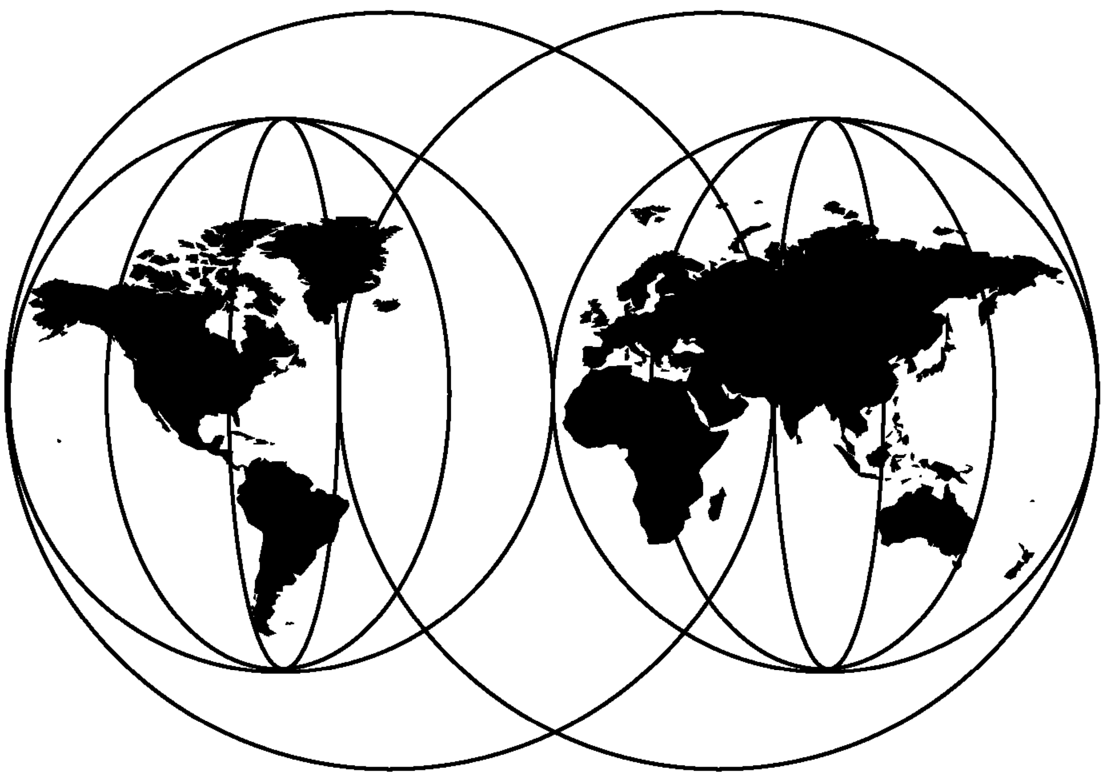
IMS/ESA Sysplex Data Sharing: An Implementation Case Study

Jim Boyle Alison Coughtrie Jean-Pierre de Villiers Steve Heisig
Gary Wicks Geoff Nicholls Attila Fogarasi



International Technical Support Organization

http://www.redbooks.ibm.com

This book was printed at 240 dpi (dots per inch). The final production redbook with the RED cover will be printed at 1200 dpi and will provide superior graphics resolution. Please see "How to Get ITSO Redbooks" at the back of this book for ordering instructions.

SG24-4831-00

International Technical Support Organization

SG24-4831-00

IMS/ESA Sysplex Data Sharing: An Implementation Case Study

August 1997

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix C, “Special Notices” on page 117.

First Edition (August 1997)

This edition applies to Version 5 Release 1 of IMS/ESA, Program Number 5695-176 and Version 2 Release 1 of IRLM Program Number 5695-164 for use with the MVS/ESA and OS/390 Operating Systems.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. QXXE Building 80-E2
650 Harry Road
San Jose, California 95120-6099

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

© Copyright IBM Corp. 1997

Figures

© Copyright IBM Corp. 1997

Tables

© Copyright IBM Corp. 1997

Preface

This redbook presents a case study of the planning and implementation of a Parallel Sysplex data sharing project using IMS/ESA Version 5, IMS Resource Lock Manager (IRLM) Version 2, and MVS/ESA Version 5.1 at Amalgamated Banks of South Africa (ABSA) located in Johannesburg, South Africa.

The steps that were taken to plan, prepare, and implement this project are outlined, with details on the challenges presented to ABSA during the migration and implementation of the solutions.

The book provides IMS/ESA system programmers, systems and application designers, database administrators, and application programmers who are responsible for the implementation of an IMS/ESA Parallel Sysplex data sharing environment with information regarding a case study of a migration to that environment.

Some knowledge of IMS block level data sharing, sysplex environments and the coupling facility, recovery procedures for failures in complex IMS/ESA environments, and the roles of IRLM and the coupling facility in support of IMS/ESA data sharing are assumed. This information is available in IBM Education and Training class U3767, “IMS/ESA Block Level Data Sharing.”

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world, working at ASBA in Johannesburg, South Africa; Melbourne, Australia; and the International Technical Support Organization San Jose Center, in San Jose, California.

Jim Boyle is an Advisory Systems Engineer in IBM Australia. He holds a degree in Engineering from the University of Melbourne and has 30 years′ of IBM experience, including 23 years working as an IMS specialist in Australia, New Zealand, and Asia. Jim installed the first IMS Fast Path accounts in Australasia and has assisted many customers in the design, installation, and performance enhancement of IMS systems. Jim has been a joint author of technical studies on IMS capacities and several IBM Redbooks on IMS and MQSeries. He is a joint inventor of a software technique to enhance the performance of input/output (I/O) subsystems in MVS/ESA.

Alison Coughtrie was a Senior Technical Specialist in IBM South Africa and now works in England with IBM at the International Support Center in Hursley, in the Data Management Center of Competency. She has 14 years of experience in the IMS field, 7 of those with IBM. Her areas of expertise include support of large IMS customer installations. She has written extensively on migration to IMS Version 5 sysplex data sharing.

Jean-Pierre de Villiers is an Advisory Technical Specialist in IBM South Africa. He has 8 years of experience in the IMS field. He holds a degree in Economics from Rand Afrikaans University. JP′s areas of expertise include IMS systems programming. He has written extensively on MSC in a Sysplex Environment.

Steve Heisig is a Software Developer on MVS Workload Management in IBM United States. He has 15 years of experience in the software development and

© Copyright IBM Corp. 1997

testing field. He holds a degree in Computer Science from the University of California at Berkeley. His areas of expertise include S/390 software development and testing.

Gary Wicks is an Advisory Systems Engineer in IBM Canada. He has 22 years of experience in the field of software support and technical support planning and management. Gary holds a degree in Math and Physics from the University of Toronto. His areas of expertise include diagnostic and implementation support for IMS/ESA customers. Gary has written extensively on IMS/ESA diagnostics, IMS MSC in Sysplex configurations, and application and database design in data sharing environments.

Geoff Nicholls is an Advisory Systems Engineer at the International Technical Support Organization - San Jose Center. Geoff has a Science degree from the University of Melbourne, Australia, where he majored in Computer Science. He worked as an application programmer and database administrator for several insurance companies before specializing in database and transaction management systems with Unisys and IBM. Since joining IBM in 1989, Geoff has worked extensively with IMS customers in Australia and throughout Asia.

Attila Fogarasi was an IMS specialist at the International Technical Support Organization- San Jose Center from 1993 to 1996. He wrote extensively and taught IBM classes worldwide on all areas of IMS/ESA. Before joining the ITSO, Mr. Fogarasi worked at the IBM Santa Teresa Labs as an IMS designer and development programmer.

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

IBM South Africa

Larry Lange
Harry Batten
Toitjie Cillié

ABSA Bank LTD.

Joris Lievens
Dave van der Merwe
Carl Moolman
André Schoeman
Neville Skead
Cassie Smith
Dolf Snyman

This book is dedicated to the memory of Stanley Michael Ghirelli.

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 131 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@us.ibm.com

Chapter 1. Introduction

The major objective of the project was to document the experiences of Amalgamated Banks of South Africa Limited (ABSA) during the implementation of an IMS/ESA sysplex data sharing environment. The structure we used parallels the activities of the project plan presented in Appendix A, "ABSA IMS Parallel Sysplex Project Plan" on page 81. In this document we follow the general project path, discussing the migration experiences that ABSA and IBM shared.

There are many excellent reference sources for background information about IMS/ESA Version 5, using IMS/ESA Version 5 in a sysplex data sharing environment, and general MVS Sysplex migration preparation (see Appendix D, "Related Publications" on page 119). This book does not attempt to repeat the information in those sources.

The major business rationale for migrating to an IMS/ESA parallel sysplex was to provide sufficient capacity for new application workloads. This rationale drove many of the decisions that IBM and ABSA made as the project evolved.

1.1 Assumptions

We have made the following assumptions in writing this book:

- IMS/ESA Version 5 runs in a production environment using the Program Isolation facility for lock management, rather than IMS Resource Lock Manager (IRLM) 1.5 or IRLM 2.1.
- A sysplex data sharing environment has been implemented before this project. The implementation includes all necessary hardware and the operating software (MVS/ESA or OS/390).

  The first move in getting to sysplex data sharing is to implement a base sysplex. In such an environment, all of the hardware is in place, including:

  – Sysplex timer
  – Channel-to-channel connections between the systems

  The base software environment is also in place, including:

  – Global resource serialization (GRS)

    You also must ensure that the Resource Name List in GRS is correct. If something starts to hang, you must be able to quickly identify that the problem stems from GRS locking and take the appropriate action.

  – Job Entry Subsystem 2 (JES2)
  – Resource Access Control Facility (RACF) properly set up for the sysplex environment.

    This is well documented in the RACF documentation.

  Having this work completed beforehand simplifies the migration to an MVS Parallel Sysplex and therefore reduces the number of problems that could be encountered during migration to an IMS sysplex data sharing. It will provide you with a lot of valuable experience with parallel sysplex operations.

© Copyright IBM Corp. 1997

- Data sharing has been exercised in development and test environments, but not yet in production.
- The current IMS production system and its applications will be cloned, using two way data sharing as much as possible. Please see 1.3, "Alternative Strategies for Workload Processing" on page 3.

1.2 History of ABSA and Current Business Status

ABSA is the controlling company of a group of South African banks, former building societies, and a number of diversified entities in the financial services field.

In 1991, ABSA was established as the holding company of the Allied, United, and Volkskas groups according to the terms of an agreement to merge the interests of the former Allied Group Limited, UBS Holdings limited, Volkskas Group Limited, and the acquisitions of certain interests from Sage Financial Services Limited. In 1992, ABSA merged with Bankorp Holding Limited and became the largest banking and financial services entity of its kind in Africa, with assets currently in excess of R115 billion (R: Rand). ABSA has branches, representative offices, or subsidiary companies in London, Frankfurt, Hamburg, Hong Kong, and Singapore.

Figure 1 presents a business profile of ABSA.

- April 1991, established as holding company of United Bank, Allied Bank, Volkskas Bank and Bankorp
- Largest banking institution in South Africa with assets of approximately US$26900 million
- 38,000 employees
- 1,900 branches and/or service points
- Products and services
  - Loans, Savings, Checks, Credit Cards, Deposits
  - Treasury, Foreign Exchange, Insurance, vehicle finance, short-term finance

Figure 1. Amalgamated Banks of South Africa Limited Business Profile

An important consideration in the conclusion of the merger between Allied, United, and Volkskas was the potential benefit to be derived from the rationalization of computer and management services. In 1991, ABSA owned six computer centers. Today it owns only two. The amalgamation of these sites required only 10 months to complete. Likewise, ABSA's common national backbone network was derived from three separate networks belonging to the individual member banking groups. In 1993, Allied accounts were merged into the United IMS system. This combined set of applications is referred to as ABSA1. Volkskas and Bankorp accounts are in the process of being moved onto the ASBA1 IMS system from CICS platforms. Because of the large amount of application rewrite involved, the new system using IMS is referred to as *ABSA2*.

