUFFER/STRINGS                                   X74000000
TS=(COLD,8,3),   BUFFERS,STRINGS                               X74000000
TST=NO,           HAS TEMP STORAGE TABLE INCLUDEDX84010000
TSMGSET=4,        4 MESSAGE SET ENTRIES                 X86000000
VTAM=YES,         VTAM ACCESS METHOD IS USED                   X86000000
WRKAREA=512,      COMMON WORK AREA OF THE CSA                 X90000000
XLT=NO,           NO TRANS LIST TABLE                       X92000000
XDCT=NO,          NO RACF FOR DCT ENTRIES                       X92000000
XJCT=NO,          NO RACF FOR JCT ENTRIES                       X92000000
XTST=NO,          NO RACF FOR TST ENTRIES                       X92000000
DUMMY=DUMMY      TO END MACRO                          98000000
END DFHSITBA                                           99000000

```

B.1.1 ODM D CICS Region SYSIN

```

SIT=90,
GRPLIST=TPNSODMD, INCLUDE DLI SAMPLE PROGRAMS & TRANSACTIONS
APPLID=CICSODMD,
SYSIDNT=ODMD,
INITPARM=(DSN2STRT='1,DBD1'),
MXT=80,
EDSALIM=500M,
XRF=NO,
SPOOL=YES,
DUMP=NO,
SRBSVC=215,
CICSSVC=216,
AUXTR=OFF,
GMTEXT=' Welcome to Poughkeepsie Parallel Sysplex System (PET)
        IBM SAA ImagePlus Object Distribution Manager MVS/ESA V2.2
        * * * * * T P N S Benchmark IODM Region #1 * * * * *'
.END

```

B.1.2 ODM2 CICS Region SYSIN

```

SIT=90,
GRPLIST=TPNSODM2, INCLUDE DLI SAMPLE PROGRAMS & TRANSACTIONS
APPLID=CICSODM2,
SYSIDNT=ODM2,
INITPARM=(DSN2STRT='1,DBD1'),
MXT=80,
EDSALIM=500M,
XRF=NO,
SPOOL=YES,

```

```
DUMP=NO,  
SRBSVC=215,  
CICSSVC=216,  
AUXTR=OFF,  
GMTEXT=' Welcome to Poughkeepsie Parallel Sysplex System (PET)  
        IBM SAA ImagePlus Object Distribution Manager MVS/ESA V2.2  
        * * * * * T P N S Benchmark IODM Region #2          * * * * *'  
.END
```

B.2 IPFAF SIT Table

The same SIT table is used both of IPFAF CICS regions. The overrides to each IPFAF CICS regions are shown in Section B.2.1, "FAFD CICS Region SYSIN" on page 68 and Section B.2.2, "FAF2 CICS Region SYSIN" on page 68.

```

SIT70      TITLE 'DFHSIT70 - CICS SAMPLE SYSTEM INITIALIZATION TABLE'      00010011
*****                                         00020001
*
* MODULE NAME = ENTMSIT                                         * 00030001
*
* DESCRIPTIVE NAME = ENTCICS SYSTEM INITIALIZATION TABLE FOR CICS v4 * 00040011
*
* ?? STRINGS DESIGNED FOR GLOBAL CHANGES VIA AN EDITOR        * 00050001
*
*
*      ENTCICS          CICS REGION NAME                            * 00060011
*      ENTG             CSD GROUPLIST                             * 00070001
*      ENTS             CICS SYSTEM IDENIFIER                     * 00080000
*      DUSER            DEFAULT USER                             * 00090000
*
* -----* 00100001
*
* CHANGE ACTIVITY :                                           * 00110001
* $MOD(ENTMSIT),COMP(STARTER),PROD(CICS/ESA):                 * 00120001
*
*      PN= REASON REL YMMDD HDXIII : REMARKS                    * 00130001
* -----* 00140001
*
*
***** 00150001
          DFHSIT TYPE=CSECT,                                     X00160001
              ADI=30,          DELAY BEFORE TAKE IS ATTEMPTED X00170001
              AIEEXIT=DFHZATDX, AUTO-INSTALL USER PROGRAM    X00180001
              AIQMAX=100,      AUTO-INSTALL MAXIMUM            X00190001
              AILDELAY=0,      AUTO-INSTALL LOGOFF DELAY       X00200001
              AIRDELAY=500,     AUTO-INSTALL RESTART DELAY      X00210001
              AKPFREQ=200,      ACTIVITY KEYPOINTING NEEDED    X00220001
              APPLID=cicsfafd,  APPL NAME OF CICS SYSTEM       X00230001
              AUXTR=OFF,        AUXILIARY TRACE OFF            X00240001
              AUXTRSW=NO,       NO AUTO SWITCH FOR AUX. TRACE  X00250001
              BMS=(FULL,COLD,ALIGN,DDS), FULL BASIC MAPPING SUPPORT X00260001
              CLSDSTP=NOTIFY,   REQUEST VTAM NOTIFY FOR PASS   X00270001
              CICS SVC=218,      CICS TYPE 3 SVC                X00280001
              CSDACC=READWRITE, CSD access                      X00290001
              CSDBKUP=STATIC,    Backuptype of CSD              X00300001
              CSDFRLOG=NO,       Journal id. for CSD forw. recovX00310001
              CSDJID=NO,         Journal id. for CSD auto. journX00320001
              CSDSTRNO=2,        CSD Number of strings          X00330001
              CSDLRNO=1,         VSAM LSR pool number for CSD   X00340001
              CSDRECOV=NONE,     CSD recoverable file option    X00350001
              DATFORM=MMDDYY,    EXTERNAL DATE DISPLAY          X00360001
              DBP=1$,            DYNAMIC BACKOUT PROGRAM        X00370001
              DBUFSZ=1024,       BUFFER SIZE FOR DYN BACKOUT    X00380001
              DCT=70,            TD AND OTHER DESTINATIONS      X00390001
              DFLTUSER=cicsuser,  DEFAULT USER                X00400011
              DIP=NO,            NO BATCH DATA INTERCHANGE     X00410001
              DLI=NO,            NO DL/I SUPPORT                 X00420001
              DUMP=NO,           DUMP SELECTION                  X00430001
              DUMPDS=AUTO,       AUTO CHOOSE DUMP DATASET       X00440001
              DUMPSW=NEXT,       NO AUTO SWICH OF DUMP DATASET X00450001

```

DURETRY=30,	RE-TRY SDUMPING FOR 30 SECONDS	X00560001
DTRPGM=DFHDYP,	DYNAMIC ROUTING PROGRAM	X00570001
FCT=NO,	SAMPLE VSAM FILE FILEA	X00640011
GMTEXT='Welcome to ImagePlus IPFAF Region',		X00650001
GNTRAN=CESN,	DEFAULT LOGON TRANSACTION	X00650001
GRPLIST=FAFLIST,	CSD GROUP LIST	X00660012
GTFTR=OFF,	DISABLE GEN. TRACE FACILITY	X00670001
ICP=COLD,	COLD START INTERVAL CONTROL	PGMX00680001
ICV=1000,	INTERVAL CONTROL EXIT TIME-MS	X00690001
ICVR=20000,	RUNAWAY TASK TIME	X00700001
ICVTS=0,	TERMINAL SCAN DELAY	X00710001
ISC=YES,	NO INTERSYSTEM COMMUNICATION	X00720001
JCT=70,	DUAL EXTENT JOURNALLING	X00740011
JESDI=30,	CEC FAILURE TIME DELAY	X00750001
LGNMSG=YES,	VTAM data available for appl	X00760001
LPA=NO,	NO CICS MANAGM MODULES FROM LPAX	X00760001
MCT=NO,	NO MONITORING	X00770001
MN=OFF,	NO MONITORING	X00780001
MNPER=OFF,	NO MONITORING	X00790001
MNEXC=OFF,	NO MONITORING	X00800001
MNEVE=OFF,	NO MONITORING	X00810001
MSGCASE=MIXED,	MIXED CASE MESSAGES	X00820001
MSGLVL=1,	PRINT START-UP MSGS	X00830001
MXT=32,	MAX NO. OF ALL CONCURRENT TASKS	X00840001
NATLANG=E,	NATIONAL LANGUAGE ENGLISH	X00850001
PARMERR=INTERACT,	ACTION ON PARAMETER ERRORS	X00860001
PDI=30,	DELAY BETWEEN HEARTBEAT & MSG	X00870001
PGCHAIN=X/,	BMS COMMAND - PAGE CHAINING	X00890001
PGCOPY=C/,	- PAGE COPY	X00900001
PGPURGE=T/,	- PAGE PURGE	X00910001
PGRET=P/,	- PAGE RETRIEVAL	X00920105
PLTPI=I7,	PLT INITIALIZATION ENTRIES	X00921005
PLTSD=S7,	PLT SHUTDOWN ENTRIES	X00922005
PRGLAY=0,	NO TERMINAL PAGE CLEAN UP	X00930001
PRINT=PAI,	REQUEST KEY FOR 3270 PRINTOUT	X00940001
PRTYAGE=1,	MULTIPLIER FOR TASK PRIORITY	X00950001
PVDELAY=30,	TIMEOUT VALUE FOR LUIT TABLE	X00960001
RENTPGM=PROTECT,	REENTRANT PGM WRITE PROTECTION	X00880001
RMTRAN=CSGM,	XRF CAPABLE TERMINAL TRANSACTN	X00970001
SEC=NO,	RACF SUPPORT	X00980001
SRBSVC=215,	Default type 4 svc for hpo	X00990001
SRT=1\$,	SYSTEM RECOVERY TABLE	X00990001
START=AUTO,	COLD START	X01000001
STARTER=YES,	ALLOWS \$ IN SUFFIX	X01010001
STGRVY=YES,	STORAGE RECOVERY FUNCTION	X01020001
STGPROT=YES,	STORAGE PROTECTION	X01030001
SUBTSKS=0,	EXTRA TCB'S REQUIRED	X01040001
SUFFIX=70,	STARTER SUFFIX	X01050011
SYSDNT=FAFD,	CICS SYTEM IDENTIFIER	X01051011
TAKEOVR=MANUAL,	CONFIRMATION OF TAKEOVER REQUD	X01060001
TCP=YES,	TERMINAL CONTROL PROGRAM	X01070001
TCT=NO,	VTAM TERMINALS ONLY	X01080013
TD=(3,3),	TRANSIENT DATA, 3 BUFS, 3 STRGSX	X01090001
INTR=ON,	INTERNAL TRACE ON	X01100001
TRTABSZ=64,	TRACE TABLE SIZE (KILOBYTES)	X01110001
TS=(,3,3),	FULL TS (3 BUFFERS, 3 STRINGS)	X01120001
TSMGSET=4,	4 MESSAGE SET ENTRIES	X01130001
TST=NO,	NO TEMP STORAGE TABLE INCLUDED	X01140001
VTAM=YES,	INITIALIZE FOR VTAM	X01170001

WRKAREA=512,	COMMON WORK AREA OF THE CSA	X01180001
XLT=NO,	TRANS LIST TABLE	X01190001
XRF=NO,	EXTENDED RECOVERY REQUIRED	X01200001
XDCT=NO,	DCT SECURITY CHECK	X01210016
XJCT=NO,	JCT SECURITY CHECK	X01210116
XPCT=NO,	TRAN SECURITY CHECK	X01211016
XPPT=NO,	TRAN SECURITY CHECK	X01211016
XPSB=NO,	TRAN SECURITY CHECK	X01211016
XTRAN=NO,	SECURITY CHECK ON TRANSACTION	X01220001
XTST=NO,	SECURITY CHECK ON TRANSACTION	01220001
END		01230001

B.2.1 FAFD CICS Region SYSIN