This merging of banks did not occur without both business and technical challenges. There were different business cultures, standards, procedures, software, and branch environments. The rate of change was high and continues to be so, as would be expected in such a dynamic environment.

The business workload profile is shown in Figure 2. It is anticipated that the business workload processed by ABSA2 will require 600 IMS transactions per second. These transactions update the retail banking databases in real time, requiring significant processor resources. A SNAPSHOT benchmark was run in Raleigh, North Carolina, USA in 1995 to estimate the hardware capacity that would be required to run this workload. Various configurations were modeled, and it was decided that a data sharing Parallel Sysplex, with two IMS control regions, each on an ES/9000 982, would be required. ABSA′s experience with its business workload and corresponding IMS transaction workload has shown that this modeling was quite accurate. ABSA plans to add a third system to the Parallel Sysplex to handle the business workload.

ABSA decided to migrate to an IMS/ESA Parallel Sysplex because it was outgrowing the monolithic processor capacity available and foresaw that a Parallel Sysplex would provide the growth path.

- Used for all transactions for Allied and United
- All (1700) ATMs come through IMS
- Routes transactions to Volkskas and Bankorp
- Volkskas and Bankorp will use IMS by September 1997
  - Approx. ***600 transactions per second***

Figure 2. General Use of IMS at ABSA and Reasons for Sysplex Introduction

1.3 Alternative Strategies for Workload Processing

ABSA reviewed and considered alternative strategies to support the computing requirements of its business plans and chose to implement a sysplex data sharing strategy.

The main consideration that led ABSA toward a sysplex data sharing strategy was its business decision to use a single IMS image to process its retail banking workload. This decision led to a technical requirement to support substantially increased online and overnight BMP workloads as the business processing from participating banks shifted from other systems, as discussed in Chapter 1, “Introduction” on page 1. ABSA perceived that the simplest and most cost-effective mechanism to provide that processing capacity for the medium-term future was to embrace sysplex data sharing technology early and to develop good techniques to exploit the technology before it had an absolute, time-critical need for it.

Operators of other IMS installations will similarly have to consider possible alternatives, such as are discussed below. The various alternatives are by now traditional techniques for handling IMS workloads (and, sometimes, CICS workloads) that, for one reason or another, are split across multiple systems. Many of these techniques are used in IMS sites with varying degrees of success, so each alternative is worthy of consideration. Yet, it is fair to point out that the IBM system strategy is definitely toward sysplex data sharing because it overcomes many limitations of the alternative techniques and provides a sound base for prolonged system development in IMS sites.

1.3.1 Larger Single CPU

For at least some period into the future, ABSA could have supported its workloads by expanding to a larger central processor complex. However, ABSA knew that before the completion of the ABSA2 application rollout, its capacity requirements would exceed the maximum CPU capabilities available on a single processor complex. The expected changes in processing cost and capacities involved in CMOS processors are expected to make a transfer from the older hardware to CMOS-based hardware attractive in the near future. Such a transfer would require a sysplex data sharing environment.

Pursuing its expansion plans along a single-system path could suffice for only a limited period, and an alternative strategy was essential before completion of the ABSA2 project.

1.3.2 System Partitioning

Various possibilities exist for partitioning IMS workloads. Typical partitioning schemes include:

Geographic partitioning, where various parts of the bank network are supported from different IMS systems

Application partitioning, where separate applications (for example, retail banking, commercial banking, mortgage business) are supported on different systems

Organizational partitioning for workloads from different parts of the ABSA organization, the component organizations that merged to form ABSA, are run independently.

Front-end and back-end partitioning, where a front-end system handles online transactions from the terminal network, and a back-end system performs most, if not all, application functions.

Each partitioning scheme suggested has some degree of feasibility (after all similar schemes are used in other organizations), but they all present fundamental difficulties. The main difficulties are that partitioning a workload substantially increases the complexity of the applications and of system management within the installation. This increase in complexity involves such issues as additional application functions required to merge the results of all processing and to support application functions that cross whatever partition boundaries are used. Also, the processing of overnight batch jobs may still present issues where the transfer of data between system partitions may introduce dependencies that severely constrain overnight operations and can restrict the business capabilities of the system. For example, if a system were partitioned geographically, additional system constraints and/or additional system complexity would be required to support the application requirements of customers whose businesses cross partition boundaries. Consolidated processing for national clients would be more difficult in a partitioned system than in a consolidated system.

1.3.2.1 Additional Functions Required

Cross-partition applications, such as a transfer transaction affecting accounts within different partitions, would require additional application level functions that would not be required for similar transactions between accounts in the same partition. Definition, development, and maintenance of those additional functions would be an additional cost that would not be required for a sysplex data sharing approach where almost all functions would be operational on each of the participating IMS systems.

Additionally, each potential change in partitioning, such as a change from a two-way geographic split to a three-way split, would be quite difficult to coordinate and would require application definition and redevelopment to support the new application structure.

To support these cross-partition functions, it may be necessary to provide additional application functions such as MSC-directed routing of messages through MSC user exits. These exits require additional development effort and pose an extra workload of highly complex programs to be maintained.

1.3.2.2 Additional System Overhead

Online transactions that cross partition boundaries would incur additional overheads imposed by IMS MSC. Of these, the additional processing costs and the I/O workload placed on the IMS logging data sets may be the most important. In the case of partitioning into a front-end and back-end system, the overheads are likely to be most significant, and the workload of the back-end system would almost certainly be far less than that of the front-end system, thus limiting the potential benefits of the splitting of the system.

1.3.2.3 Multiple Systems Coupling at ABSA

MSC operates with the IMS transaction manager to provide transparent routing or balancing of a transaction workload among two or more IMS systems across a sysplex. ABSA decided not to use the IMS Workload Router, because that would have required the use of a single IMS as a front-end system to route all transactions across MSC links. ABSA believed the performance costs of this approach would have been prohibitive. For more information, see *IMS/ESA Multiple Systems Coupling in a Parallel Sysplex*, SG24-4750.

1.3.3 Benefits of Sysplex

ABSA chose to develop an IMS sysplex data sharing installation to support the ABSA2 system so that multiple IMS systems could be used to support the workload, with very few transactions restricted to a specific IMS system (that is, there would be few applications with specific system affinity). ABSA plans for an initial configuration of two IMS systems running on separate IBM ES/9000 982 systems. Part of the online network will be connected to each system, and MSC is to be used to transfer those few transactions with system affinity to the appropriate system as they arrive on the “other” IMS system.

In this system design, all data is accessible to each system, so all transactions and BMP programs can concurrently access any data required for the business functions they implement within a simple single system image. This system structure will facilitate the incremental addition of processing capacity by adding CMOS processors to the sysplex, with minimal effect on the overall configuration.

1.3.4 Future Expectations

IBM intends to extend the exploitation of IMS sysplex data sharing in future versions of IMS/ESA. Some of the extensions, introduced in IMS/ESA Version 6, are expected to ease the constraints of the Version 5.1 implementation. IMS/ESA Version 6 allows the sharing of DEDBs containing SDEPs and DEDBs using the virtual storage option (VSO), and provides the ability for the IMS Transaction Manager to share IMS message queues and fast path messages (through the shared Expedited Message Handler (EMH)) across the sysplex.

These developments are of particular interest to ABSA, which expect that its early implementation of IMS/ESA Version 5 sysplex data sharing will position it to exploit those new technologies early, providing more flexibility in the operation of its IMS systems.

1.3.5 Trade-offs

ABSA decided that the trade-offs involved in exploiting sysplex data sharing early in the life of that technology were manageable for several reasons:

- ABSA had participated in the Quality Partnership Program (QPP) for IMS Version 5 and had established confidence in the quality standards of new IMS software.
- ABSA had built some expertise in managing the introduction of new technologies and was quite confident in its skills.
- ABSA expected that a simple one-processor system could not support its projected workloads for more than a relatively short time. They expected the use of sysplex data sharing would actually prove to be simpler than any alternative scheme, such as system partitioning on geographic boundaries.
- ABSA believed that the effort to migrate to sysplex data sharing now would actually be less than the combined effort of moving to a partitioned system (and potentially removing that partitioning in the fairly short term) and then moving to sysplex data sharing, which they foresaw as inevitable within a few years.

1.4 General Information about the Sysplex Data Sharing Migration

Many groups of professionals were involved in this project. The major supporting teams came from the following organizations:

- ABSA
  - Facilities staff members, including networking, MVS systems programming, DATABASE2 (DB2), IMS, automation, operations, scheduling, capacity planning, operations analysis (training, operational and recovery procedures)
  - Application development programmers

    The applications staff were very heavily involved in this project because of the restrictions in IMS Version 5. This is discussed in 4.2, “Database and Application Considerations” on page 48.
- IBM South Africa
  - Account team
  - Customer engineering
  - Sysplex migration support group

- IBM USA
  - International Technical Support Organization, San Jose Center
  - IMS Development, Santa Teresa Laboratory
- IBM Europe, Middle East, Africa, Montpellier Center
  - Parallel Sysplex Competency Center

Fifteen staff members from the above local and international locations participated in this migration project but not all were active at any one time.

The first planning session was held in May 1995, with a target for the Parallel Sysplex production environment of October 1996. With the development of the ABSA2 application project, the target timescale was shortened dramatically with an August 1996 production target in selected branch locations.

The full Parallel Sysplex environment, which includes both IMS and DB2 two-way data sharing, was implemented in time for the month-end processing for October 1996 and has been in production successfully since then. ABSA is planning to expand to three-way data sharing by April 1997. This third system will use 9672 (CMOS-based) processors. The primary objectives of the project are shown in Figure 3.

- Provide sufficient capacity for ABSA2 workload
- Enable full data sharing for IMS and DB2 data
- Use "cloned" IMS systems - transactions can run on either IMS
- Make transparent to applications wherever possible

Figure 3. Primary Objectives of the Sysplex Data Sharing Project

The ABSA systems are tightly integrated. It was therefore very difficult to isolate any single application to a single IMS. For this reason, ABSA decided that all data (both IMS and DB2) be fully shareable between the two systems. Many Data entry databases (DEDBs) with sequential dependents (SDEPs) as well as main storage databases (MSDBs) were used with the ABSA application systems. Therefore, sharing solutions outside those supported with IMS/ESA Version 5.1 had to be developed. We discuss those solutions in 4.2, “Database and Application Considerations” on page 48.

ABSA decided the IMS systems would be “cloned,” with each of the IMS systems sharing data in the sysplex being nearly identical copies of each other. The factors influencing the cloning included:

- Application integration
- System manageability
- Transparency of sysplex for clients

  ABSA did not want users to have to logon to one system for certain transactions, and to another for other applications.

The following activity list provides more information about the gradual implementation approach associated with the IMS/ESA data sharing Parallel Sysplex migration:

January 1996	MVS - base sysplex
March 1996	MVS - Parallel Sysplex
May 1996	IMS - One-way data sharing (two production systems for IMS with IRLM SCOPE=LOCAL)
May 1996	DB2 - One-way data sharing
August 1996	IMS - Two-way data sharing (IRLM SCOPE=GLOBAL) with one pilot branch
August 1996	DB2 - Two-way data sharing with one pilot branch
August 1996	Stress test
August 1996	Begin roll out to branch locations countrywide
October 1996	Rollout to branch locations complete

Chapter 2. ABSA IMS Sysplex Project: The Planning Segment

Planning for an IMS sysplex data sharing project is an activity whose scope should not be underestimated. The ABSA Parallel Sysplex project was initially defined as creating an MVS sysplex platform, and the importance of project management was underestimated. As a result, the scope of the project changed quickly. Additional requirements were added to the project, and there was a high level of change, driven by those additional requirements and an increased understanding of the complexities of implementing a sysplex. Figure 4 on page 10 shows the project planning highlights.

ABSA management′s realization that, without a highly structured project management and commitment of resources the project could not succeed, led to a higher focus on project management. This focus in turn led to a major improvement in communications between the subproject groups, with regular formalized feedback sessions, and agreements on the scope of the project. Strict change control was implemented, and both ABSA and IBM created contingency plans for identified risks.

ABSA and IBM assigned full-time project managers. Personnel from each of the following areas were identified (along with backups) and were assigned responsibility for implementing their portion of the sysplex implementation:

- IMS systems support
- IMS and DB2 database administration
- MVS
- Applications
- Operations
- Automation
- Storage management
- Capacity planning and monitoring
- Performance
- Production scheduling

With the volume of changes required for implementing both the hardware and software, careful planning is a key success factor (see Figure 4 on page 10). Avoid including items on the critical path that increase the risk unnecessarily. Move tasks from later stages of the project to earlier stages if their presence on the critical path increases risk. Also ensure that only subprojects relevant to a Parallel Sysplex implementation are included in the project plan. Otherwise, the scope of the project increases, as do the potential risks.

Another key success factor is the establishment of an effective forum for communicating change. Hold weekly meetings with all support areas. Note key accomplishments and communicate planned activities for the forthcoming week. Discuss any "issues" that have affected the project thus far. ABSA held weekly status meetings for each subproject, where subproject-specific details were discussed.

Although the primary objective was to make the implementation as transparent as possible, ABSA found that it was particularly important to keep the

© Copyright IBM Corp. 1997

Applications and Operations departments informed throughout the project, even during periods when they did not directly participate.

- Project planning takes longer than you think!
- Set up a dedicated team - with two project managers and backups for all positions
- Document the project plan in a format that can be distributed to all members and modified if necessary
- Assign responsibilities in the scope of work
- Communication is key
  - "Transparency" does not mean you do not have to tell anyone what you are doing!

Figure 4. Project Planning Highlights

Throughout this chapter we refer to **STEP** numbers from the project plan in Appendix A, "ABSA IMS Parallel Sysplex Project Plan" on page 81. The project plan contains a Reference column with cross-reference entries to major sections in this book that discuss a given step. All steps in the project plan are listed in Appendix A, "ABSA IMS Parallel Sysplex Project Plan" on page 81 to assist in the creation of your own project plan. Most of the steps in this plan are discussed in this document. There are several steps which are not documented in the text, as they should be self-explanatory in the Appendix.

2.1 Understand the Current Environment

STEP 1 (see page 82)

The first step in developing any large migration project proceeds from an understanding of the core business and technical reasons for undergoing the migration. These reasons must be communicated and documented. Below we review the major tasks of Step 1 and the choices ABSA made for its migration.

- **Document the core business functions performed by the computing services in your organization**

  Ddetermine the core business functions that must continue to be available in the event of degraded systems availability. Because the ABSA systems are tightly integrated, it was difficult to isolate a specific set of applications, databases, or network resources as being critical. For ABSA, all functions are critical.

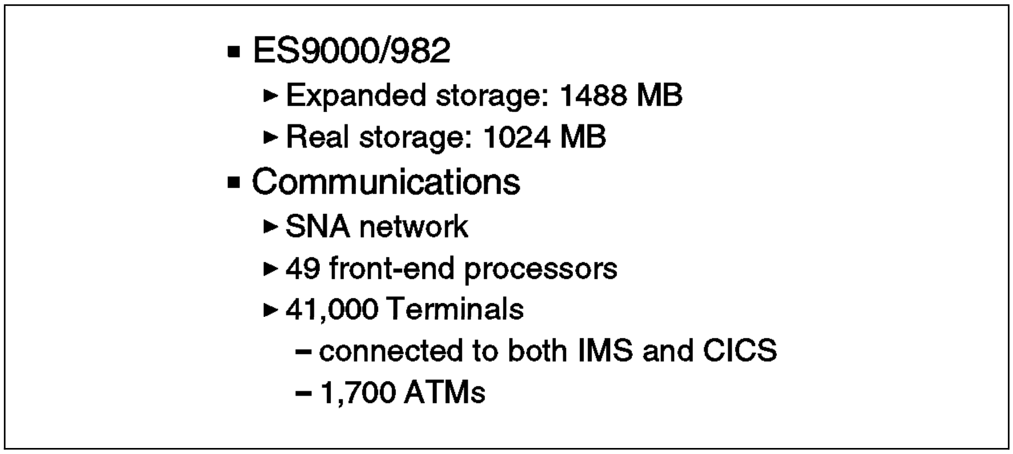


Figure 5. ABSA Production Hardware Environment (April 1997)

- **Map out the current hardware configuration and system software levels in use (see Figure 5 and Figure 6 on page 12)**

  ABSA's philosophy is to be up-to-date with hardware and software, and it has always been at the leading edge in both. It has an almost symmetric sysplex for its production site, backed up with equipment installed in a backup site 20 km away. ABSA also purchased CMOS-based processing equipment early, so it was familiar with this new technology in its specific environment. In the future, CMOS will play an increasingly important role at ABSA. Currently CMOS facilities are used for one of the coupling facilities.

  ABSA1 runs on an ES/9000-982 processor with:

  – Expanded storage of 1488 MB

  – Real storage of 1024 MB

  Since the late 1960s, ABSA has been a world leader in online banking. It has an extensive countrywide online network covering not only the major centers but all branches and agencies, even in remote locations. Traditionally ABSA has run a host-centric Systems Network Architecture (SNA) network, but it is in the process of rearchitecting the network to integrate with and handle multiple protocols.

  The communications environment consists of:

  – An SNA network

  – 49 front-end processors, which are a mixture of 3745s with 900 frames, and 3720s

  – 41,000 terminals defined to IMS, of which approximately 20,000 are active system logical unit program (SLUP) or logical unit type 0 (LU0) terminals

  – 1,700 Automated Teller Machines (ATMs)

  ABSA views itself as a world leader in transaction processing and has been part of the IMS Quality Partnership Program for several years. ABSA installed IMS/ESA Version 5 as the first step in its sysplex implementation and plans to move to IMS/ESA Version 6 as soon as it becomes available for the shared queue support. DB2 is also a critical part of the ABSA production environment, and ABSA keeps up-to-date with DB2 releases. (This project could not have been successful without DB2 data sharing.)

Current Production Software Environment

- MVS SP 5.1
- IMS V5
- DB2 V4
- VTAM 4.3
- RACF 1.9.2
- DFSMS 1.2
- NetView 3.1

Figure 6. Production Software Environment at ABSA (April 1997)

ABSA upgraded to VTAM/ESA Version 4.3 in July 1996. It will implement OS/390 Release 2 in March 1997.

- **List the external links and types of links that connect IMS to external subsystems**

  Advanced Program-to-Program Communication (APPC) or Logical Unit 6.2 (LU6.2) links connect ABSA to partner insurance companies. AS/400 and CICS applications are connected through LU6.1 protocols to IMS. Applications using CICS and MQSeries are under development, and use of IMS Open Transaction Manager Access (OTMA) is planned. It is important to understand how IMS currently connects to external subsystems because these connections will be required in the distributed IMS sysplex environment.

- **List the type of workload that is currently present in major applications**

  ABSA uses IMS full function and fast path applications and DB2 applications for ATM, branch, headquarters, and international operations support. The transaction rate peaks at 325 transactions per second during normal office hours, with an additional 40 transactions per second for DB2.

  ABSA places a lot of importance on the performance of its online workload. Service Level Agreements are in place for all workloads, and the actual performance is checked daily against these agreements. See Figure 7 on page 13 and Figure 8 on page 13 for a description of the ABSA production IMS environment.

Current Production IMS Environment

- Databases
  - 270 full function VSAM databases (82,000 cyl)
  - 65 DEDBs (104,000 cyl)
  - 9 MSDBs
- Regions
  - 77 MPPs
  - 10 IFPs
  - 42 concurrent BMPs
- Real-time update of all financial transactions
  - Peak: 325 tran/sec Average: 186 tran/sec
  - January 1997 - 86 million FF transactions
  - January 1997 - 46 million fast path transactions

Figure 7. ABSA Production IMS Environment (April 1997)

During January 1997, 95.2% of the 86 million full function IMS transactions were processed with a response time of less than 1 second. Similarly, 99.5% of the 46 million fast path expedited message handling (EMH) transactions were processed with a response time of less than 1 second.

All batch processing is run as batch message programs (BMPs).

Current Production IMS Environment

- Transactions per day
  - FF: 3.5 million EMH: 1.6 million
- Logging rate
  - Average at peak time: 703,951 bytes/sec
  - Maximum at peak time: 4.19 MB/sec
- DBRC share control
- No ETO yet
- ISC links to CICS (4.1, 3.3 and 2.1.2)
- APPC links to insurance companies
- Availability
  - 24x7, with 2-4 hour weekly change window

Figure 8. ABSA Production IMS Environment (April 1997), Continued

Dual logging is used for the write ahead data sets (WADS) and online logging data sets (OLDS). These data sets are on IBM 3390 direct access storage devices (DASDs) connected through 3990-6 storage controllers. Each

WADS is on its own pack. DASD fast write (DFW) is used for the OLDS and WADS.

The Extended Terminal Option (ETO) is used in the development environment and is planned for production.

- **Identify the types of users which require the most resources**

  Currently at ABSA, applications accessing full function databases require the majority of system resources. Such applications include online and BMP-driven applications.

- **Map out the service level commitments for systems in your organization that will be migrated**

  ABSA has a Service Level Contract (SLC) for 98% host services availability of the online production system. Achievement of this SLC is checked daily.

- **Prepare a current view of the enterprise computing environment with a view to processors, major application systems, and databases**

- **Document the on-line and batch schedules in your organization that will be migrated**

  ABSA has committed to providing online availability 24 hours per day 7 days per week, with a change window of 2-4 hours available each week on Sunday morning if required. Currently, the ATM systems can only function when connected to IMS. There are no off-host capabilities for the ATM systems.

  IMS/ESA system generations are run twice per month, in the early hours of Sunday mornings. Image copies and database reorganizations are also scheduled for Sundays. Concurrent image copies are taken weekly, and batch image copies are scheduled once per month.

  Implementation of new application function or system software can be done only during the 2-4 hour "change window" on Sunday mornings. Careful planning is required to ensure that testing and fallback can be achieved during this timeframe.

- **Identify any application backouts because of database lockouts**

  It is vital to examine the applications and databases that will be part of the data sharing environment, to determine whether contention resulting in long access delays or lockouts could occur. These delays can be caused by application design or database "hot spots."

  Although ABSA was not sure of the extent of locking contention in the current stand-alone environment, the project team identified a potential problem: converting from an MSDB to a hierarchic direct access method (HDAM) database, to allow sharing of the data stored in this database across the sysplex, caused contention. Both application and program specification block generation (PSBGEN) changes were required to relieve the contention. In Chapter 4, "Preparation for IMS Sysplex" on page 37 we discuss this activity in more detail, along with some of the tools to detect contention in database access.

- **Identify the performance and tuning tools and processes available presently**

  ABSA has a performance and monitoring team in place. Daily systems trends are monitored through IBM Service Level Reporter (SLR). More complete monthly systemwide reports showing performance trends are also created. Automated queries are used to check the status of the IMS system at any time. The performance monitoring tools in use at ABSA include:

  – IBM SLR

  – IMS Monitor

  – The Fast Path Log Analysis utility (DBFULTA0)

  – Resource Measurement Facility (RMF) Report III for control interval (CI) status

  – IMS File Select and Formatting Print utility (DFSERA10) with exit DFSERA30 to report deadlocks

  – IRLM V2.1 CTRACE along with the DL/I and lock traces

  – Omegamon from Candle Corp.

  – IMS Fastpath DC Monitor extensions from Innovative Designs.

- **List the databases and applications that currently present affinity status**

  Many applications have been designed to access a single resource that cannot be shared across the sysplex. These resources could include:

  – IMS databases such as MSDBs or DEDBs with SDEP segments at the IMS/ESA Version 5 and earlier levels.

  – Applications that serialize their function

    The resource may be a central interest rate table, for example. To ensure the integrity of the information, IMS locks the data for exclusive access. Any transactions accessing this resource are processed serially, as only one transaction can lock the data at any time. Eventually, the rate of access to this central resource will become a limiting factor in the performance of the application. At ABSA, the high transaction rate in a single system highlighted this problem and action was taken to eliminate it. Other organizations may encounter this problem.