```

SIT=70,
GRPLIST=TPNSFAFD, INCLUDE DLI SAMPLE PROGRAMS & TRANSACTIONS
APPLID=CICSFADF,
SYSIDNT=FAFD,
INITPARM=(DSN2STRT='30,DBD1'),
MXT=80,
XRF=NO,
SPOOL=YES,
DUMP=NO,
SRBSVC=215,
CICSSVC=216,
AUXTR=OFF,
GMTEXT=' Welcome to Poughkeepsie Parallel Sysplex System (PET)
        IBM SAA ImagePlus Folder Application Facility MVS/ESA V2.2.1
        * * * * * TPNS Benchmark Region #1 * * * * *'
.END

```

B.2.2 FAF2 CICS Region SYSIN

```

SIT=70,
GRPLIST=TPNSFAF2, INCLUDE DLI SAMPLE PROGRAMS & TRANSACTIONS
APPLID=CICSF2AF,
SYSIDNT=FAF2,
INITPARM=(DSN2STRT='30,DBD1'),
MXT=80,
XRF=NO,
SPOOL=YES,
DUMP=NO,
SRBSVC=215,
CICSSVC=216,
AUXTR=OFF,
GMTEXT=' Welcome to Poughkeepsie Parallel Sysplex System (PET)
        IBM SAA ImagePlus Folder Application Facility MVS/ESA V2.2.1
        * * * * * TPNS Benchmark Region #2 * * * * *'
.END

```

Appendix C. IODM Exit: EKCCDDTE

The EKCCDDTE exit is enabled by installing PTF UN77405. The following sample program shows what we used for our configuration.

```
000100 IDENTIFICATION DIVISION.                                00010000
000200 PROGRAM-ID.      EKCCDDTE.                              00020000
000300*****                                                    00030000
000400* START OF SPECIFICATIONS                                00040000
000500*****                                                    00050000
000600* MODULE-NAME: EKCCDDTE                                  00060000
000700*                                                         00070000
000800* DESCRIPTIVE-NAME: DISPERSE DISPLAY TRANSACTIONS EXIT  00080000
000900*=====*                                               00090000
001000* STATUS: VERSION 2 RELEASE 2.0                          00100000
001100*                                                         00110000
001200* FUNCTION:                                             00120000
001300*                                                         00130000
001400*   THIS EXIT WILL ENABLE THE USER TO DISPERSE THEIR DISPLAY 00140000
001500*   REQUESTS OVER MULTIPLE TRANSACTIONS.                 00150000
001600*                                                         00160000
001700*=====*                                               00170000
001800* OPERATION:                                           00180000
001900*                                                         00190000
002000* 1) RETRIEVE DISPLAY TRANSACTION FROM THE FRONT-END APPLICATION.00200000
002100* 2) EXTRACT THE DESIRED FIELDS FROM THE DISPLAY COMMAND.  00210000
002200* 3) USE THE EXTRACTED FIELDS TO DETERMINE WHICH TRANSACTION 00220000
002300*   SHOULD BE ISSUED.                                     00230000
002400* 4) REISSUE THE DISPLAY REQUEST USING THE NEW TRANSACTION ID. 00240000
002500*                                                         00250000
002600*=====*                                               00260000
002700* RECOVERY OPERATION:                                    00270000
002800*   CICS ERRORS: WRITE OUT AN ERROR MESSAGE AND THE DISPLAY 00280000
002900*   COMMAND TO A TEMPORARY STORAGE QUEUE                 00290000
003000*                                                         00300000
003100*   CICS ABENDS: CICS WILL HANDLE ANY ABENDS              00310000
003200*                                                         00320000
003300*   NOTE: THE USER CAN CHANGE HOW THIS EXIT HANDLES CICS   00330000
003400*   ERRORS AND CICS ABENDS. THE ERROR ROUTINE PROVIDED    00340000
003500*   IN THIS EXIT IS ONLY A SAMPLE.                         00350000
003600*                                                         00360000
003700*=====*                                               00370000
003800* NOTES:                                               00380000
003900*                                                         00390000
004000* * ALL TRANSACTION IDS GENERATED BY THIS EXIT MUST BE DEFINED 00400000
004100*   IN THE CICS PROGRAM CONTROL TABLE (PCT). IF YOU DO NOT  00410000
004200*   DEFINE THESE TRANSACTION IDS IN THE CICS PCT, A TRANSIDERR 00420000
004300*   WILL OCCUR ON THE CICS START COMMAND.                  00430000
004400*                                                         00440000
004500* * ALSO, EACH NEW TRANSACTION ID MUST HAVE AN ENTRY IN THE DB200450000
004600*   RESOURCE CONTROL TABLE (RCT) WITH AN ASSOCIATED DB2 PLAN. 00460000
004700*   THE CURRENT DB2 PLAN FOR DISP TRANSACTIONS IS EKCPDISP.  00470000
004800*   YOU MAY EITHER USE THIS PLAN, OR CREATE NEW ONES TO LIMIT 00480000
004900*   THE NUMBER OF DB2 THREADS PER TRANSACTION.              00490000
005000*                                                         00500000
005100* * IF YOUR INSTALLATION IS USING A CUSTOM FRONT-END        00510000
005200*   APPLICATION, YOU MAY WANT TO CONSIDER CHANGING THE        00520000
005300*   TRANSACTION ID FOR THE DISPLAY COMMAND FROM THE FRONT-END 00530000
```

```

005400*      APPLICATION.  THAT WILL ELIMINATE THE NEED FOR THIS EXIT.  00540000
005500*
005600*      * SINCE THIS EXIT INTERCEPTS THE DISPLAY COMMAND BEFORE IT  00560000
005700*      REACHES IODM, IODM CAN NOT HANDLE ANY ERRORS THAT OCCUR IN  00570000
005800*      THIS EXIT.  THEREFORE, THIS EXIT MUST HANDLE ALL ERROR  00580000
005900*      SITUATIONS THAT OCCUR BEFORE THE DISPLAY COMMAND IS  00590000
006000*      RESTARTED WITH THE NEW TRANSACTION ID.  ONCE IODM GETS  00600000
006100*      CONTROL, IT WILL HANDLE ANY ERROR SITUATIONS.  00610000
006200*
006300*      * THE FUNCTION OF THIS EXIT IS TO ALLOW THE USER TO HAVE  00630000
006400*      MORE CONTROL OVER HOW IODM AND CICS PROCESS DISPLAY  00640000
006500*      COMMANDS.  HOWEVER, THE ULTIMATE GOAL OF THIS EXIT IS  00650000
006600*      TO ENSURE THAT THE DISPLAY COMMAND DOES GET PROCESSED  00660000
006700*      BY IODM.  THEREFORE, IF AN ERROR OCCURS IN THIS EXIT  00670000
006800*      OR YOU ARE UNABLE TO DETERMINE WHICH TRANSACTION TO  00680000
006900*      START, YOU SHOULD ALWAYS HAVE A DEFAULT TRANSACTION  00690000
007000*      WHICH YOU CAN USE TO PROCESS THE DISPLAY COMMAND.  00700000
007100*      THE ONLY TYPE OF ERROR THAT WOULD CAUSE YOU NOT USE  00710000
007200*      THE DEFAULT TRANSACTION ID AND PROCESS THE DISPLAY  00720000
007300*      COMMAND IS A CICS ERROR.  IN THIS CASE YOU SHOULD  00730000
007400*      PROBABLY LOG THE ERROR AND TERMINATE PROCESSING.  00740000
007500*
007600**=====*00760000
007700* DEPENDENCIES:  00770000
007800*
007900*      * THE FRONT-END APPLICATION THAT IS BEING USED, AND WHETHER  00790000
008000*      OR NOT THE RETURN TRANSACTION ID OPTION IS UTILIZED.  REFER  00800000
008100*      TO PARAGRAPHS 300-RETRIEVE-COMMAND AND 600-START-DISPLAY  00810000
008200*      FOR MORE DETAILS ON THIS DEPENDENCY.  00820000
008300*
008400* RESTRICTIONS:  00840000
008500*
008600*      * THIS EXIT SHOULD ONLY BE USED TO INTERCEPT DISPLAY COMMANDS  00860000
008700*      COMING FROM THE FRONT-END APPLICATION.  ALL OTHER IODM  00870000
008800*      COMMANDS SHOULD BE PROCESSED IN THE NORMAL WAY.  00880000
008900*
009000*
009100* MODULE TYPE: PROGRAM  00910000
009200*
009300* LANGUAGE:      COBOL II.  00930000
009400*
009500* ATTRIBUTES:  AMODE(31) RMODE(ANY) DATA(31)  00950000
009600*
009700**=====*00970000
009800* CALLING INTERFACE:  00980000
009900*
010000*      * REFER TO CHAPTER 9, "DISPLAYING OBJECTS", IN THE IODM  01000000
010100*      APPLICATION PROGRAMMER'S GUIDE FOR FRONT-END APPLICATIONS  01010000
010200*      (SC31-7540-02) ON HOW TO ISSUE DISPLAY COMMANDS TO IODM  01020000
010300*      FROM EITHER AN IMS OR CICS FRONT-END APPLICATION.  01030000
010400*
010500**=====*01050000
010600* PARAMETERS:  01060000
010700*
010800*      NONE  01080000
010900*
011000* FILES/QUEUES-USED:  01100000
011100*
011200*      QUEUE - EKCCDDTE - USED TO LOG ANY ERRORS THAT OCCUR WHILE  01120000

```

```

011300*          THIS EXIT IS EXECUTING.  THIS QUEUE IS          01130000
011400*          JUST A SAMPLE.  IF YOU DESIRE, YOU CAN        01140000
011500*          CHANGE THE NAME OF THIS QUEUE OR THE          01150000
011600*          METHOD THIS EXIT USES TO LOG ERROR            01160000
011700*          MESSAGES.                                     01170000
011800*                                                    01180000
011900* CONTROL-BLOCKS:                                       01190000
012000*                                                    01200000
012100*     INSTALLATION DEPENDENT                             01210000
012200*                                                    01220000
012300**=====*01230000
012400* RETCODE (RETURN CODES):                               01240000
012500*                                                    01250000
012600*     INSTALLATION DEPENDENT                             01260000
012700*                                                    01270000
012800* REACODE (REASON CODES):                               01280000
012900*                                                    01290000
013000*     INSTALLATION DEPENDENT                             01300000
013100*                                                    01310000
013200* MESSAGES:                                           01320000
013300*                                                    01330000
013400*     USR0001A - INDICATES THAT A CICS ERROR OCCURRED.  01340000
013500*     THAT DETECTED THE ERROR ALONG WITH THE CICS EIBFN  01350000
013600*     AND EIBRCODE FIELDS ARE DISPLAYED IN THE MESSAGE.  01360000
013700*     THIS IS JUST A SAMPLE MESSAGE.  YOU CAN CHANGE THE  01370000
013800*     MESSAGE ID OR THE CONTENT OF THE MESSAGE TO MEET  01380000
013900*     YOUR NEEDS.                                        01390000
014000*                                                    01400000
014100* ABEND-CODES:                                         01410000
014200*                                                    01420000
014300*     INSTALLATION DEPENDENT                             01430000
014400*                                                    01440000
014500**=====*01450000
014600* EXIT-NORMAL:                                         01460000
014700*                                                    01470000
014800*     * EXIT PROGRAM                                     01480000
014900*                                                    01490000
015000* EXIT-ERROR:                                          01500000
015100*                                                    01510000
015200*     * WRITES OUT AN ERROR MESSAGE AND THE DISPLAY COMMAND TO  01520000
015300*     A TEMPORARY STORAGE QUEUE (EKCCDDTE)                01530000
015400*     * EXIT PROGRAM                                     01540000
015500*                                                    01550000
015600* SUBROUTINES:                                         01560000
015700*                                                    01570000
015800*     INSTALLATION DEPENDENT                             01580000
015900*                                                    01590000
016000***END OF SPECIFICATIONS*****01600000
016100/                                                    01610000
016200 ENVIRONMENT DIVISION.                                  01620000
016300 CONFIGURATION SECTION.                                01630000
016400 SOURCE-COMPUTER.  IBM-370-SERIES.                    01640000
016500     EJECT                                             01650000
016600*****01660000
016700*           DATA DIVISION * 01670000
016800*****01680000
016900 DATA DIVISION.                                       01690000
017000*****01700000
017100*           WORKING STORAGE SECTION * 01710000

```