  – Applications that access memory, files, or networks that are associated with a single system

  – Applications using hardware devices that are available on only one system. ABSA has an optical disk device, which cannot be shared across a sysplex

  – User exits or user modifications that have coded affinities to specific systems

  – IMS log analysis programs that review only one system′s logs per run.

- **Determine if you have the necessary skillset in house to perform the migration of the IMS, IRLM, and MVS components into a sysplex data sharing environment**

  It is necessary to have a core group with technical and managerial skills to support current computing business applications.

  The migration to IMS block level data sharing in a Sysplex environment involves people with skills in MVS, IMS, VTAM systems programming, project planning, automated operations, operations, change control, performance monitoring, database administration, Security, Application design and coding, and disaster recovery. ABSA and IBM identified people with skills in these areas for potential future involvement with the migration.

- **Review your change and problem control processes and document them**

  As with any major migration project, the rate of change may introduce problems which must be documented in a problem control process. From account executives to operations staff, there is a need to be aware of scheduled changes and problems, albeit at different levels of detail. Also, a process to communicate the status of the project must be available to all groups and levels in the organization for a project of this size.

  One element of professional change management gained prominence during this project: the need for design and implementation reviews by all affected support groups in the project before major changes were introduced. For example, if the automation support group were to introduce a new level of systems automation into the data sharing sysplex environment, the changes should be reviewed by representatives from the automation, MVS, IMS, DB2, and transaction processing system support groups. Because of the enhanced interdependencies among the elements in a sysplex environment, it is vital that all groups take responsibility for the process of change control.

2.2 Architect an Overall Component Image of the Sysplex Environment

STEP 2 (see page 82)

Now that you have a good understanding of the current environment, you can create the structure of the sysplex configuration. Below we review the major tasks of Step 2.

- **List the business reasons associated with migrating to a Sysplex environment.**

  There are several business reasons for choosing a Parallel Sysplex solution:

  - Return on investment (ROI) with the use of CMOS-based processors.
  - Positioning and flexibility associated with sysplex ease of growth and workload management.
  - Obtaining a competitive advantage through the use of leading edge technology.
  - Effective sharing of database resources among subsystems.
  - Capacity for expected growth of computing requirements.

  With the introduction into the corporate computing structure of the new ABSA2 application systems to support two new financial institutions, more processing capacity was required. This is the major business reason why ABSA chose the sysplex solution.

  Although the decision to migrate to this technology may not be made at the level of the systems support group, it is mandatory to understand why this large investment in technology and effort is being made.

- **List the technical reasons associated with migrating to a Sysplex environment.**

  The migration was driven by business expansion that would require a processing capacity estimated at around 600 IMS transactions per second for the ABSA1 and ABSA2 workloads. This capacity is beyond that of a single system.

- **What will the production and development configurations look like?**

  Here decisions are made on the manner in which the available hardware will be used for the IMS data sharing project as well as how the IMS subsystem would interface with the external environment—through the traditional network, intersystem communications (ISC), multiple systems coupling (MSC), or APPC.

  Figure 9 shows the production configuration ABSA wants to have operational. All external links to outside institutions as well as ATMs will be on one IMS. All BMPs will process on one IMS.

  The use of the MSC links to control affinity routing are discussed in Chapter 5, "Implementation of IMS Sysplex Data Sharing" on page 59.

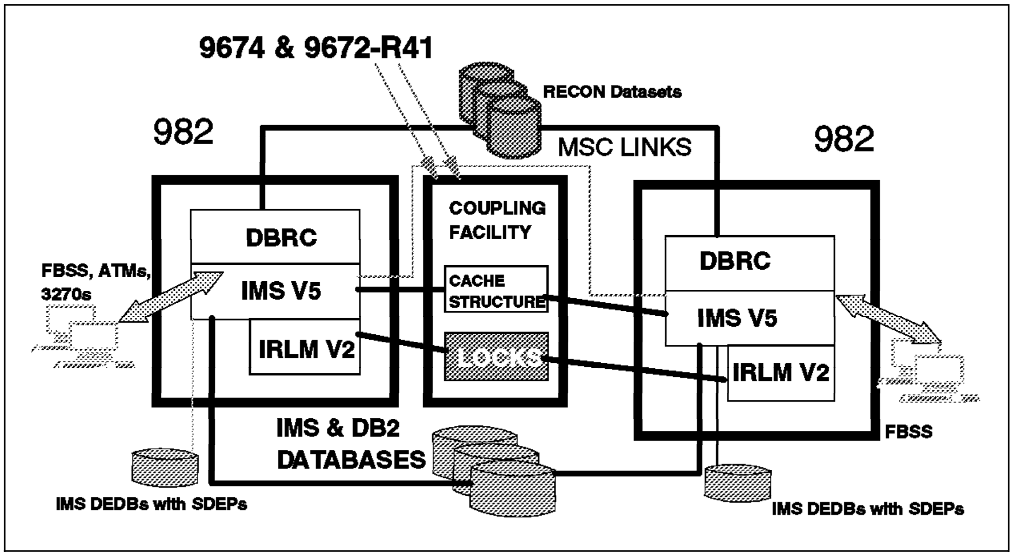


Figure 9. The Planned Production Configuration

Figure 10 on page 18 shows the user acceptance testing systems configuration in development. Additional IMS systems were used in the development cycle, but it was not practical to incorporate them into the sysplex. There is one coupling facility for this environment. For testing of recovery and failure scenarios, an additional coupling facility is planned.

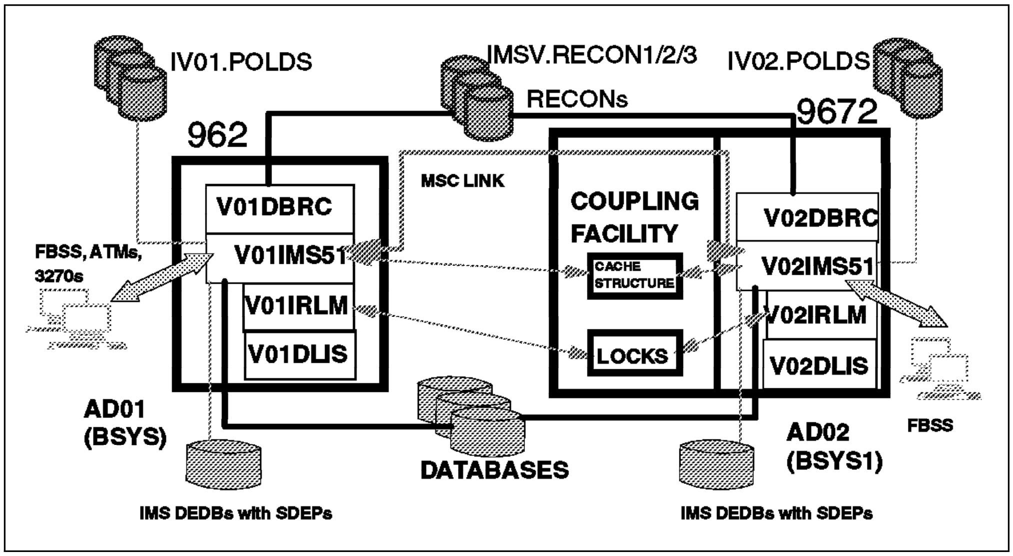


Figure 10. User Acceptance Test Systems Configuration

- **Map how the network will be connected to the Sysplex for use with the IMS environment**

  Each terminal will connect directly to one of two IMS control regions, and MSC will route transactions between systems on the basis of affinity status. Workload balancing was achieved by splitting the network rather than using MSC routing.

- **What IMS resources will be cloned**

  To manage system resources for highly integrated application environments the two IMS images were cloned as much as possible. DEDBs with sequential dependents and an MSDB with some specific applications had to undergo conversion to allow a data sharing environment.

- **What resources have affinities to one specific system**

  If a particular resource is identified as having an affinity to a particular MVS image or hardware resource, it must be noted at this point for study later during the data sharing compliance review.

  – ABSA′s optical disk applications (this is specific to ABSA)

  – MSDBs and DEDBs with SDEPs that are not converted in a data-sharing-compatible mode

  – BMPs that will be run on only one system

  – Hardware cryptographic facilities (this is specific to ABSA)

  – External links associated with only one system

  – Software packages that are not licensed to all processors in the sysplex configuration

- **Map the remote system connections to IMS**

  External connections at ABSA were associated mainly with ISC (LU6.1) and APPC (LU6.2) interfaces

2.2.1 Plan for Running the Sysplex in Degraded Mode

STEP 3 (see page 83)

The business and technical reasons for moving to a sysplex environment include the ability to operate production environments in a degraded mode when parts of the sysplex are unavailable. However, core business processing must be able to continue, even if in a constrained manner.

Below we review the major tasks of Step 3.

- **Define what *degraded mode* means based on the above sysplex architectural design**

  Each sysplex design will offer opportunities to exploit that design to enhance availability. With the integration of the ABSA systems, the entire environment would suffer if major components were unavailable (such as some production databases). By cloning as much as possible, using two processors does supply computing support to some end users if one processor or MVS image becomes unavailable.

- **Identify what core business functions must continue to operate when the sysplex is in degraded mode**

  To increase the availability of the identified core business functions, their logical position within the sysplex design must be mapped out. A system to prioritize the importance of business computing functions may have to be designed. Affinities that currently exist within core business applications may have to be resolved so that cloning of such programs can occur.

- **Map out how the network would interface with a sysplex running in degraded mode**

  Because transaction routing will probably be a large part of a degraded operational mode, the external network will have to be designed to provide input to the Sysplex in a nonrestrictive mode if necessary. This could include identifying how network resources would be reconnected to other systems if one system did fail.

  At this point, ABSA had not developed a network environment layout to operate in degraded modes of operation.

2.2.2 Develop the Sysplex Project Plan

STEP 4 (see page 84)

Now that you understand the current and target environment and have an idea of degraded mode operations within a sysplex, it is time to develop a detailed project plan. Figure 11 on page 21 shows the issues and contraints associated with the migration project.

Below we describe a few of the tasks associated with Step 4.

- **Obtain necessary executive and financial approvals**

  A project of this scope will receive attention from many groups within the organization. With the demands that will be made on corporate resources, it is imperative that the necessary management levels are aware and commit to support the project. An overview of the project timeline, major milestones, costs, and expected benefits must be communicated to the decision makers at this time.

- **Ensure that there is an effective process to communicate project status to executive management**

  The migration schedule with start dates, responsibility owners, and planned and actual completion dates will assist in the dialog. To obtain management understanding and effective support of the ongoing migration efforts, timely updates focusing on targets to completion of major steps along with early warning of risks and difficulties are vital. Reviews at milestone points will assist in management′s understanding of the progress and risk status along with new resourcing requests that might be tabled.

- **Assign a project coordinator**

  ABSA and IBM each appointed a full-time professional project manager from their senior ranks. These individuals drove most of the communication on the project through updates to the project plan, chairing meetings, and communicating with management in each organization.

- **Select the project management tools to use**

  The majority of project management tools were based on personal computer (PC) decision support products such as Microsoft Project, Lotus Ami Pro and Lotus Word Pro to map out the project schedule with start and end dates and time allocated to project members to complete each task. The use of problem control and tracking systems such as Servicelink were suggested but not used to any extent.

- **Determine if you have the necessary skillset in house to perform the migration of the IMS, IRLM, and MVS components**

  In 1995 personnel from ABSA participated in an IBM International Technical Support Organization project to write a redbook entitled *IMS/ESA Sysplex Data Sharing*, SG24-4303. Representatives from ABSA presented material on IMS Parallel Sysplex implementation in Europe during the spring of 1996. So the account was well prepared to embark on this migration. The primary IBM support staff on site had received the necessary training. The IBM Education and Training class, “IMS/ESA Block Level Data Sharing,” provides the necessary knowledge about IMS block level data sharing definition requirements for IMS, IRLM, and the CF and recovery procedures in this environment.

- **Ensure that the necessary resources (weekend migration and testing/time and systems access) are available and booked in advance**

  During a project covering at least several months, pressure may be applied to the process in the form of staff changes, other projects competing for corporate resources, or delays resulting from application or systems software problems. Contingency plans should be developed now to ensure that alternative plans can be activated if staff or systems access resources are limited.