```

017200***** 01720000
017300 WORKING-STORAGE SECTION. 01730000
017400 01 WORK-EYECATCHER. 01740000
017500 05 WORK-WSBEGIN PIC X(29) VALUE 01750000
017600 '***** BEGINNING OF WS *****'. 01760000
017700 05 WORK-MODNAME PIC X(08) VALUE 'EKCCDDTE'. 01770000
017800 05 WORK-MODVERS PIC X(04) VALUE 'V220'. 01780000
017900 05 WORK-WHENCOMP PIC X(20) VALUE SPACES. 01790000
018000 01800000
018100***** 01810000
018200* GENERAL WORK AREA * 01820000
018300***** 01830000
018400 01840000
018500 01 WORK-VARIABLES. 01850000
018600 05 WORK-EIBFN PIC X(02) VALUE SPACES. 01860000
018700 05 WORK-EIBRCODE PIC X(06) VALUE SPACES. 01870000
018800 01880000
018900 05 WORK-GDS-LENGTH PIC S9(04) COMP VALUE 0. 01890000
019000 05 WORK-RTERMID PIC X(04) VALUE SPACES. 01900000
019100 05 WORK-RTRANSID PIC X(04) VALUE SPACES. 01910000
019200 05 WORK-TRANSID PIC X(04) VALUE SPACES. 01920000
019300 05 WORK-QUEUE PIC X(08) VALUE 'EKCCDDTE'. 01930000
019400 05 WORK-INDEX PIC S9(04) COMP VALUE 0. 01940000
019500 01950000
019600 05 WORK-DISPLAY-RECEIVED-FLAG PIC X(01) VALUE 'N'. 01960000
019700 88 WORK-DISPLAY-RECEIVED VALUE 'Y'. 01970000
019800 88 WORK-NOT-RECEIVED VALUE 'N'. 01980000
019900 05 WORK-FOUND-FLAG PIC X(01) VALUE 'N'. 01990000
020000 88 WORK-FOUND VALUE 'Y'. 02000000
020100 88 WORK-NOT-FOUND VALUE 'N'. 02010000
020200 05 WORK-FIXED-AREA PIC X(01) VALUE 'B'. 02020000
020300 88 WORK-BEGIN-OF-FIXED VALUE 'B'. 02030000
020400 88 WORK-END-OF-FIXED VALUE 'E'. 02040000
020500 05 WORK-OBJECT-LOCATION PIC X(01) VALUE 'L'. 02050000
020600 88 WORK-LOCAL-OBJECT VALUE 'L'. 02060000
020700 88 WORK-REMOTE-OBJECT VALUE 'R'. 02070000
020800 05 WORK-TSYSID-FLAG PIC X(01) VALUE 'N'. 02080000
020900 88 WORK-TSYSID-SET VALUE 'Y'. 02090000
021000 88 WORK-TSYSID-NOT-SET VALUE 'N'. 02100000
021100 05 WORK-APUSERID-FLAG PIC X(01) VALUE 'N'. 02110000
021200 88 WORK-APUSERID-SET VALUE 'Y'. 02120000
021300 88 WORK-APUSERID-NOT-SET VALUE 'N'. 02130000
021400 05 WORK-UTERMID-FLAG PIC X(01) VALUE 'N'. 02140000
021500 88 WORK-UTERMID-SET VALUE 'Y'. 02150000
021600 88 WORK-UTERMID-NOT-SET VALUE 'N'. 02160000
021700 05 WORK-OBJDEFIN-FLAG PIC X(01) VALUE 'N'. 02170000
021800 88 WORK-OBJDEFIN-SET VALUE 'Y'. 02180000
021900 88 WORK-OBJDEFIN-NOT-SET VALUE 'N'. 02190000
022000 05 WORK-WKSTATID-FLAG PIC X(01) VALUE 'N'. 02200000
022100 88 WORK-WKSTATID-SET VALUE 'Y'. 02210000
022200 88 WORK-WKSTATID-NOT-SET VALUE 'N'. 02220000
022300 02230000
022400***** 02240000
022500* CONSTANT WORK AREA * 02250000
022600***** 02260000
022700 02270000
022800 01 CONSTANTS. 02280000
022900 02290000
023000*/* THESE CONSTANT VARIABLES CONTAIN THE GDS IDENTIFIERS 02300000

```

023100*	OF IODM PARAMETERS AND GDS STRUCTURES. YOU CAN OBTAIN		02310000
023200*	THE GDS IDS FOR IODM PARAMETERS IN THE IODM APPLICATION		02320000
023300*	PROGRAMMER'S GUIDE FOR FRONT-END APPLICATIONS. */		02330000
023400			02340000
023500	05 CNST-TSYSID-GDS-ID	PIC X(02) VALUE X'0005'.	02350000
023600	05 CNST-OSYSID-GDS-ID	PIC X(02) VALUE X'0006'.	02360000
023700	05 CNST-OBJNAME-GDS-ID	PIC X(02) VALUE X'0007'.	02370000
023800	05 CNST-APUSERID-GDS-ID	PIC X(02) VALUE X'000C'.	02380000
023900	05 CNST-UTERMID-GDS-ID	PIC X(02) VALUE X'000E'.	02390000
024000	05 CNST-OBJDEFIN-GDS-ID	PIC X(02) VALUE X'0014'.	02400000
024100	05 CNST-COLLNAME-GDS-ID	PIC X(02) VALUE X'003E'.	02410000
024200	05 CNST-WKSTATID-GDS-ID	PIC X(02) VALUE X'0045'.	02420000
024300	05 CNST-OBJLIST-GDS-ID	PIC X(02) VALUE X'00FC'.	02430000
024400			02440000
024500*/*	THESE CONSTANT VARIABLES ARE SAMPLES ON HOW YOU CAN DISPERSE		02450000
024600*	DISPLAY REQUESTS IN REGARDS TO USERS OR GROUPS OF USERS.		02460000
024700*	THESE CONSTANTS ARE USER SPECIFIED. */		02470000
024800			02480000
024900	05 CNST-APUSERID-ACCOUNTING	PIC X(08) VALUE 'ACCOUNT'.	02490000
025000	05 CNST-APUSERID-RESEARCH	PIC X(08) VALUE 'RESEARCH'.	02500000
025100	05 CNST-APUSERID-ADMIN	PIC X(08) VALUE 'ADMIN'.	02510000
025200			02520000
025300*/*	THESE CONSTANT VARIABLES CONTAIN THE IDENTIFIERS OF THE NEW		02530000
025400*	TRANSACTIONS THAT WILL BE USED TO REISSUE THE DISPLAY		02540000
025500*	REQUEST. THESE CONSTANTS ARE USER SPECIFIED. */		02550000
025600			02560000
025700	05 CNST-TRANSID-ACCOUNTING	PIC X(04) VALUE '????'.	02570000
025800	05 CNST-TRANSID-RESEARCH	PIC X(04) VALUE '????'.	02580000
025900	05 CNST-TRANSID-ADMIN	PIC X(04) VALUE '????'.	02590000
026000	05 CNST-TRANSID-LARGE-REQUEST	PIC X(04) VALUE '????'.	02600000
026100	05 CNST-TRANSID-SMALL-REQUEST	PIC X(04) VALUE '????'.	02610000
026200	05 CNST-TRANSID-LOCAL	PIC X(04) VALUE '????'.	02620000
026300	05 CNST-TRANSID-REMOTE	PIC X(04) VALUE '????'.	02630000
026400	05 CNST-TRANSID-DASD	PIC X(04) VALUE '????'.	02640000
026500	05 CNST-TRANSID-LIBRARY	PIC X(04) VALUE '????'.	02650000
026600	05 CNST-TRANSID-SHELF	PIC X(04) VALUE '????'.	02660000
026700	05 CNST-TRANSID-DEFAULT	PIC X(04) VALUE 'DISD'.	02670000
026800			02680000
026900	*****		02690000
027000*	ERROR MESSAGE(S) WORK AREA		* 02700000
027100	*****		02710000
027200			02720000
027300 01	WORK-CICS-ERR-MSG.		02730000
027400	05 FILLER	PIC X(32)	02740000
027500	VALUE 'USR0001A CICS ERROR OCCURRED -- '.		02750000
027600	05 FILLER	PIC X(24)	02760000
027700	VALUE 'MODERR: EKCCDDTE EIBFN: '.		02770000
027800	05 WORK-DISPLAY-EIBFN	PIC X(04) VALUE SPACES.	02780000
027900	05 FILLER	PIC X(11)	02790000
028000	VALUE ' EIBRCODE: '.		02800000
028100	05 WORK-DISPLAY-EIBRCODE	PIC X(12) VALUE SPACES.	02810000
028200			02820000
028300	*****		02830000
028400*	HEX CONVERSION ROUTINE WORK AREA		* 02840000
028500	*****		02850000
028600			02860000
028700 01	WORK-HEX-CONVERSION-VARIABLES.		02870000
028800	05 WORK-DIVIDEND	PIC 9(09) COMP VALUE ZEROS.	02880000
028900	05 WORK-DIVIDEND-CHAR REDEFINES WORK-DIVIDEND.		02890000