- **Develop a risk assessment plan and review with all involved functions and management levels as required**

  The risk assessment plan could cover the following topics:

  - Disruption to “the business as usual” functions because of:

    - Requirements to redesign contention-prone applications and “hot spot” databases.

    - Conflicting schedules with users of traditional test facilities and time periods.

- Unforeseen outages during the initial implementation stages that could affect reaching the Service Level Agreement targets.

– Lack of stress testing before production cutover because the required production test environment with the necessary processors, coupling facilities, disk and tape packs are not available.

– Loss of vital staff during the long migration period with the reduction in group skillset.

– Implementation of a new environment that operates in parallel with existing systems. Such an environment presents opportunities to share resources that should not be shared and excludes system elements that should be part of the shared sysplex complex.

– The amount of change that the sysplex will bring. These changes are in the areas of operations, systems maintenance, automation, scheduling, applications, software levels, and procedures.

– Difficulty in maintaining healthy cross-functional relationships between the many groups involved with a long project as time goes on.

- Extremely tight implementation schedule
- Constrained resources
- Operations readiness
- Availability of facilities for testing
- Hardware availability
- IBM rescources for specific product support
- Leading edge environment
- Software release levels
- Workload movement

Figure 11. Issues and Constraints Associated with the Migration Project

Figure 12 on page 22 summarizes the positive elements associated with the migration effort.

Strengths

- Dedicated staff
- Good cooperation
- Common vision
- Good project control
- Focused IBM support
  - Local
  - Overseas
- Good fallback procedures
- Gradual Implementation

Figure 12. Strengths Associated with the Migration Project

- **Develop the project plan assigning ownership and start and expected completion dates to items**

  Figure 13 provides a high level overview of the planned migration milestones for ABSA's Parallel Sysplex data sharing environment. Management should be informed of the project status at these milestones.

- Project planning
- Name changes
- IRLM implementation
- Database conversions
- Building additional IMS systems
- Activating one-way data sharing
- Workload balancing
- Activating two-way data sharing

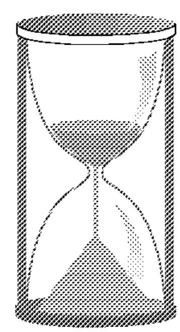


Figure 13. Major Migration Milestones

Now that the project plan is in place with a communication process and corporate commitment behind the migration, all parties can begin to start the move to an IMS data sharing sysplex.

Chapter 3. Environmental Preparation

The installation of IMS in a Parallel Sysplex introduces some new complexities which are discussed in the *IMS/ESA Systems Reference Library* and in Chapters 4, 5 and 6 of this book. Other factors, nonspecific to sysplex technology, that ABSA found necessary to address before and during the data sharing implementation include general "health" of the system environment, increases in complexity that arise from operating multiple IMS systems, increased workload that IMS in a Parallel Sysplex facilitates, and lack of familiarity with parallel sysplex technology.

In this chapter, we review issues pertinent to the ABSA project as well as other issues to consider to ensure a successful IMS Parallel Sysplex implementation. Many of these issues are not directly related to IMS in a parallel sysplex but are just as important for the normal operation of any IMS installation as they are for a sysplex migration. Preparing for an IMS sysplex data sharing migration is simply a good occasion to review the general health of your IMS system and to initiate remedial actions for any problems identified.

In Chapter 1 we examined the key items to review regarding the pre-sysplex environment and the target implementation. It is the understanding developed during the planning segment of the project that forms the base for all activities during this preparation segment.

3.1 System Review

The first step in determining the readiness of an IMS customer for an IMS sysplex data sharing implementation project is a review of the overall health of the system environment, identifying any problem areas that need attention. Some of the areas pertinent in such a review include:

- **User perceptions**

  Do users regard the system as effective and efficient, or are they justifiably dissatisfied with the service they are receiving? Are too many unresolved problems and outages affecting end users?
- **Scheduling efficiency**

  Can the staff responsible for workload scheduling perform their functions efficiently with reasonable conformance to established schedules?
- **System stability**

  Are system outages rare events, and does the overall availability meet user expectations?
- **Error rediscovery and problem management**

  Are errors usually fixed after their first occurrence, or do the same errors tend to persist and recur frequently? Are errors corrected in a timely manner? Is there effective management and technical level supervision for:

  – Hardware failures

  – Systems software failures

  – Operational problems

  – Abends or lockouts associated with application programs

  – Network failures?

© Copyright IBM Corp. 1997

Are an assigned problem number, problem description, severity, and correction logged for these problems? Is trend analysis performed on this data and proactive change implemented when necessary?

- **Software maintenance**

  Are there effective procedures for regular maintenance of system software and for the management of software changes for the main subsystems affected by IMS in a Parallel Sysplex (IMS/ESA, MVS/ESA, and IRLM)? Is the maintenance cycle frequent enough to ensure product currency and potential stability? Are processes in place to periodically review the status of new maintenance available for software products?

- **System performance management**

  Are the performance monitoring and data reduction tools understood by all who might have a need to review the performance of the system? Are instances of poor system performance recognized and corrected promptly? Is system performance reasonable for the workload and facilities used?

The answers to these questions can help planners include any necessary "cleanup" work in the project plan to correct any issues that have been highlighted and that could inhibit a smooth IMS/ESA Parallel Sysplex installation.

3.2 System Management Practices

In view of the complexities of IMS in a Parallel Sysplex implementation, a number of problems must be expected. Suitable system management tools and practices are therefore strongly recommended.

Some form of data collection and reporting is required to facilitate management of problems and ensure that system issues are resolved expeditiously. IBMs TME 10 Information/Management (Infoman) is an example of a problem management software package that would be suitable for such use.

In this discussion we recommend various problem management tools and reports that are designed to assist management to understand:

Problem status A problem status report lists existing problems and their current status. It is designed to reveal whether the acknowledged problems are being corrected promptly, and whether there are undue delays in the collection and supply of diagnostic data to those who will be analyzing the information.

Problem history A problem history report tracks whether problems are occurring in particular areas of system operation and management, and whether there is any change in the rate of occurrence.

Problem trends A problem trends report indicates whether there is any change in the rate or severity of code defects that have occurred.

Note: The content of the sample reports presented in this section is entirely fictitious and provided for illustration only.

3.2.1 Problem Status Report

Customers have found problem status reports useful in the management of software issues in IMS Parallel Sysplex implementation projects. A problem status report shows the status and a summary of the history of each problem that has occurred but is not completely resolved. This report helps expedite the resolution of problems by highlighting where in the problem resolution cycle each problem is and who is responsible for the next step in that cycle. Table 1 shows a sample of the style of report used at ABSA during the project.

A formal problem management process should be followed for all problems and issues that arise, whether they result from in-house issues or IBM or vendor problems. During its IMS data sharing implementation ABSA used a similar system for all sysplex-related software problems and issues during its daily status meetings.

Table 1. Problem Status Report

ABSA Problem No. IBM Problem No.	**Date**	**System**	**Sev**	**Symptom**	**Description**	**Status**	**Days Open**	**Owner**
A6025 0X852	96/06/02	Prod-A	3	ABU3508	Initial dump did not include desired data. IBM supplied slip trap for reccurrence. Slip trap installed 96/06/09. No recurrence yet.	Review 96/6/16	12	A.S.
A6026 0X231	96/6/12	Prod-B	2	Wait	Wait in DFSDN250 Customer reviewing dump	ABSA review	2	D.S.
A6...								
A6...								

A problem status report can be used at regular problem review meetings to establish quickly which outstanding problems require the highest priority and to identify the groups responsible for resolving each problem.

3.2.2 Problem History

Recording and analyzing the history of problem occurrences assists in identifying any components of the system experiencing undue or increasing rates of problems. Typically, problems would be analyzed by a cross-functional group made up of systems and application programmers, operations and automation staff, database administrators, and performance and recovery staff. Any group whose problem rate was high would be alerted and consulted on the most appropriate corrective action. Corrective action might include a review and redefinition of work practices, allocation of additional tools and facilities, or the short-term or long-term provision of additional staff resources or education.

For effective analysis, problems must be diagnosed completely, not stopping at the level of symptoms, but continuing to an understanding of the fundamental problems. As an example, if an IMS system generation fails from a job control language (JCL) error, a superficial error analysis could attribute the problem to a typing error of the system programmer. A deeper analysis might show that the system maintenance procedure is faulty because it requires the system programmer to build JCL manually. A complete fix for the problem might be the introduction of an automated procedure (an ISPF panel or a REXX program) that does away with the need for system programmers to type JCL.

The sample history report shown in Figure 14 is a simplified sample (with dummy data) of a format several organizations have found effective.

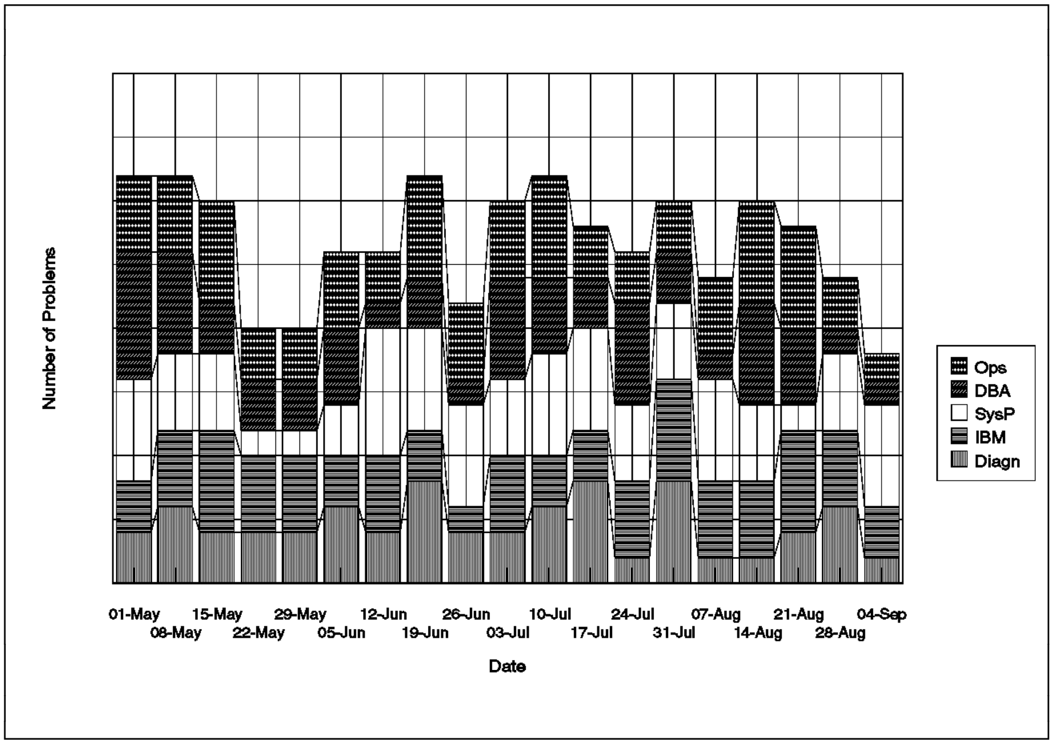


Figure 14. Problem History Report

In the sample report, weekly totals of open problems are categorized by responsible group:

- Diagnostics under way by ABSA and onsite IBM representatives
- IBM support center (either in-country or international)
- ABSA systems programming
- ABSA database administration
- ABSA operations

3.2.3 Problem Trends

Some customers maintain a history of problems in the form of a problem trend report. This report identifies the trends associated with specific products, such as IMS; particular applications; specific automated operational processes; and system or data recovery attempts. The report can be used to determine the effectiveness of earlier corrective activity.

3.3 Operator Skill Levels

A high level of operator skills is required for the operation of a complex application and the simultaneous operation of multiple IMS systems in a Parallel Sysplex configuration, particularly when problems occur. Operator training is an essential component of an IMS data sharing project, and care must be taken to ensure that adequate skills are developed in such aspects of operations as:

- Single IMS system operations
- Use of global commands, and facilities to ensure that all subsystems respond to global commands. This is an area where current sysplex operation is complex, and special training is required.
- Startup and shutdown of multiple IMS systems
- Startup and shutdown of the Parallel Sysplex coupling facility
- IRLM startup and shutdown
- IMS trace enablement and operation
- System dump operation
- Emergency restart operations and procedures

If automated operations take the place of operations personnel, the above areas of focus must be reviewed by those developing automated procedures in conjunction with the data sharing sysplex migration.

3.4 Dump Management

Successful problem analysis requires abend options to be set appropriately so that each system abend produces a system dump containing the required diagnostic data. These dumps are typically fairly large data sets and may pose storage and management problems in a constrained environment. They are important and costly information resources for the problem resolution process. Any loss of a dump is the loss of data about an error or abend, exposing you to a reccurrence of the problem. The availability of all diagnostic data from each failure, and the immediate and systematic diagnosis of that data to complete problem resolution, is the best way to prevent problem reccurrence.

Rapid change can introduce problems through such factors as system software or application errors and operational or automation problems. ABSA was required to review its management of diagnostic data during the course of the migration. In this section we describe some of the activities that resulted from the review.

3.4.1 SLIP Trap Dump Control

Manual entry of dump commands to collect data from multiple address spaces can be quite complex. ABSA followed an initial recommendation to use a series of IEASLPxx members in SYS1.PARMLIB. These members can be activated by using the MVS SET SLIP command when a set of address space dumps is required.

The following values are recommended for dumping local and remote address spaces for IMS when you issue the SET SLIP command:

IMS: The IMS-related member of PARMLIB is SYS1.PARMLIB(IEASLPIM) and should be set to:

```
SLIP SET,IF,ENABLE,A=SVCD,N=(IEAVEDS0),ID=IMSD,ML=1,
JOBLIST=(XCF*,IRLM*,DLI*,DBR*,IMS*),
SDATA=(RGN,XESDATA,ALLNUC,CSA,LSQA,PSA,SQA,SUM,SWA,TRT),
REMOTE=(JOBLIST=(XCF*,IRLM*,DLI*,DBR*,IMS*),
SDATA=(RGN,XESDATA,ALLNUC,CSA,LSQA,PSA,SQA,SUM,SWA,TRT)),END
```

IRLM: The IRLM-related member of PARMLIB is SYS1.PARMLIB(IEASLPIR) and should be set to:

```
SLIP SET,IF,ENABLE,A=SVCD,N=(IEAVEDS0),ID=IRLM,ML=1,
JOBLIST=(XCF*,IRLM*),
SDATA=(RGN,XESDATA,ALLNUC,CSA,LSQA,PSA,SQA,SUM,SWA,TRT),
REMOTE=(JOBLIST=(XCF*,IRLM*),
SDATA=(RGN,XESDATA,ALLNUC,CSA,LSQA,PSA,SQA,SUM,SWA,TRT)),END
```

In the above SLIP SET commands, the IMS*, DLI*, DBR*, IRLM*, and XCF* names represent generic forms of the names used for the IMS, DL/I separate address space (DLISAS), database recovery control (DBRC), IRLM, and XCF address spaces. Module IEAVEDS0 is the MVS dispatcher, so the SLIP will be matched shortly after the SET SLIP= command is issued. This SLIP will be used only when an image of the important address spaces is required as a “snapshot” of the sysplex.

3.4.2 Suppression of Dumps

In some abend cases, dump data for duplicate instances of specific problems may not be required. On some occasions, ABSA used a SLIP suppress trap to prevent repetitive dumps. Later, SUPPRESSALL was coded in the DAE member to reduce the number of duplicate dumps that occurred because of abending components or OEM code that did not have the VRADAE string specified for the suppress option.

IMS can suppress dumps through the use of the DFSFDOT0 table (see the *IMS/ESA Version 5 Customization Guide*, SC26-8020), but ABSA used only the default abend code values.

There were no instances where a dump was suppressed that was later found to be required.

3.4.3 Storage and Indexing of Dumps

Because abend dumps are quite large data sets that are usually used only a few times after they are written, and then not at high I/O rates, they are excellent candidates for effective storage using Hierarchical Storage Management (HSM) storage classes. Automatically allocated dump data sets in a reserved SMS storage class are an effective mechanism for storing abend dumps, with SYS1.DUMP preallocated backups available for emergency use if the SMS class becomes full.

A good technique is to have the dump data sets initially sent to an SMS storage class where the dumps will migrate from DASD to tape after a few days.

The initial examination and indexing of a dump consists of several steps:

1. Allocate a dump identifier.

2. Do a preliminary assessment of the problem and annotate the problem management record with severity and dump-identification data.
3. If the dump is to be accessed within the next few days for problem analysis, move it to a storage class where it can stay on accessible DASD for the period.
4. Otherwise, allow the dump to migrate fairly quickly to offline storage unless it happens to be needed for diagnosis of other problems.

During the project, ABSA did not use an internal problem tracking system that would relate a problem symptom to the problem owner. This caused the following problems:

- Difficulty in matching the occurrence of the problem with the problem record from vendor software support organizations or corrective maintenance numbers
- If the problem occurred infrequently, the original problem would lose focus or be confused with other problems such as "another AB0C4 in IMS."
- It became difficult to determine when a dump could be deleted. This also allowed HSM to migrate the dump data set to tape when there was still a need to have immediate access to the documentation.

Control over the storage, access, and identification of storage dumps improved as a result of the implementation of the above suggestions.

3.4.4 Stand-alone Dumps

Stand-alone dumps have their own set of problems and issues. Because system programmers are physically remote from most of the systems, they must use a remote operations facility to take the dump. Modifications to remote operation setup and configuration changes that affect terminal and tape addresses can interfere with the stand-alone dump procedures. System programmers usually discover this while trying to take a dump. Because production systems may be down at this time, the usual course of action is to skip taking the dump.

Dumps may occur for a problem that is already known. A process must exist to quickly determine whether the documentation being collected has previously been captured. This requires fast and easy access to the status of all currently open problems, as described in 3.2, "System Management Practices" on page 24.

The interactive problem control system (IPCS) facility should be available on all systems where dump or trace analysis is to be performed, rather than transmitting dumps to a system that has IPCS set up. The transmission causes delays during a critical part of the problem resolution process. It also opens a window of exposure for the problem to recur. For example, there may be VTAM problems transmitting or JES problems receiving the dump because of earlier problems in these components.

ABSA support staff originally experienced delays in reviewing diagnostic material such as stand-alone dumps because the dumps had to be transmitted from one system to another.

3.4.5 Other Diagnostic Information

Other sources of diagnostic information are required to solve problems. System log (SYSLOG), log record (LOGREC), and system management facility (SMF) records have to be managed so they can be identified and retrieved easily when needed. The collection of this information can slow problem diagnosis and add frustration when naming conventions are different on each system and it is not obvious which data set contains records for which interval of time. A "dry run" of the collection and review of diagnostic data retrieved from systems in the sysplex should result in effective, time-saving procedures.

3.5 Common System Images, Product and Service Levels

Applications and systems at ABSA are the result of the merger of several installations. This has caused some predictable systems management problems:

- Different data set and VTAM APPLID (application identifier) naming conventions present confusion for programmers and operations.
- Difficulty introducing a common service level from the development to production systems with a controlled process.

The result of these problems can be seen when corrective maintenance or parameter changes applied to one system are not propagated to follow-on systems and the initial problem reoccurs.

The ABSA2 project will merge many of the applications, and the value in cloning and standardization will be worth the investment. The cost of having complicated, undocumented, nonstandardized setups is most noticeable when bringing new people into the organization. If there is only an oral tradition, it is costly in time and accuracy in passing it on. The entire system software installation and maintenance process at ABSA is under review with the goal of standardizing and automating as much as possible so that common system images are easy to obtain.

3.6 System Maintenance Process

The term "application of maintenance" can mean either the addition of fixes to a system or the removal of fixes from the system.

The following maintenance process and associated procedures were recommended to ABSA to achieve the general aims discussed in maintenance coordination recommendations listed above. They were designed to:

- Ensure that all maintenance is applied to multiple components of systems in a coordinated and managed way
- Ensure that the maintenance status of each system is known and documented
- Ensure an orderly process for maintenance upgrades
- Ensure that maintenance upgrades normally progress forward through the hierarchy of systems, from test to production
- Ensure that emergency procedures are available to apply additional maintenance in a systematic way so that changes are not lost.

- Prevent changes that are inconsistent with the defined strategy and increase the risk of incorrect maintenance application

This process is described here in terms of IMS maintenance, but it is equally suitable for all MVS environments, and it is recommended as a standard for the coordinated maintenance of all subsystems involved in a data sharing sysplex project; that is, IMS, MVS, VSAM, JES, VTAM, DB2, and IRLM.

3.6.1 Maintenance Coordination

Experience during the ABSA project confirmed the suitability of some practices that are commonly used to manage system maintenance in a complex environment. Recommendations for maintenance coordination activities are documented below.

1. **Establish a maintenance coordination group**

   The objective of a data sharing sysplex is to use both hardware and software facilities in a logically coupled environment where there is close assignment of function and interface among all components. No longer can a large account view the various operating system and application enablers (like DB2 and IMS) as separate entities because it is the effective functioning of all components together that results in sysplex success.

   Establish a group of system programmers to coordinate system maintenance across all major components of a sysplex installation. Representatives should be drawn from the various departments or subgroups that would update system functions for IMS, DB2, IRLM, MVS, DFP, VSAM, RACF, and VTAM. Although members of this group would still report to their own management path and continue to work in their product speciality, they would meet to map out the current change package for the sysplex and communicate with their peers on product interdependence. In this way communication would be enhanced and the best maintenance skill set would be available to bring the general competency of all members to higher levels.

   A maintenance coordination group provides skills to ensure that the total system configuration is maintained effectively and that all system components can support a data sharing sysplex.

2. **Establish cyclic maintenance packages**

   In the dynamically changing environment that existed during the ABSA sysplex data sharing implementation, it was vital to enforce control over the system changes that were introduced. The introduction of monthly change packages is recommended during this period.

   ABSA has a maintenance team to install all the host-based software. Members of the maintenance team would introduce the maintenance members (usually program temporary fixes (PTFs)) into the change package during their weekly meetings, and that package would be migrated from systems programming machines, through development to production sites. If an individual wants to include additional maintenance after a cutoff date, a separate "emergency change review and installation" procedure should be invoked as discussed in 3.6.3, "Emergency Maintenance Process" on page 33.

3. **Regular service reviews**

Members of the maintenance coordination group should have ready access to tools such as ServiceLink. To understand the current maintenance recommendations associated with the software component under review, ServiceLink should be used on a regular basis to review:

- Preventive service planning (PSP) buckets
- Automatic software alert process (ASAP)
- Lists of high impact or pervasive authorized program analysis report (HIPER APARs)
- Program temporary fixes in error (PE PTFs)

This activity provides group members with an early opportunity to review reported IBM software problems and to order and receive PTFs early if they are urgently required for applications. Although this information is available by contacting the IBM Service organization, the ABSA and IBM team agreed on the benefits of obtaining as much information “inhouse” as possible.

4. **Maintain online status data**

   Maintain an online system updated by all members of the cross support function maintenance team and accessible by any interested group in the customer organization. The system could be as simple as a flat file listing what the current maintenance package is, where it exists on the various systems, and what is planned for the maintenance package currently being developed.

5. **Avoid multiple SMP APPLYs**

   Currently at ABSA, IMS maintenance is applied to:

   a. BSYS
   b. The development system through SMPE
   c. ASYS
   d. Production system

   It is then accepted, used as part of the sysgen process, and reapplied.

   This process requires multiple sets of SMP/E libraries. Although the process functions well, another approach would be to use SMP/E on BSYS and then copy the necessary libraries over to ASYS for sysgen use. This approach would assist in the migration package phasing process.

6. **Integrate processes**

   The maintenance process must include a comprehensive problem tracking and feedback cycle. When a maintenance package is introduced into any system, any problems should be recorded with a tool that can be viewed by other groups and track trends on problem counts, component sources, length of time to resolution, and ownership duration during the life of the problem. When problem resolution occurs, corrections or operation changes must be documented and introduced into the maintenance package. As the maintenance package moves from system to system, any problems discovered and corrected either within the package or with the installation of the package will move with it.