```

029000      10 FILLER                                PIC X(02).                02900000
029100      10 WORK-DIVIDEND-CHAR-2-BYTES          PIC X(02).                02910000
029200      05 WORK-DIVISOR                        PIC 9(04) COMP VALUE ZEROS. 02920000
029300      05 WORK-QUOTIENT                      PIC 9(04) COMP VALUE ZEROS. 02930000
029400      05 WORK-REMAINDER                    PIC 9(04) COMP VALUE ZEROS. 02940000
029500      05 WORK-NUM-HALFWORDS                PIC 9(09) COMP VALUE ZEROS. 02950000
029600      05 WORK-HEX-FIELD-SUB                PIC 9(09) COMP VALUE ZEROS. 02960000
029700      05 WORK-HALFWORD-SUB                PIC 9(09) COMP VALUE ZEROS. 02970000
029800      05 WORK-HEX-TABLE.                  02980000
029900      10 FILLER                                PIC X(16) VALUE          02990000
030000      '0123456789ABCDEF'.                  03000000
030100      05 FILLER REDEFINES WORK-HEX-TABLE.  03010000
030200      10 WORK-HEX-VALUE                    PIC X(01)                03020000
030300      OCCURS 16 TIMES.                    03030000
030400      05 WORK-HEX-FIELD.                  03040000
030500      10 WORK-HEX-NUMBER                  PIC X(01)                03050000
030600      OCCURS 16 TIMES.                    03060000
030700      05 WORK-FIELD-TO-CONV.              03070000
030800      10 WORK-HALFWORD-TO-CONV          PIC X(02)                03080000
030900      OCCURS 4 TIMES.                    03090000
031000      03100000
031100***** 03110000
031200*  EXTRACT PROCESS WORK AREA *            03120000
031300***** 03130000
031400      03140000
031500 01  WORK-EXTRACT-PROCESS-VARIABLES.      03150000
031600*/*  FIXED LIST AREA FIELDS. */          03160000
031700      05 WORK-TSYSID-VALUE                PIC X(04) VALUE SPACES.  03170000
031800      05 WORK-APUSERID-VALUE             PIC X(08) VALUE SPACES.  03180000
031900      05 WORK-UTERMID-VALUE              PIC X(08) VALUE SPACES.  03190000
032000      05 WORK-WKSTATID-VALUE             PIC X(08) VALUE SPACES.  03200000
032100      05 WORK-OBJDEFIN-VALUE             PIC S9(04) COMP VALUE 0. 03210000
032200*/*  OBJECT LIST AREA FIELDS. */          03220000
032300      05 WORK-MAX-OBJECTS                 PIC S9(04) COMP VALUE 15. 03230000
032400      05 WORK-OSYSID.                    03240000
032500      10 WORK-OSYSID-COUNT                PIC S9(04) COMP VALUE 0. 03250000
032600      10 WORK-OSYSID-VALUE              PIC X(04) VALUE SPACES  03260000
032700      OCCURS 15 TIMES.                    03270000
032800      05 WORK-OBJNAME.                    03280000
032900      10 WORK-OBJNAME-COUNT              PIC S9(04) COMP VALUE 0. 03290000
033000      10 WORK-OBJNAME-VALUE              PIC X(40) VALUE SPACES  03300000
033100      OCCURS 15 TIMES.                    03310000
033200      05 WORK-COLLNAME.                  03320000
033300      10 WORK-COLLNAME-COUNT            PIC S9(04) COMP VALUE 0. 03330000
033400      10 WORK-COLLNAME-VALUE            PIC X(44) VALUE SPACES  03340000
033500      OCCURS 15 TIMES.                    03350000
033600*/*  FIELD EXTRACTION VARIABLES. */      03360000
033700      05 WORK-EXTRACT-GDS-ID             PIC X(02) VALUE SPACES.  03370000
033800      05 WORK-EXTRACT-VALUE              PIC X(128) VALUE SPACES. 03380000
033900      05 WORK-EXTRACT-CHAR-REDEF REDEFINES WORK-EXTRACT-VALUE. 03390000
034000      10 WORK-EXTRACT-CHAR-VALUE         PIC X(08).                03400000
034100      10 FILLER                            PIC X(120).              03410000
034200      05 WORK-EXTRACT-NUM-REDEF REDEFINES WORK-EXTRACT-VALUE. 03420000
034300      10 WORK-EXTRACT-NUM-VALUE         PIC S9(04) COMP.         03430000
034400      10 FILLER                            PIC X(126).              03440000
034500      03450000
034600***** 03460000
034700*  GDS WORK AREAS *                      03470000
034800***** 03480000

```

034900				03490000
035000	01	GDS-PARAMETER	PIC X(04).	03500000
035100	01	GDS-PARM-REDEF REDEFINES GDS-PARAMETER.		03510000
035200	05	GDS-ID	PIC X(02).	03520000
035300	05	GDS-PARM-LENGTH	PIC S9(04) COMP.	03530000
035400				03540000
035500	01	GDS-VARIABLES.		03550000
035600	05	GDS-INDEX	PIC S9(04) COMP.	03560000
035700				03570000
035800	01	GDS-BUFFER.		03580000
035900	05	GDS-CMDCODE	PIC X(04).	03590000
036000	05	GDS-CMD-LENGTH	PIC S9(09) COMP.	03600000
036100	05	FILLER	PIC X(04).	03610000
036200	05	GDS-PARAMETERS	PIC X(1788).	03620000
036300				03630000
036400				03640000
036500	01	WORK-ENDOFWS	PIC X(23) VALUE	03650000
036600		'***** END OF WS *****'.		03660000
036700		*****		03670000
036800*		END OF WORKING STORAGE		*03680000
036900		*****		*03690000
037000		EJECT		03700000
037100		*****		03710000
037200*		LINKAGE SECTION		* 03720000
037300		*****		03730000
037400		LINKAGE SECTION.		03740000
037500				03750000
037600		*****		03760000
037700*		PROCEDURE DIVISION		* 03770000
037800		*****		03780000
037900		PROCEDURE DIVISION.		03790000
038000				03800000
038100		100-MAINLINE.		03810000
038200				03820000
038300*/*		PREPARE MODULE FOR PROCESSING. */		03830000
038400		PERFORM 200-INIT THRU		03840000
038500		200-INIT-EXIT.		03850000
038600				03860000
038700*/*		RETRIEVE DISPLAY COMMAND FROM THE FRONT-END APPLICATION. */		03870000
038800		PERFORM 300-RETRIEVE-COMMAND THRU		03880000
038900		300-RETRIEVE-COMMAND-EXIT.		03890000
039000				03900000
039100*/*		EXTRACT SPECIFIED FIELDS FROM THE DISPLAY COMMAND. */		03910000
039200		PERFORM 400-EXTRACT-FIELDS THRU		03920000
039300		400-EXTRACT-FIELDS-EXIT.		03930000
039400				03940000
039500*/*		DETERMINE WHICH NEW TRANSACTION TO USE. */		03950000
039600		PERFORM 500-GENERATE-TRAN THRU		03960000
039700		500-GENERATE-TRAN-EXIT.		03970000
039800				03980000
039900*/*		REISSUE THE DISPLAY COMMAND USING THE NEW TRANSACTION. */		03990000
040000		PERFORM 600-START-DISPLAY THRU		04000000
040100		600-START-DISPLAY-EXIT.		04010000
040200				04020000
040300*/*		TERMINATE PROCESSING. */		04030000
040400		PERFORM 900-RETURN THRU		04040000
040500		900-RETURN-EXIT.		04050000
040600				04060000
040700		100-MAINLINE-EXIT.		04070000

040800	EXIT.	04080000
040900	EJECT	04090000
041000		04100000
041100		04110000
041200	200-INIT.	04120000
041300	*/ * THIS PARAGRAPH WILL CONTAIN ANY MODULE PREPARATION STEPS. */	04130000
041400		04140000
041500	*/ * ALL CICS ERRORS WILL BE PROCESSED IN PARAGRAPH	04150000
041600	* 800-CICS-ERROR. */	04160000
041700	EXEC CICS HANDLE CONDITION	04170000
041800	ERROR (800-CICS-ERROR)	04180000
041900	END-EXEC.	04190000
042000		04200000
042100	MOVE WHEN-COMPILED TO WORK-WHENCOMP.	04210000
042200		04220000
042300	200-INIT-EXIT.	04230000
042400	EXIT.	04240000
042500		04250000
042600	300-RETRIEVE-COMMAND.	04260000
042700	*/ * THIS PARAGRAPH WILL RETRIEVE THE DISPLAY COMMAND AND THE	04270000
042800	* RETURN TRANSACTION ID FROM THE FRONT-END APPLICATION. */	04280000
042900		04290000
043800	*/ * SET THE MAXIMUM LENGTH ALLOWED FOR THE GDS DATA. */	04300000
043900	MOVE LENGTH OF GDS-BUFFER TO WORK-GDS-LENGTH.	04310000
044900		04320000
045000	*/ * IF YOUR INSTALLATION IS USING IPFAF/API CICS, USE THE	04330000
045100	* FOLLOWING RETRIEVE COMMAND AND DELETE THE OTHER TWO RETRIEVE	04340000
045200	* COMMANDS IN THIS PARAGRAPH. */	04350000
045300	EXEC CICS RETRIEVE	04360000
045400	INTO (GDS-BUFFER)	04370000
045500	LENGTH (WORK-GDS-LENGTH)	04380000
045600	RTRANSID (WORK-RTRANSID)	04390000
045700	END-EXEC.	04400000
045800		04410000
049100	*/ * SET FLAG TO INDICATE THAT THE DISPLAY COMMAND	04420000
049200	* WAS RETRIEVED. */	04430000
049300	SET WORK-DISPLAY-RECEIVED TO TRUE.	04440000
049400		04450000
049500	300-RETRIEVE-COMMAND-EXIT.	04460000
049600	EXIT.	04470000
049700		04480000
049800	400-EXTRACT-FIELDS.	04490000
049900	*/ * THIS PARAGRAPH WILL EXTRACT ALL THE REQUIRED INFORMATION	04500000
050000	* FROM THE DISPLAY COMMAND TO PROPERLY DETERMINE WHICH	04510000
050100	* TRANSACTION SHOULD BE USED TO RESTART THE DISPLAY COMMAND	04520000
050200	* REQUEST. */	04530000
050300		04540000
050400	*/ * EXTRACT TSYSID FIELD FROM DISPLAY COMMAND. */	04550000
050500	MOVE CNST-TSYSID-GDS-ID TO WORK-EXTRACT-GDS-ID.	04560000
050600	PERFORM 410-EXTRACT-FIXED-FIELD THRU	04570000
050700	410-EXTRACT-FIXED-FIELD-EXIT.	04580000
050800	IF WORK-FOUND THEN	04590000
050900	MOVE WORK-EXTRACT-CHAR-VALUE TO WORK-TSYSID-VALUE	04600000
051000	SET WORK-TSYSID-SET TO TRUE	04610000
051100	END-IF.	04620000
051200		04630000
054900	*/ * EXTRACT THE OBJECT LIST AREA FIELDS FROM THE DISPLAY	04640000
055000	* COMMAND. */	04650000
055100	PERFORM 420-EXTRACT-OBJECT-FIELDS THRU	04660000

055200	420-EXTRACT-OBJECT-FIELDS-EXIT.	04670000
055300		04680000
055400*/*	VALIDATE THE FIELDS EXTRACTED FROM THE DISPLAY COMMAND. */	04690000
055500*	PERFORM 430-VALIDATE-FIELDS THRU	04700000
055600*	430-VALIDATE-FIELDS-EXIT.	04710000
055700		04720000
055800	400-EXTRACT-FIELDS-EXIT.	04730000
055900	EXIT.	04740000
056000		04750000
056100	410-EXTRACT-FIXED-FIELD.	04760000
056200*/*	THIS PARAGRAPH WILL EXTRACT THE PARAMETER VALUE FOR THE	04770000
056300*	SPECIFIED FIELD FROM THE DISPLAY GDS DATA. IF THE FIELD	04780000
056400*	IS NOT PRESENT IN THE GDS DATA, THE NOT-FOUND FLAG WILL	04790000
056500*	BE SET TO TRUE AFTER THIS PARAGRAPH TERMINATES. THIS	04800000
056600*	PARAGRAPH WILL ONLY SEARCH IN THE FIXED LIST AREA OF THE	04810000
056700*	DISPLAY COMMAND FOR THE SPECIFIED PARAMETER. */	04820000
056800		04830000
056900*/*	INITIALIZE FLAGS AND DATA AREA. */	04840000
057000	SET WORK-NOT-FOUND TO TRUE.	04850000
057100	SET WORK-BEGIN-OF-FIXED TO TRUE.	04860000
057200	INITIALIZE WORK-EXTRACT-VALUE.	04870000
057300		04880000
057400*/*	INITIALIZE THE GDS INDEX TO 13 TO BY-PASS THE 12 BYTE	04890000
057500*	GDS COMMAND HEADER INFORMATION. */	04900000
057600	MOVE 13 TO GDS-INDEX.	04910000
057700		04920000
057800*/*	PARSE THROUGH THE DISPLAY COMMAND UNTIL THE SPECIFIED	04930000
057900*	FIELD IS FOUND, END OF THE FIXED LIST AREA FOR THE DISPLAY	04940000
058000*	COMMAND IS REACHED, OR THE END OF THE DISPLAY COMMAND	04950000
058100*	IS ENCOUNTERED. */	04960000
058200	PERFORM	04970000
058300	UNTIL (WORK-FOUND OR	04980000
058400	WORK-END-OF-FIXED OR	04990000
058500	(GDS-INDEX >= WORK-GDS-LENGTH))	05000000
058600		05010000
058700*/*	GET NEXT GDS ID AND PARAMETER LENGTH. */	05020000
058800	MOVE GDS-BUFFER(GDS-INDEX:4) TO GDS-PARAMETER	05030000
058900		05040000
059000*/*	DETERMINE IF THE CURRENT PARAMETER IS THE SPECIFIED	05050000
059100*	PARAMETER OR THE START OF THE OBJECT LIST AREA. */	05060000
059200	EVALUATE GDS-ID	05070000
059300	WHEN WORK-EXTRACT-GDS-ID	05080000
059400		05090000
059500*/*	ADVANCE INDEX TO START OF PARAMETER DATA AREA,	05100000
059600*	EXTRACT THE PARAMETER VALUE, AND SET FOUND FLAG	05110000
059700*	TO TRUE. */	05120000
059800	ADD 4 TO GDS-INDEX	05130000
059900	MOVE GDS-BUFFER(GDS-INDEX:GDS-PARM-LENGTH) TO	05140000
060000	WORK-EXTRACT-VALUE(1:GDS-PARM-LENGTH)	05150000
060100	SET WORK-FOUND TO TRUE	05160000
060200		05170000
060300	WHEN CNST-OBJLIST-GDS-ID	05180000
060400		05190000
060500*/*	THE FIELD WAS NOT FOUND. SET END OF FIXED	05200000
060600*	AREA TO TRUE. */	05210000
060700	SET WORK-END-OF-FIXED TO TRUE	05220000
060800		05230000
060900	WHEN OTHER	05240000
061000		05250000

061100*/*	ADVANCE TO NEXT PARAMETER. */	05260000
061200	COMPUTE GDS-INDEX = GDS-INDEX +	05270000
061300	GDS-PARM-LENGTH + 4	05280000
061400	END-EVALUATE	05290000
061500	END-PERFORM.	05300000
061600		05310000
061700	410-EXTRACT-FIXED-FIELD-EXIT.	05320000
061800	EXIT.	05330000
061900		05340000
062000	420-EXTRACT-OBJECT-FIELDS.	05350000
062100*/*	THIS PARAGRAPH WILL EXTRACT ALL OF THE OBJECT LIST AREA	05360000
062200*	FIELDS SPECIFIED IN THE EVALUATE STATEMENT. THE SEARCH	05370000
062300*	WILL STOP WHEN THE END OF THE GDS DATA IS REACHED.	05380000
062400*	A SEPARATE COUNTER SHOULD BE USED TO KEEP TRACK OF HOW	05390000
062500*	MANY TIMES A PARAMETER WAS FOUND IN THE GDS DATA. */	05400000
062600		05410000
062700*/*	INITIALIZE THE GDS INDEX TO 13 TO BY-PASS THE 12 BYTE	05420000
062800*	GDS COMMAND HEADER INFORMATION. */	05430000
062900	MOVE 13 TO GDS-INDEX.	05440000
063000		05450000
063100*/*	PARSE THROUGH THE DISPLAY COMMAND UNTIL THE END OF THE	05460000
063200*	DISPLAY COMMAND IS ENCOUNTERED. */	05470000
063300	PERFORM	05480000
063400	UNTIL (GDS-INDEX >= WORK-GDS-LENGTH)	05490000
063500		05500000
063600*/*	GET NEXT GDS ID AND PARAMETER LENGTH. */	05510000
063700	MOVE GDS-BUFFER(GDS-INDEX:4) TO GDS-PARAMETER	05520000
063800		05530000
063900*/*	CHECK TO SEE IF THE PARAMETER IS ONE THAT SHOULD BE	05540000
064000*	SAVED. IF NOT, GO TO THE NEXT PARAMETER IN THE GDS	05550000
064100*	DATA. */	05560000
064200	EVALUATE GDS-ID	05570000
064300	WHEN CNST-OSYSID-GDS-ID	05580000
064400		05590000
064500*/*	IF MAXIMUM NUMBER OF OBJECTS IS NOT REACHED,	05600000
064600*	ADVANCE INDEX TO START OF PARAMETER DATA AREA,	05610000
064700*	INCREMENT THE OSYSID COUNTER, AND EXTRACT THE	05620000
064800*	DATA. */	05630000
064900	IF (WORK-OSYSID-COUNT < WORK-MAX-OBJECTS) THEN	05640000
065000	ADD 4 TO GDS-INDEX	05650000
065100	ADD 1 TO WORK-OSYSID-COUNT	05660000
065200	MOVE GDS-BUFFER(GDS-INDEX:GDS-PARM-LENGTH) TO	05670000
065300	WORK-OSYSID-VALUE(WORK-OSYSID-COUNT)	05680000
008090	IF WORK-TSYSID-SET	05690000
008100	MOVE WORK-TSYSID-VALUE TO	05700000
008110	GDS-BUFFER(GDS-INDEX:GDS-PARM-LENGTH)	05710000
008120	END-IF	05720000
065400	ADD GDS-PARM-LENGTH TO GDS-INDEX	05730000
065500	ELSE	05740000
065600	COMPUTE GDS-INDEX = GDS-INDEX +	05750000
065700	GDS-PARM-LENGTH + 4	05760000
065800	END-IF	05770000
065900		05780000
069400	WHEN OTHER	05790000
069500		05800000
069600*/*	ADVANCE TO NEXT PARAMETER. */	05810000
069700	COMPUTE GDS-INDEX = GDS-INDEX +	05820000
069800	GDS-PARM-LENGTH + 4	05830000
069900	END-EVALUATE	05840000