ABSA′s IMS environment included five IMS systems, which are referred to as IMSA through IMSE for the purposes of simplicity in this discussion.

IMSA System programmers initial test bed

IMSB Development system used for most program development and testing
IMSC Stress testing environment
IMSD Production IMS system, which was duplicated for sysplex support during the migration project
IMSE System programmers emergency test bed for testing of emergency fixes.

3.6.2 Normal Maintenance Process

The objective of the normal maintenance process (Figure 15) is for maintenance to be generally applied to IMSA and then migrated sequentially through IMSB, IMSC, IMSD, and IMSE.

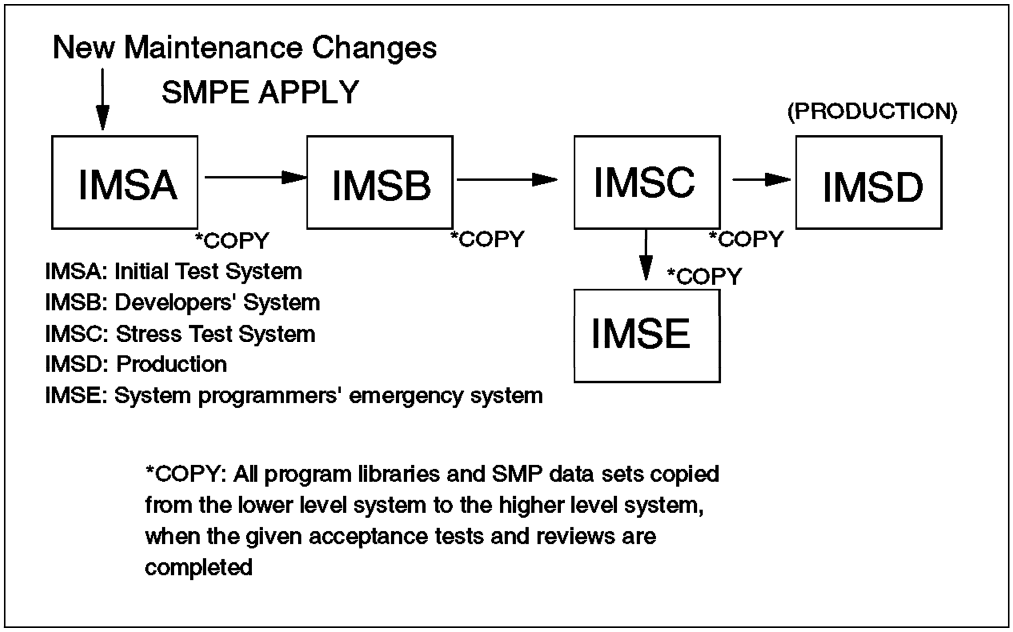


Figure 15. Normal Maintenance Process

All maintenance is to be normally applied to the least tested system and then migrated through to the most thoroughly tested system. Each migration step is to be taken only when the testing at the previous level has completed.

Each migration is to be done as a complete replacement of the higher level system—all libraries and SMP data sets are to be copied from the lower level system.

The trigger for migration of a set of maintenance from one level to another is the satisfactory completion of the exit criteria for the lower level testing. Thus, the maintenance migration system ties in with other aspects of the system management organization.

3.6.3 Emergency Maintenance Process

If a system problem occurs for which the application of maintenance is required ahead of the schedule for normal maintenance, an emergency maintenance process can be used (see Figure 16 on page 34). The steps of the process are described below.

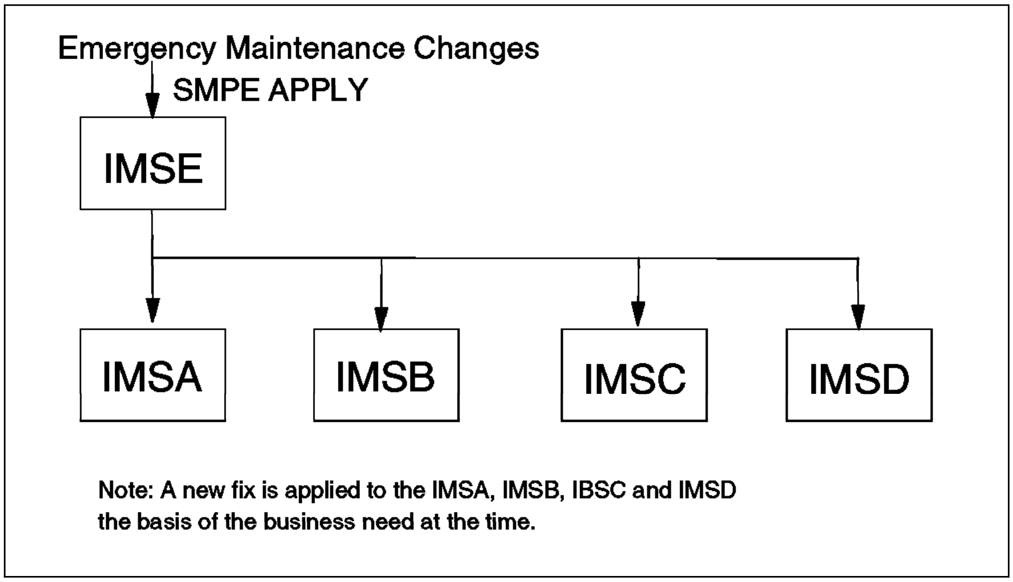


Figure 16. Emergency Maintenance Process

1. System Programming Management authorizes the introduction of specific items of maintenance, at a specific level of the system testing process.

   This authorization includes:
   - Identification of the specific fixes involved
   - Identification of the system level at which the fix is to be initially applied
   - Definition of the level of testing that is required before the fix is moved into the maintenance cycle

   Emergency fixes could be either the rapid escalation of maintenance that is already in the testing cycle or the introduction of new maintenance in response to a specific problem.
2. Apply the fix and test it in the IMSE system, which is at the same maintenance level as the stress test system and possibly the production system.
3. When the testing in IMSE has satisfied the criteria defined for that fix, apply the fix at the required level, which could be any of IMSA through IMSD, depending on the severity of the problem and the urgency of resolution.
4. Apply the fix to all lower-level systems. For example, if the fix is introduced at IMSC, it would also be applied to IMSA and IMSB. Then it would be migrated from IMSC to IMSD when IMSC testing is complete.
5. Build job streams to carry out the above steps in a user friendly and automated way— with no user specification other than "Run the SMP APPLY to C-B-A job for fix PN12345." If necessary, parts of the jobs could be held, pending further checking and testing, but it is important that the job streams perform all steps as defined and do not depend on the user running a combination of jobs.

3.6.4 Control of the Maintenance Process

Because SMP is used to install and maintain major subsystems, it is vital to ensure that the process is not contaminated with outside JCL streams or programs.

The only jobs authorized by RACF (or another security product) to update the SMP and program libraries should be the specific, tested job streams performing the tasks associated with the normal and emergency maintenance processes described above. Other SMP processes that address the relevant libraries should be removed from all job libraries (and those libraries should be checked regularly for “homemade” additions). Development of nonconforming job streams should be strongly discouraged.

The SMP program should be RACF protected to restrict access to authorized users only.

Chapter 4. Preparation for IMS Sysplex

A laboratory sysplex system was created on which the ABSA system programmers did their initial testing. During 1996 ABSA implemented a data sharing sysplex for the development test system (IMSV). IMSV is the system on which final testing was done before anything was moved to the production system at the prime site, so it made sense to create a sysplex environment there. The “conversion” system, which is a production lookalike was used for stress testing. The infrastructure for a sysplex was developed in parallel at the prime site, and once ABSA was happy that the sysplex “worked” on the development system, it moved to data sharing sysplex mode on the production system.

In this chapter we focus on the specific tasks associated with preparing for the implementation phase of the IMS data sharing sysplex environment. Many of these activities can and should be performed in parallel, to reduce the total time required for the project.

4.1 Preparation Steps Associated with IMS System Environment

A number of tasks are required to prepare your IMS systems and environment for sysplex data sharing. This section offers suggestions on naming standards for IMS objects, lists the data sets that can and cannot be shared, and describes of the tasks for migrating from IRLM Version 1.5 to IRLM Version 2.1 and the setup of the coupling facility.

4.1.1 Develop Naming Conventions

STEP 5 (See page 85)

The first logical preparation step is to develop naming conventions for IMS facilities. The conventions would include naming standards for IMS data sharing groups, IMS subsystems, IMS system data sets, IMS region job names, IMS region started tasks, and the coupling facility structures. It is in everyone′s best interest not to modify the current installation naming conventions, but, where necessary, modifications could assist in operations within the sysplex environment.

Note: The naming conventions contained in the project schedule in Appendix A, “ABSA IMS Parallel Sysplex Project Plan” on page 81 relate to the development test systems at ABSA. The naming conventions we refer to in this text relate to the production environment because the sample procedures and control statement streams we use come from the production libraries. The project schedule relates to naming conventions used for the development test systems because it was created and used first for that migration. When the production migration occurred, ABSA was very familiar with the process so the specific naming conventions were not repeated in the project schedule.

© Copyright IBM Corp. 1997

4.1.2 Naming Standards

STEP 7 (see page 85)

ABSA had to come up with new naming standards to identify these components of its IMS sysplex:

- Data sets (shared and nonshared)
- Regions
- Started tasks
- IMS identifiers (IMSIDs)
- Coupling facility structures

4.1.2.1 Shared IMS Data Sets

STEP 8 (see page 85)

IMS data sets that are shared between IMS subsystems in the same data sharing group were prefixed with IMSP. For example, IMSP.DBDLIB was shared between IMS subsystems with IMSIDs of IMSP and IP02.

The following data sets were shared in ABSA′s environment:

- DBDLIB
- FORMAT/A/B
- PGMLOAD
- PGMLOAD.LOADERS
- PROCLIB
- UPROCLIB
- RECON 1/2/3 *
- REFERAL
- TFORMAT
- USOURCE
- USOURCEN
- MATRIX/A/B
- MODBLKS/A/B
- URESLIB (dsn of IMSPRD.URESLIB)
- UPARMLIB

Note: * The RECON 1, 2, and 3 data sets must be shared.

4.1.2.2 Nonshared IMS Data Sets

Data sets that are unique (nonshared) between the IMS production subsystems at ABSA begin with IPOX.*, where X identifies the IMS subsystem to which the data set belongs. For example, IPO1.POLDS01 is the online log data set for IMSP, and IP02.POLDS01 is the online log data set for IP02.

The following data sets (prefixed by IP01.* or IP02.*) were nonshared in the ABSA environment:

- ACBLIB/A/B [1]

- PSBLIB/A/B [1]
- POLDS01/2/3/4/5/6/7/8/9 [2]
- SOLDS01/2/3/4/5/6/7/8/9 [2]
- WADS1/2/3/4 [2]
- IMSMON [2]
- DFSTRA01/2
- QBLKS [2]
- SHMSG/1/2/3 [2]
- LGMSG/1/2/3 [2]
- MODSTAT [2]
- MSDBCP1/2 [2]
- MSDBDUMP [2]
- BACKUP.MSDBINIT (0) [2]
- MSDBINIT [2]
- RDS [2]
- TCFSLIB
- SYS01/2/-/22(these are the TM SYSOUT DS)
- RESLIB (dsn of SYSM.IP01.RESLIB or SYSM.IP02.RESLIB)
- JOBS
- JCLLIB

Note: 1. These data sets are nonshared, because of the solution chosen for databases with sequential dependent segments. See 4.2.3, "DEDBs with SDEPs: Conversion for Data Sharing" on page 51 for a description of this solution. Had it not been necessary to share the SDEPs, these two data sets would have been shared across the sysplex.

Note: 2. These data sets must be nonshared.

4.1.2.3 Started Tasks

STEP 11 (see page 89)

Table 2 lists the naming conventions in the production data sharing group for started tasks at ABSA.