```

070000          END-PERFORM.                                05850000
070100                                                  05860000
070200 420-EXTRACT-OBJECT-FIELDS-EXIT.                    05870000
070300          EXIT.                                      05880000
070400                                                  05890000
070500 430-VALIDATE-FIELDS.                               05900000
070600*/* THIS PARAGRAPH SHOULD BE USED TO VALIDATE THE FIELDS AND 05910000
070700* THEIR VALUES THAT WERE EXTRACTED FROM THE DISPLAY COMMAND. 05920000
070800* THE FOLLOWING FIELDS AND THEIR ASSOCIATED VARIABLES CAN 05930000
070900* BE USED TO VALIDATE THE FIELDS:                    05940000
071000*                                                  05950000
071100*          FIELD                                VARIABLE 05960000
071200*          -----                                - 05970000
071300*          TSYSID                                WORK-TSYSID-SET 05980000
071400*                                                  WORK-TSYSID-VALUE 05990000
071500*          OSYSID                                *WORK-OSYSID-COUNT 06000000
071600*                                                  WORK-OSYSID-VALUE - AN ARRAY 06010000
071700*          OBJNAME                                *WORK-OBJNAME-COUNT 06020000
071800*                                                  WORK-OBJNAME-VALUE - AN ARRAY 06030000
071900*          APUSERID                              WORK-APUSERID-SET 06040000
072000*                                                  WORK-APUSERID-VALUE 06050000
072100*          UTERMID                              WORK-UTERMID-SET 06060000
072200*                                                  WORK-UTERMID-VALUE 06070000
072300*          OBJDEFIN                              WORK-OBJDEFIN-SET 06080000
072400*                                                  WORK-OBJDEFIN-VALUE 06090000
072500*          **COLLNAME                            *WORK-COLLNAME-COUNT 06100000
072600*                                                  WORK-COLLNAME-VALUE - AN ARRAY 06110000
072700*          **WKSTATID                            WORK-WKSTATID-SET 06120000
072800*                                                  WORK-WKSTATID-VALUE 06130000
072900*                                                  06140000
073000*          * THE WORK-OSYSID-COUNT, WORK-OBJNAME-COUNT, AND 06150000
073100*          WORK-COLLNAME-COUNT VARIABLES SHOULD ALL EQUAL 06160000
073200*          THE WORK-OBJDEFIN-VALUE (IF THE FRONT-END 06170000
073300*          APPLICATION IS NOT IPFAF 1.2). IF THE FRONT-END 06180000
073400*          APPLICATION IS IPFAF 1.2, ONLY THE WORK-OSYSID-COUNT 06190000
073500*          AND WORK-OBJNAME-COUNT VARIABLES SHOULD EQUAL 06200000
073600*          WORK-OBJDEFIN-VALUE. THIS IS BECAUSE IPFAF 1.2 06210000
073700*          DOES NOT SUPPLY THE COLLNAME FIELD ON THE DISPLAY 06220000
073800*          COMMANDS.                                  06230000
073900*                                                  06240000
074000*          ** IF YOUR FRONT-END APPLICATION IS IPFAF 1.2, THESE 06250000
074100*          FIELDS WILL NOT BE CONTAINED IN THE DISPLAY COMMAND. 06260000
074200*                                                  06270000
074300*          THE CODE IN THIS PARAGRAPH IS STRICTLY INSTALLATION 06280000
074400*          DEPENDENT. */                              06290000
074500                                                  06300000
074600*****                                              06310000
074700*** ADD ANY NECESSARY VALIDATION CHECKS HERE. *** 06320000
074800*****                                              06330000
074900                                                  06340000
075000*/* THIS TEST WILL SEE IF ANY REMOTE IODM PROCESSING IS INVOLVED 06350000
075100* IN THIS DISPLAY COMMAND. REMOTE IODM PROCESSING INVOLVES 06360000
075200* ANY OBJECTS THAT ARE STORED AT A REMOTE IODM OR THE OBJECTS 06370000
075300* INVOLVED IN THE DISPLAY COMMAND WILL BE DISPLAYED AT A 06380000
075400* WORKSTATION ON A REMOTE IODM. */                  06390000
075500          PERFORM                                    06400000
075600          VARYING WORK-INDEX                        06410000
075700          FROM 1 BY 1                               06420000
075800          UNTIL ((WORK-INDEX > WORK-OSYSID-COUNT) OR 06430000

```

075900	WORK-REMOTE-OBJECT)	06440000
076000		06450000
076100*/*	IF TSYSID DOESN'T EQUAL OSYSID, IODM REMOTE PROCESSING	06460000
076200*	IS INVOLVED. */	06470000
076300	IF (WORK-TSYSID-VALUE NOT = WORK-OSYSID-VALUE(WORK-INDEX))	06480000
076400	SET WORK-REMOTE-OBJECT TO TRUE	06490000
076500	END-IF	06500000
076600	END-PERFORM.	06510000
076700		06520000
076800	430-VALIDATE-FIELDS-EXIT.	06530000
076900	EXIT.	06540000
077000		06550000
077100	500-GENERATE-TRAN.	06560000
077200*/*	THIS PARAGRAPH WILL BE USED TO DETERMINE WHAT THE NEW	06570000
077300*	TRANSACTION ID VALUE WILL BE. THIS TRANSACTION ID CAN BE	06580000
077400*	DETERMINED BY USING THE PARAMETERS EXTRACTED FROM THE GDS	06590000
077500*	DATA FOR THE DISPLAY COMMAND. */	06600000
077600*		06610000
077700***	NOTE TO PROGRAMMER***	06620000
077800*	ALL TRANSACTION IDS GENERATED BY THIS PARAGRAPH MUST BE	06630000
077900*	DEFINED IN THE CICS PROGRAM CONTROL TABLE (PCT). IF YOU	06640000
078000*	DO NOT DEFINE THESE TRANSACTION IDS IN THE CICS PCT,	06650000
078100*	A TRANSIDERR WILL OCCUR ON THE CICS START COMMAND.	06660000
078200*****		06670000
078300		06680000
078400		06690000
078500*/*	SET THE NEW TRANSACTION ID TO THE DEFAULT TRANSACTION ID	06700000
078600*	IN CASE THE NEW TRANSACTION ID CAN NOT BE DETERMINED FROM	06710000
078700*	THE SELECTION LOGIC BELOW. */	06720000
078800	MOVE CNST-TRANSID-DEFAULT TO WORK-TRANSID.	06730000
078900		06740000
079000*****		06750000
079100***	ADD SELECTION LOGIC FOR THE NEW TRANSACTION ID HERE. ***	06760000
079200*****		06770000
079300		06780000
079400	500-GENERATE-TRAN-EXIT.	06790000
079500	EXIT.	06800000
079600		06810000
079700	600-START-DISPLAY.	06820000
079800*/*	THIS PARAGRAPH WILL REISSUE THE DISPLAY COMMAND TO IODM	06830000
079900*	USING THE NEW TRANSACTION ID. */	06840000
080000		06850000
080100***	NOTE TO PROGRAMMER***	06860000
080200*	REFER TO THE FOLLOWING COMMENTS ON WHICH CICS START	06870000
080300*	COMMAND YOU SHOULD USE FOR YOUR INSTALLATION. AFTER	06880000
080400*	DETERMINING THE CORRECT CICS START COMMAND, YOU MUST	06890000
080500*	EITHER COMMENT OUT OR DELETE THE OTHER TWO CICS START	06900000
080600*	COMMANDS.	06910000
080700*****		06920000
080800		06930000
080900		06940000
082000*/*	IF YOUR INSTALLATION IS USING IPFAF/API CICS, USE THE	06950000
082100*	FOLLOWING START COMMAND AND DELETE THE OTHER TWO START	06960000
082200*	COMMANDS IN THIS PARAGRAPH. */	06970000
082300	EXEC CICS START	06980000
082400	TRANSID (WORK-TRANSID)	06990000
082500	FROM (GDS-BUFFER)	07000000
082600	LENGTH (WORK-GDS-LENGTH)	07010000

082700	RTRANSID (WORK-RTRANSID)	07020000
082800	END-EXEC.	07030000
		07040000
083900	***NOTE TO PROGRAMMER***	07050000
084000*	IF YOUR INSTALLATION IS USING A CUSTOM FRONT-END	07060000
084100*	APPLICATION, YOU MAY WANT TO CONSIDER CHANGING THE	07070000
084200*	TRANSACTION ID FOR THE DISPLAY COMMAND FROM THE FRONT-END	07080000
084300*	APPLICATION. THAT WILL ELIMINATE THE NEED FOR THIS EXIT.	07090000
084400*	IF YOU CHOOSE NOT TO CHANGE THE TRANSACTION ID IN THE	07100000
084500*	FRONT-END APPLICATION, USE THE FOLLOWING GUIDELINES FOR	07110000
084600*	CHOOSING A CICS START COMMAND TO REISSUE THE DISPLAY	07120000
084700*	COMMAND:	07130000
084800*		07140000
084900*	A. IF YOUR FRONT-END APPLICATION RUNS IN AN IMS REGION	07150000
085000*	AND YOU USE A RETURN TRANSACTION ID, USE THE CICS	07160000
085100*	START COMMAND ASSOCIATED WITH IPFAF/API IMS.	07170000
085200*	B. IF YOUR FRONT-END APPLICATION RUNS IN A CICS REGION	07180000
085300*	AND YOU USE A RETURN TRANSACTION ID, USE THE CICS	07190000
085400*	START COMMAND ASSOCIATED WITH IPFAF/API CICS.	07200000
085500*	C. IF YOUR FRONT-END APPLICATION RUNS IN EITHER AN IMS OR	07210000
085600*	CICS REGION AND YOU DO NOT USE A RETURN TRANSACTION	07220000
085700*	ID, USE THE CICS START COMMAND ASSOCIATED WITH	07230000
085800*	IPFAF 1.2.	07240000
085900	*****	07250000
086000		07260000
086100	600-START-DISPLAY-EXIT.	07270000
086200	EXIT.	07280000
086300		07290000
086400	800-CICS-ERROR.	07300000
086500*/*	THIS PARAGRAPH WILL PROCESS ANY CICS ERRORS THAT MAY OCCUR	07310000
086600*	WHILE EITHER RECEIVING THE DISPLAY COMMAND FROM THE FRONT-END	07320000
086700*	APPLICATION OR REISSUING THE DISPLAY COMMAND WITH THE NEW	07330000
086800*	TRANSACTION ID. */	07340000
086900		07350000
087000*/*	SAVE CICS EIB FIELDS. */	07360000
087100	MOVE EIBFN TO WORK-EIBFN.	07370000
087200	MOVE EIBRCODE TO WORK-EIBRCODE.	07380000
087300		07390000
087400*/*	RESET HANDLE CONDITION TO PREVENT LOOPING. */	07400000
087500	EXEC CICS HANDLE CONDITION	07410000
087600	ERROR	07420000
087700	END-EXEC.	07430000
087800		07440000
087900*/*	CONVERT EIBFN TO DISPLAYABLE HEX. */	07450000
088000	MOVE WORK-EIBFN TO WORK-FIELD-TO-CONV.	07460000
088100	MOVE 1 TO WORK-NUM-HALFWORDS.	07470000
088200	PERFORM 810-HEX-CONVERSION THRU	07480000
088300	810-HEX-CONVERSION-EXIT.	07490000
088400	MOVE WORK-HEX-FIELD TO WORK-DISPLAY-EIBFN.	07500000
088500		07510000
088600*/*	CONVERT EIBRCODE TO DISPLAYABLE HEX. */	07520000
088700	MOVE WORK-EIBRCODE TO WORK-FIELD-TO-CONV.	07530000
088800	MOVE 3 TO WORK-NUM-HALFWORDS.	07540000
088900	PERFORM 810-HEX-CONVERSION THRU	07550000
089000	810-HEX-CONVERSION-EXIT.	07560000
089100	MOVE WORK-HEX-FIELD TO WORK-DISPLAY-EIBRCODE.	07570000
089200		07580000
089300*/*	TO ENSURE THAT ONLY ONE INVOCATION OF THIS EXIT IS LOGGING	07590000
089400*	ERROR MESSAGES TO THE TEMPORARY STORAGE QUEUE AT A TIME,	07600000

089500*	ENQUEUE ON THE QUEUE NAME WHILE YOU ARE WRITING TO THE QUEUE	07610000
089600*	AND ISSUE A DEQUEUE REQUEST WHEN YOU ARE FINISHED WRITING TO	07620000
089700*	THE QUEUE. */	07630000
089800	EXEC CICS ENQ	07640000
089900	RESOURCE (WORK-QUEUE)	07650000
090000	LENGTH (LENGTH OF WORK-QUEUE)	07660000
090100	END-EXEC.	07670000
090200		07680000
090300*/*	WRITE A CICS ERROR MESSAGE TO A TEMPORARY STORAGE QUEUE. */	07690000
090400	EXEC CICS WRITEQ TS	07700000
090500	QUEUE (WORK-QUEUE)	07710000
090600	FROM (WORK-CICS-ERR-MSG)	07720000
090700	LENGTH (LENGTH OF WORK-CICS-ERR-MSG)	07730000
090800	END-EXEC.	07740000
090900		07750000
091000*/*	WRITE OUT THE DISPLAY COMMAND IF IT WAS RECEIVED. */	07760000
091100	IF WORK-DISPLAY-RECEIVED THEN	07770000
091200	EXEC CICS WRITEQ TS	07780000
091300	QUEUE (WORK-QUEUE)	07790000
091400	FROM (GDS-BUFFER)	07800000
091500	LENGTH (WORK-GDS-LENGTH)	07810000
091600	END-EXEC	07820000
091700	END-IF.	07830000
091800		07840000
091900*/*	RELEASE ENQUEUE ON TEMPORARY STORAGE QUEUE. */	07850000
092000	EXEC CICS DEQ	07860000
092100	RESOURCE (WORK-QUEUE)	07870000
092200	LENGTH (LENGTH OF WORK-QUEUE)	07880000
092300	END-EXEC.	07890000
092400		07900000
092500*/*	TERMINATE PROCESSING. */	07910000
092600	PERFORM 900-RETURN THRU	07920000
092700	900-RETURN-EXIT.	07930000
092800		07940000
092900	800-CICS-ERROR-EXIT.	07950000
093000	EXIT.	07960000
093100		07970000
093200	810-HEX-CONVERSION.	07980000
093300*/*	THIS PARAGRAPH WILL TAKE AS INPUT A HEXADECIMAL NUMBER	07990000
093400*	AND CONVERT IT INTO DISPLAYABLE CHARACTERS. */	08000000
093500		08010000
093600*/*	INITIALIZE THE DISPLAYABLE CHARACTER ARRAY AND	08020000
093700*	ITS INDEX. */	08030000
093800	INITIALIZE WORK-HEX-FIELD.	08040000
093900	MOVE 1 TO WORK-HEX-FIELD-SUB.	08050000
094000		08060000
094100*/*	PERFORM THE FOLLOWING LOOP FOR EACH 2 BYTES OF HEXADECIMAL	08070000
094200*	DATA THAT NEEDS TO BE CONVERTED. */	08080000
094300	PERFORM	08090000
094400	VARYING WORK-HALFWORD-SUB	08100000
094500	FROM 1 BY 1	08110000
094600	UNTIL (WORK-HALFWORD-SUB > WORK-NUM-HALFWORDS)	08120000
094700		08130000
094800*/*	GET THE NEXT 2 BYTES OF HEXADECIMAL DATA. */	08140000
094900	MOVE WORK-HALFWORD-TO-CONV(WORK-HALFWORD-SUB) TO	08150000
095000	WORK-DIVIDEND-CHAR-2-BYTES	08160000
095100	MOVE 4096 TO WORK-DIVISOR	08170000
095200		08180000
095300*/*	THE FOLLOWING LOOP WILL CONVERT ONE HEXADECIMAL	08190000

095400*	CHARACTER AT A TIME. */	08200000
095500	PERFORM 4 TIMES	08210000
095600		08220000
095700*/*	DETERMINE THE VALUE OF THE HEXADECIMAL CHARACTER	08230000
095800*	AND MOVE THE VALUE TO THE DISPLAYABLE CHARACTER	08240000
095900*	ARRAY. */	08250000
096000	DIVIDE WORK-DIVIDEND BY WORK-DIVISOR	08260000
096100	GIVING WORK-QUOTIENT	08270000
096200	REMAINDER WORK-REMAINDER	08280000
096300	MOVE WORK-HEX-VALUE (WORK-QUOTIENT + 1) TO	08290000
096400	WORK-HEX-NUMBER (WORK-HEX-FIELD-SUB)	08300000
096500		08310000
096600*/*	PREPARE VARIABLES TO CONVERT NEXT HEXADECIMAL	08320000
096700*	CHARACTER. */	08330000
096800	ADD 1 TO WORK-HEX-FIELD-SUB	08340000
096900	DIVIDE 16 INTO WORK-DIVISOR	08350000
097000	MOVE WORK-REMAINDER TO WORK-DIVIDEND	08360000
097100	END-PERFORM	08370000
097200	END-PERFORM.	08380000
097300		08390000
097400	810-HEX-CONVERSION-EXIT.	08400000
097500	EXIT.	08410000
097600		08420000
097700	900-RETURN.	08430000
097800*/*	THIS PARAGRAPH WILL TERMINATE PROCESSING FOR THIS MODULE. */	08440000
097900		08450000
098000	EXEC CICS RETURN END-EXEC. GOBACK.	08460000
098100		08470000
098200	900-RETURN-EXIT.	08480000
098300	EXIT.	08490000

Appendix D. Software Products and Service Levels

In this section, we list the core software products we used for our benchmark at the sysplex. For additional information regarding the environment, please consult the *OS/390 Parallel Sysplex Test Report, GC28-1963*.

D.1 Core Sysplex Software Products Installed

Some of our core sysplex software products are elements or features of OS/390, and some are not.

D.1.1 Core OS/390 Elements and Features Installed

The following figure lists those elements and features of OS/390 that we consider to be part of our core.

Table 13. Core OS/390 Elements and Features Installed

FMID	Element or Feature	Release	Base Service Level
HDZ11C0 JDZ11CB	DFSMS Note: Included in these FMIDs are DFSMSdfp, DFSMSdss, DFSMSshsm, and DFSMSrmm.	OS/390 R3	GA level
HCS6031	HCD		
HIO1104	IOCP		
HJE6603	JES2		
HJS6603	JES3		
HBB6603	MVS Base Control Program (BCP)		
HRF2230	RACF (part of the OS/390 Security Server)		
HRM6603	RMF		
HQX1700	SDSF		
HVT4401	VTAM		

D.1.2 Other Core Sysplex Software Products Installed

The following figure lists core products that are not elements or features of OS/390.

Table 14 (Page 1 of 2). Other Core Software Products Installed

FMID	Product	Version/Release	Base Service Level
HCI4100 JCI4101 JCI4102 JCI4103 JCI4106 JCI4107	CICS/ESA	V4/R1	SMC9709

<i>Table 14 (Page 2 of 2). Other Core Software Products Installed</i>			
FMID	Product	Version/ Release	Base Service Level
HCI5100 JCI5101 JCI5102 JCI5103 JCI5104 JCI5106 JCI5107 JCI5109	CICS TS	Release 1	SMC9710
HLR2120 JLR2121 JLR2122	CICSplex SM	V1/R2	SMC9630 + selected APARs and PTFs (listed in Table 15 on page 86)
HCCV300 JCCV302	CICSVR	V2/R3	SMC9630 + selected PTFs (listed in Table 15 on page 86)
HMK5100 JMK5102 JMK5106 JMK5112 JMK5114 JMK5116 JMK5124 JMK5126 JMK5134 JMK5154 JMK5164	IMS/ESA	V5/R1	SMC9705
HDB4410 JDB4414 HIX4410 HIY4410 HIZ4410 HOG9100 JOG910E HDB441A JDB4415 JDB4416	DB2	V4/R1	SMC9708
HIR2101	IRLM	V2/R1	SMC9708

The following are additional APARs and PTFs we installed for CICSplex SM and CICSVR above the SMC9630 base service level:

<i>Table 15. APARs/PTFs for CICSplex SM and CICSVR</i>		
Product	APAR or PTF	Number
CICSplex SM	APARs	PN83198 PN89301 PN89809 PN89855 PN90219 PN90759 PN92552 PN92673 PN92690 PN92700 PN92784 PQ00374 PQ01072 PQ02522
	PTFs	UN94970 UN95800 UN96733 UN96735 UN97002 UN97048 UN97133 UN97138 UN97241 UN99633 UN99643 UQ00486 UQ00860 UQ01084 UQ01148 UQ01910
CICSVR	PTFs	UN93076 UN93690 UN93692 UN93827

D.2 Additional OS/390 Elements and Features Installed

The following additional elements and features are all part of OS/390 R3, and service is at GA level:

<i>Table 16 (Page 1 of 3). Additional OS/390 Elements and Features Installed</i>	
FMID	Element or Feature
HBD6602	BDT
JBD6201	BDT FTF
JBD6202	BDT SNA NJE
HBKP300 JBKP310	BOOKMANAGER BUILD
HBKM300 JBKM310	BOOKMANAGER READ
HLB4601 JLB4602 JLB4603 HLC4631 HTV4621 JTV4622 JTV4623 HXV4641 JXV4642 HCKVB00 JCKVB03	C/C++
HDZ11SM HDZ11SE	DFSMS/MVS NETWORK FILE SYSTEM--SERVER DFSMS/MVS NETWORK FILE SYSTEM--CLIENT
HSM1F00 JSM1FC0 JSM1FJ0 JSM1FK0 JSM1FN0 JSM1F01 JSM1F42	DFSORT
EDU1G01 FDU1G07 FDU1G08	DEVICE SUPPORT FACILITY-ICKDSF DSF ISMF MODS DSF ISMF PANELS
EER3500	EREP
HSWF100	ESCON DIR 1.1 J2
HEZ6603	EZ APPLICATION ENABLING
HFST101	FFST
HGD3200 JGD3219	GDDM/MVS
HGD3201	GDDM/MVS PGF
JCS6033	HCD NLS US ENGLISH
HMQ4120	HIGH LEV ASSMBLR
JMQ412A	HIGH LEV ASSMBLR TOOLKIT
HIMW200 JIMW201 JIMW210 JIMW220	INTERNET CONNECTION SECURE SERVER FOR OS/390
HMOS100 HPRF226	IP PRINTWAY NETSPOOL

<i>Table 16 (Page 2 of 3). Additional OS/390 Elements and Features Installed</i>	
FMID	Element or Feature
HIF4402 JIF4404 JIF4405 JIF4410 JIF4470	ISPF DM BASE ISPF CLIENT SERVER ISPF CLIENT SERVER RUNTIME ISPF DM ENGLISH ISPF SCLM
JJE66N3	JES2--ENU
HMR6160 JMR616N	LANRES FOR MVS LANRES FOR MVS - ENU
HQW7300 JQW7301	OS/390 LAN SERVER OS/390 LAN SERVER NLS US ENGLISH
HMWL710 JMWL71B JMWL75H JMWL75S	LANGUAGE ENVIRONMENT
EMI2220	MICR/OCR 2.2.0 J2
HTCP320 JTCP321 JTCP322 JTCP325 JTCP326 JTCP327	TCP/IP BASE TCP/IP CICS SOCKETS TCP/IP KERBEROS DES TCP/IP IMS SOCKETS TCP/IP NPF TCP/IP OE APPLICATIONS
HOT1150 JOT15N0	OPENEDITION APPLICATION SERVICES OPENEDITION APPLICATION SERVICES NLS US ENGLISH
HMB3122 JMB3125	OPENEDITION DCE BASE SERVICES OPENEDITION DCE USER DATA PRIVACY FEATURE (DES)
H0H2130	OPENEDITION DCE DISTRIBUTED FILE SERVICE
HRSS110	OPENEDITION DCE SECURITY SERVER
H0GI200	OPEN SYS ADAPTER/SUPP FACILITY
JRM62N3 HIPM110	RMF--ENU RMF: PERFORMANCE MONITORING COMMON FUNCTIONS
HQG1406 HSR1401 JSR1413 HPRF220 JPRF223 H0HL140 H0HL141 J0HL145 J0HL146 J0HL14F J0HL14T J0HL14U H0HL15D J0HL152 H0HL15E J0HL153 J0HL154 J0HL15B	SOFTCOPY PRINT SOLUTION AND FONTS: BOOKMASTER DOCUMENT COMPOSITION FACILITY PSF/MVS BASE WITH ALL PRINTER SUPPORT PSF/MVS RESOURCES AFP FONT COLLECTION - SELECTED FOR OS/390
HMP1A00 JMP1A01	SMP/E SMP/E - ENU
JQX1701	SDSF--ENU
HSLW110 JSLW1N1	SYSTEMVIEW FOR MVS BASE SYSTEMVIEW FOR MVS BASE - ENG
H08K310 J08K3N0	SOMOBJECTS FOR MVS RUNTIME LIBRARY SOMOBJECTS FOR MVS RUNTIME LIBRARY ENGLISH

<i>Table 16 (Page 3 of 3). Additional OS/390 Elements and Features Installed</i>	
FMID	Element or Feature
H08K311 J08K3N1	SOMOBJECTS APPLICATION DEVELOPMENT ENVIRONMENT SOMOBJECTS APPLICATION DEVELOPMENT ENVIRONMENT ENGLISH
H18K114	SOMOBJECTS SERVICE CLASSES
ETI1106	TIOC
HTE25D2 JTE25D0	TSO/E TSO/E - ENU
H0GG121	VISUALLIFT
HFX1112	3270 PC FILE TRANSFER PROGRAM

D.3 Other Sysplex Software Products Installed

These are installed products that are not elements or features of OS/390:

<i>Table 17. Other Sysplex Software Products Installed</i>			
FMID	Product	Version/ Release	Base Service Level
JCL1403 JCL1404 JCL1406 JCL1407	COBOL II	V1/R4	SMC9710
HCQ1400	COBOL II COMPILER AND LIBRARY	V1/R4	SMC9710
HCL1400	COBOL II LIBRARY	V1/R4	SMC9710
JCL1402	COBOL II LIBRARY CICS	V1/R4	SMC9710
HGA1311 HGA1310	GAM MVS/ESA OPT GAM/SP /ESA42	V1/R3.1	RSU9701 plus HIPER/PE fixes

Appendix E. Special Notices

This publication is intended to help people who support ImagePlus MVS/ESA to do capacity planning for their system. Some tuning tips are also included. The information in this publication is not intended as the specification of any programming interfaces that are provided by ImagePlus MVS/ESA. See the PUBLICATIONS section of the IBM Programming Announcement for ImagePlus MVS/ESA for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of

including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AFP	AT
CICS	CICS/ESA
DB2	DFSMS
DFSMS/MVS	DFSMSdfp
DFSMSdss	DFSMShsm
DFSMSrmm	DFSORT
ES/9000	ESCON
FFST	GDDM
IBM	ImagePlus
IMS/ESA	MVS (logo)
MVS/DFP	MVS/ESA
OS/2	OS/390
Parallel Sysplex	Presentation Manager
PSF	RACF
RMF	S/390
SAA	System/390
VTAM	400

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

Appendix F. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

F.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see "How to Get ITSO Redbooks" on page 95.

- *IBM SAA ImagePlus MVS/ESA Document Capture Implementation*, GG24-4080
- *International Standards -- A Document Imaging Review*, GG24-2544

F.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
Application Development Redbooks Collection	SBOF-7290	SK2T-8037
Personal Systems Redbooks Collection	SBOF-7250	SK2T-8042

F.3 Other Publications

These publications are also relevant as further information sources:

- *OS/390 Parallel Sysplex Test Report*, GC28-1963 (orderable only as part of SK2T-6700. This is a full specification of the environment used for this redbook.)
- *ImagePlus MVS/ESA V2R2 General Information Manual*, GC31-7537
- *ImagePlus MVS/ESA V2R2 Planning Guide*, GC31-7538
- *ImagePlus MVS/ESA V2R2 Library Guide*, GC31-7539
- *ImagePlus MVS/ESA V2R2 Master Glossary and Index*, GC31-7555
- *ImagePlus MVS/ESA V2R2 Installation Guide*, SC31-7567
- *ImagePlus MVS/ESA V2R2 STAF User's Guide*, SC31-7568
- *ImagePlus MVS/ESA V2R2 Migration Guide*, GG22-2005
- *FaxRouter/2 V2.0 ImagePlus MVS/ESA Integration Guide*, SC31-6742
- *Large Systems Performance Reference*, SC28-1187

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at <http://www.redbooks.ibm.com>.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** —to order hardcopies in United States
- **GOPHER link to the Internet** - type `GOPHER.WTSCPOK.ITSO.IBM.COM`
- **Tools disks**

To get LIST3820s of redbooks, type one of the following commands:

```
TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
```

To get BookManager BOOKs of redbooks, type the following command:

```
TOOLCAT REDBOOKS
```

To get lists of redbooks, type one of the following commands:

```
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
```

To register for information on workshops, residencies, and redbooks, type the following command:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1996
```

For a list of product area specialists in the ITSO: type the following command:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
```

- **Redbooks Web Site on the World Wide Web**

<http://w3.itso.ibm.com/redbooks>

- **IBM Direct Publications Catalog on the World Wide Web**

<http://www.elink.ibm.com/pbl/pbl>

IBM employees may obtain LIST3820s of redbooks from this page.

- **REDBOOKS category on INEWS**
- **Online** —send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibm.com with the keyword `subscribe` in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (<http://www.redbooks.ibm.com/redpieces.htm>). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** — send orders to:

In United States:	IBMMAIL usib6fpl at ibmmail	Internet usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications Publications Customer Support P.O. Box 29570 Raleigh, NC 27626-0570 USA	IBM Publications 144-4th Avenue, S.W. Calgary, Alberta T2P 3N5 Canada	IBM Direct Services Sortemosevej 21 DK-3450 Allerød Denmark
--	--	--

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States) or (+1)001-408-256-5422 (Outside USA)** — ask for:

Index # 4421 Abstracts of new redbooks
Index # 4422 IBM redbooks
Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com

- **On the World Wide Web**

Redbooks Web Site	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibm.com/pbl/pbl

- **Internet Listserver**

With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibm.com with the keyword `subscribe` in the body of the note (leave the subject line blank).

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (<http://www.redbooks.ibm.com/redpieces.htm>). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

First name _____ Last name _____

Company _____

Address _____

City _____ Postal code _____ Country _____

Telephone number _____ Telefax number _____ VAT number _____

Invoice to customer number _____

Credit card number _____

Credit card expiration date _____ Card issued to _____ Signature _____

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Index

Numerics

3995 3

A

ADOC 6
ADOR 6, 27
API 1, 2, 13
APPC 13
 SIT
application design 17, 30
application programming interface (see API) 1
application, integration 5
automation, office 1

B

background transactions 29
 single threading 29
benchmark 5, 17, 18, 19, 20, 21, 26
 constraint 19
 results 19
 run 1 17, 18
 run 2 17, 20
 run 3 17, 21
 scenarios 5
bibliography 93
bottlenecks 19, 20
buffer pool
 settings
buffer pool settings 12
buffer pools 22
business transaction 7

C

caching 11, 30
capacity planning 23, 25, 26
 applying 26
 number of users 23, 25
capture ratio 18, 19
capturing 2
 batch 2
 high-speed 2
 scanning 2
cautions 22, 28
caveats 22, 28
CICS 12, 13, 18, 30
 auto-install 13
 CICS-3270
 SIT 13
 transactions 18
CICS transactions 23

CICS-3270 3
color 1
components of ImagePlus MVS/ESA 2
 Folder Application Facility (IPFAF) 2
 image workstations 2
 Object Access Method (OAM) 2
 Object Distribution Manager (IODM) 2
 Optical Storage Subsystem Products (OSSP) 2
concurrent access 1
contention 22
CPU utilization 18, 20, 21, 23, 27, 30
customer 8
customer service 1
customers 5
 typical 5

D

DASD 11, 12
 I/O contention 12
data 3
 integrity 3
Data Facility Storage Management Subsystem
 (DFSMS) 3
data security 1
databases 11
DB2 3, 12, 17, 21, 29, 30, 31
 buffer pool settings 12
 configuring 12
 DBRM 30
 indexes 31
 preload program 31
 integrity 3
 logging 3
 not a limiting factor 17
 number of accesses 21
 plan 30
 recovery 3
 type 2 indexes 31
DBRM 30
definitions 9
deleting 20
design, application 30
DFSMS 3, 12
DFSMS/MVS 3
displaying 2, 16, 17
displaying queues 25
distributed service center 5

E

editing 2
efficiency 1
ES/9000 11, 18

ES9000 28
estimating 22
exporting 2

F

facsimile (see also fax) 3
fax 2
FaxRouter/2 3
financial 5
FLDR 7
FMIDs
 additional OS/390 elements and features
 installed 87
 core OS/390 elements and features installed 85
 other core software products installed 85
 other sysplex software products installed 89
folder 3, 6
Folder Application Facility (IPFAF) 2
 Folder Work Application feature
Folder Work Application feature 12
Folder Work Application Feature (FWA)
form number 8
FWA 13

G

getwork 7, 18, 20, 21, 24, 26, 28, 31
 indexing 18
getwork and windup loop 6, 7

H

hardware 11
high-speed capture 2
high-volume 1
history view 25
how to use this information 23

I

I/O contention 12
IBM 3995 Optical Library Dataserver Products 3
IBM ES/9000 11
image processing 1
 high-volume 1
image workstations 2, 3
imageplus 9
 terminology 9
ImagePlus Capture Facility (ICPF) 3
ImagePlus Intelligent Forms Assist/2 (IPFA) 3
ImagePlus Intelligent Forms Facility/2 (IPFO) 3
ImagePlus Workstation Program (IWMP) 3
index and scan 6
indexing 2, 3, 6, 17, 20, 23, 26
industrial strength 1
information, office 1
inserting 20

insurance 5
integrity 3
 data 3
introduction 1
IODM 12, 16
 multiple copies 16
IPFAF 12, 16
 Folder Work Application feature 12
IWPM/2 16, 17

L

LAN 1
Large System Performance Reference 18
LE/370 12
Local Area Network (see LAN) 1
lock contention 30
LU2 3, 13, 17
LU6.2 3, 6, 13, 17

M

management control 1
mounts per hour 22
MVS
 DFP
 overhead
MVS overhead 19
MVS/DFP 3

N

network delays 17
number of users 23, 25
 creating 26
 selecting

O

OAM 3, 16
object 2
 movement 2
Object Access Method (OAM) 2
Object Distribution Manager (&ODM.)
Object Distribution Manager (IODM) 2
office automation 1
office information 1
OINT 13
optical 22
optical storage 11
Optical Storage Subsystem Products (OSSP) 2, 3
OSSP 3
overhead 17, 18
 capture ratio 18
 MVS
overhead, MVS 19

P

- package 30
- paper 1
- parameters, TPNS 18
- performance 22
- performance reference, large system 18
- plan 30
- preload program 31
- printing 2, 16
- processing, image 1
- processor resource 28, 30
- processor utilization 17
- productivity 1

Q

- queue 5, 7
- queues 8, 17, 21, 26, 28, 31
- queues, displaying 25
- queues, work 3

R

- RCT 12, 30
- receiving 17
- redirection 3
- rerouting 7
- resource utilization 16, 22, 23, 26, 29, 30
 - DB2 29
 - threads 30
- response 2
 - fast 2
- response time 17, 19, 20, 22
 - average 19, 22
 - fast
 - maximum 19, 22
- results 19
- retrieving 2, 3
- RLOB/transaction type 8
- RMF 18, 19
- routing 2, 6, 7, 26

S

- S/390 1
- scanning 1, 2, 6, 17, 22, 23, 26
- scenarios 5
- scripts, TPNS 6
- security, data 1
- sending 17
- service level
 - additional OS/390 elements and features installed 87
 - core OS/390 elements and features installed 85
 - other core software products installed 85
 - other sysplex software products installed 89
- service level agreement 17

- single threading 29
- SNA 3
 - LU2 3
 - LU6.2 3
- SNA LU6.2 2
- software 2, 11, 85
 - sysplex products 85
- storing 2, 3, 6, 17, 26
- supervisor 7, 25
- sysplex 17
- system load 8
- System/390 1

T

- tables 5, 6, 7, 8, 12, 20, 22, 29
 - folder 7
 - IPFAF static configuration 7
 - object 7, 22
 - small 22
- tape 11, 22
- TEMPID 6
- terminology 9
- threads 30
- throughput 22
- timeouts 29
- tips, tuning 29
- TPNS 11, 12, 17, 18, 20
 - parameters 18
 - response time 20
 - response times 17
 - TPNS scripts
 - getwork and windup loop
 - index and scan
 - run 1
 - run 2
 - run 3
 - workflow functions
- TPNS scripts 6, 7, 9, 23, 24, 25, 27, 28
 - Getwork and windup loop 6, 7
 - getwork windup script 24
 - Index and scan 6, 9
 - indexing and scanning 23
 - run 1 27
 - run 2 27
 - run 3 23, 28
 - workflow function 7
 - Workflow functions 6
 - workflow functions script 25
- tracking 3
- transaction
 - flows
 - profile
- transaction flows 23
- transaction profile 20
- transaction, business 7
- transactions 22
 - background
 - single threading

- transactions (*continued*)
 - business
 - multiple 22
 - per second
 - transactions per second 17, 28, 30
 - transactions per section 21
 - transactions, background 29
 - single threading 29
 - tuning tips 29

U

- unit code 8
- updating 20
- user assignment 8
- user profile 8
- utilization 21

V

- valid unit/RCODE queue definition 8
- view, history 25

W

- windup 7, 24
- work delivery 25
- work in process 3
- work queues 3
- work queues (see also queues)
- work queues (see queues)
- workflow 7, 21, 25
- workflow function 7
- Workflow functions 6
- workload 6
- workstations, image 3

Z

- ZPARMS 12

ITSO Redbook Evaluation

Image and Workflow Library: ImagePlus MVS/ESA Performance and Capacity Planning Benchmark
SG24-4975-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at <http://www.redbooks.com>
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@vnet.ibm.com

**Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)**

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes___ No___

If no, please explain:

What other redbooks would you like to see published?

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)



Printed in U.S.A.

SG24-4975-00