Table 2. Started Task Naming Conventions

Task	Name in IMSP	Name in IP02
Control Region	P01IMS51	P02IMS51
DLI	P01DLIS	P02DLIS
DBRC	P01DBRC	P02DBRC
IRLM	P01IRLM	P02IRLM

4.1.2.4 Coupling Facility Structures

The IMS coupling facility structure names at ABSA are IMSPIRLM, IMSPOSAM, and IMSPVSAM.

4.1.3 Migrate Lock Manager from Program Isolation to IRLM 2.1

STEPs 14 through 16 (see pages 90 through 90)

In January 1996, ABSA implemented IRLM 2.1 on all of its development IMS systems with SCOPE=LOCAL. In May 1996 ABSA implemented IRLM 2.1 with SCOPE=LOCAL on its productions IMS systems.

ABSA uses automation to ensure that IRLM is recycled whenever IMS is recycled. IRLMs are not “shared” among multiple subsystems. Figure 17 presents an overview of the IRLM 2.1 implementation activities.

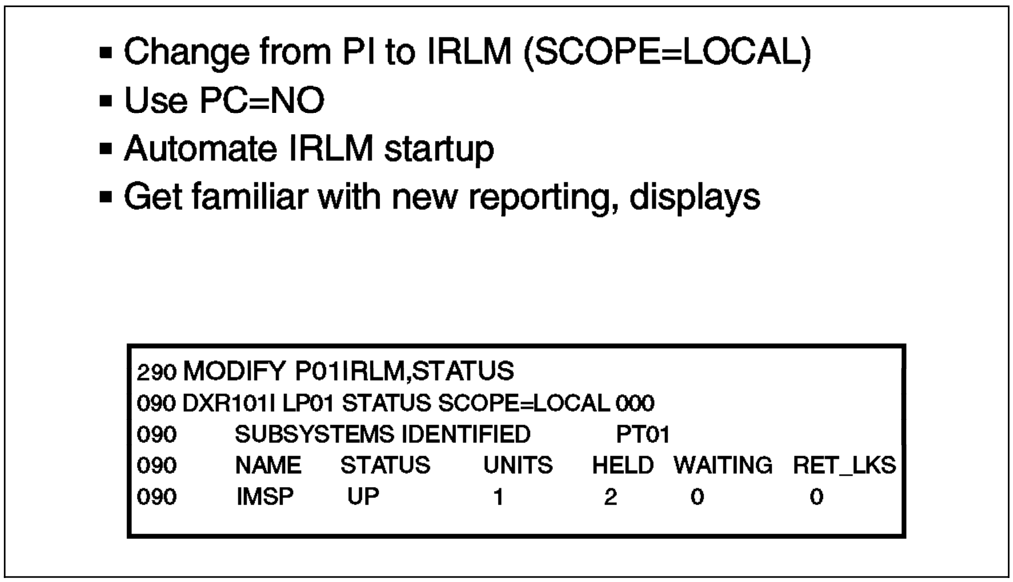


Figure 17. Early IRLM 2.1 Implementation Activities

4.1.3.1 MVS Subsystem Definition Tasks for IRLM

The following MVS-related activities are associated with the implementation of IRLM as a subsystem:

1. Add the IRLM entry to the program properties table (PPT)

   MVS preconditioning should have already defined a PPT entry for DXRRLM00. If it has been deleted, define a unique MVS subsystem name in MVS for each IRLM subsystem that runs on that MVS. For more information see *MVS/ESA System Modifications*, GC28-1831. Figure 18 on page 41 is the PPT that ABSA used for IRLM.

```
 /*                         IRLM - RESOURCE LOCK MANAGER
PPT PGMNAME(DXRRLM00)  /* BITS WERE '6870FFFF00000000'
         CANCEL        /* PROGRAM CAN BE CANCELLED         (DEFAULT)
         KEY(7)        /* PROTECT KEY ASSIGNED IS 7
         NOSWAP        /* PROGRAM IS NOT-SWAPPABLE
         NOPRIV        /* PROGRAM NOT PRIVILEGED           (DEFAULT)
         DSI           /* DOES NOT REQUIRE DATA SET INTEGRITY (DFLT)
         SYST          /* PROGRAM IS A SYSTEM TASK
         PASS          /* CAN BYPASS PASSWORD PROTECTION
         AFF(NONE)     /* NO CPU AFFINITY                  (DEFAULT)
         NOPREF        /* NO PREFERRED STORAGE FRAMES    (NODEFAULT)
 /*                         IRLM - RESOURCE LOCK MANAGER
```

Figure 18. ABSA's Program Properties Table for IRLM

ABSA's PPT produces the following properties:

- PPTNAME=DXRRLM00
- PPTBYTE1=X'68', which indicates unique protection key, nonswappable, nontimed, system task
- PPTKEY=X'70', which defines the protect key as 7
- PPTCPUA=X'FFFF', which states that central processing unit (CPU) affinity is not required

2. Assign the IRLM group to a cross-systems coupling facility (XCF) transport class with a message length of 395 bytes. ABSA specified a CLASSDEF with GROUP(IRLMP) and CLASSLEN(395) in the COUPLExx member of SYS1.PROCLIB to provide for optimized message lengths and signaling paths owned by the group.

4.1.3.2 IRLM 2.1 Startup Procedure

There are several new or changed parameters in the startup procedure for IRLM V2.1. Figure 19 shows the P01IRLM procedure used to start up IRLM on the first ABSA data sharing production system.

```
//DXRJPROC PROC RGN=40M,
//              IRLMNM=LP01,
//              IRLMID=1,
//              SCOPE=GLOBAL,
//              GROUP=IRLMP,
//              DEADLOK='2,1',
//              PC=NO,
//              MAXCSA=40,
//              MAXUSRS=2,
//              LOCKTAB=
//        EXEC  PGM=DXRRLM00,DPRTY=(15,15),
//        PARM=(&IRLMNM,&IRLMID,&SCOPE,&DEADLOK,&MAXCSA,
//        &PC,&MAXUSRS,&GROUP,&LOCKTAB),REGION=&RGN
//SYSABEND DD SYSOUT=A
```

Figure 19. Procedure P01IRLM Used to Start Up IRLM on One Production System

Below we review the parameters in light of their use at ABSA. For more information refer to *IMS/ESA Version 5 Installation Volume 2: System Definition and Tailoring*, SC26-8024.

IRLMNM=

IRLM requires a 4-byte MVS subsystem name for its internal processing. The standing recommendation is to use the same subsystem name for all IRLMs in the data sharing group so that IMS subsystems and jobs can be moved without having to move the IRLM. At ABSA, the IRLMNM value set in P02IRLM (the IRLM startup procedure for the second IMS production data sharing system on a separate processor) is LP02. The IRLMNM value set for the first IRLM is LP01. When two IRLMs reside in the same MVS system, each must have a unique MVS subsystem name.

IRLMID=

This value must be different for each IRLM in the data sharing group. Therefore the startup procedure for the IRLM running on the second IMS data sharing system has a value of IRLMID=2 in its IRLM startup procedure P02IRLM.

SCOPE=

SCOPE=LOCAL had originally been specified when sharing was limited to intrasystem and XCF and SLM were not required or used. With SCOPE=GLOBAL specified, intersystem sharing can be performed, both XCF and SLM are required, and the coupling facility is used.

GROUP=

This is the XCF group to which this IRLM belongs. All IRLMs in this group have to specify the same LOCKTABL parameter, and the IRLMID values must be unique within the group.

DEADLOK=

At ABSA, the DEADLOK parameter value was set at DEADLOK=′2,1′ rather than the default. The lower the values, the better the concurrency and performance but the higher the CPU processing costs. The minimum values generally give the best performance results. These values were the same on both IRLMs in the production data sharing group.

PC=

ABSA used the PC=YES option initially even though it adds CPU path length as it moves the IRLM locks to extended private (EPVT) from the extended common storage area (ECSA) and uses the facilities of the program call (PC) instruction for communication. PC=NO was later introduced because it provides the shortest code path length and is recommended for performance. With PC=YES, the IRLM region size through the startup RGN= parameter determines the amount of storage for IRLM control blocks. ABSA allocated an RGN value of 40 MB of storage in conjunction with their earlier use of PC=YES.

If IRLM experiences an out-of-storage condition, ABENDU3300 occurs for applications requesting locks. ABSA had to ensure that all BMPs issued sufficient checkpoints to release locks and storage. Although a backout of the failing BMP will release the storage and locks, until the backout is complete other requests for locks will probably abend on ABENDU3300 or ABENDU3303s.

MAXCSA=

This parameter is used because PC=NO was specified. The MAXCSA value of 40 specifies the maximum amount of ECSA (in megabytes) that IRLM is to use for its dynamic control block structures.

MAXUSERS=

Each MAXUSERS parameter for the two IRLMs used in the final test development and production systems, respectively, had the same value of 2 because batch workload is not part of the sysplex and both production and development test had two data sharing members in each of their respective groups. This MAXUSERS value is used for calculating the lock hash table entry size. If the parameters did not have the same value, the highest value specified for an active IRLM would be used in the hash table entry size calculations for all of the IRLMs.

LOCKTABL=

LOCKTABL is an optional parameter for specifying the lock table to be used by the group, and ABSA decided not to specify it here. Instead it used the CFNAMES statement in the DFSVSMxx member in PROCLIB. In fact this parameter is not used when the CFNAMES statement is set.

DPRTY=

The dispatching priority of IRLM was set at (15,15) so that it had the highest priority of all the IMS started tasks and regions. If the dispatching priority is set lower than other IMS tasks, then the IRLM could be delayed from servicing requests and cause more accumulation of storage in extended common storage area (ECSA) than necessary. Lock releases could also be delayed, resulting in a sysplex-wide slow down for the IMS data sharing partners.

4.1.3.3 IRLM V2.1 Diagnostics

Because the commands and trace and reporting facilities in IRLM differ from those of program isolation (PI), a certain amount of time was required to become familiar with the differences. For example the IRLM V2.1 diagnostic trace uses the MVS component trace (CTRACE) facility for problem diagnosis. We cover this trace in more detail in 4.2.7, “Ensure Diagnostic Procedures and Facilities Are in Place” on page 55.

4.1.4 Design the Coupling Facility IMS Environment

STEP 21 (see page 91)

ABSA uses two coupling facilities in production (one for backup purposes). The vast majority of data is shared between the two cloned IMS systems in the sysplex. CFC1 is the name of the first coupling facility. It is a D/T 9674 coupling facility. The second coupling facility, CFC2, is partitioned within a D/T 9672.

ABSA′s coupling facility structures for IMS/ESA Version 5 are:

- An IRLM lock structure consisting of a lock hash table and a lock list.
- A VSAM cache structure for buffer invalidates
- Overflow sequential access method (OSAM) cache structure for buffer invalidates (although ABSA does not use OSAM support for databases).

The following references detail how to determine the size of the structures in a coupling facility:

MVS/ESA Version 5 Sysplex Migration Guide, SG24-4581

OS/390 MVS Setting Up a Sysplex, GC28-1779

OS/390 MVS Programming: Sysplex Services Guide, GC28-1771

IMS/ESA Sysplex Data Sharing, SG24-4303

ABSA has a coupling facility resource management (CFRM) policy for defining the coupling facility structures. The administrative data utility, IXCMIAPU, is used for defining the CFRM policy. Generally the steps ABSA followed were:

1. Create the couple data set with the IXCL1DSU utility. Figure 20 on page 45 shows the JCL and control statement.

2. Define the name of the coupling facility and the structures to the CRFM policy. Also define preference and exclusion lists and the amount of dump space in the coupling facility for dumping coupling facility structure data. Figure 21 on page 46 shows the JCL and a portion of the control statements.

3. Run the IXCMIAPU utility to place the CFRM policy into the primary CFRM couple data set.

4. Issue the following command from any active system in the sysplex to activate the CFRM policy:

   SETXCF START,POLICY,TYPE=CFRM,POLNAME=policy name

   Issue the following command to verify that each MVS image that requires connectivity is connected to the coupling facility:

   D XCF,CF,CFNAME=name

   Chapter 7 contains more information about the MVS Display XCF command.

```
//STEP1    EXEC PGM=IXCL1DSU
//STEPLIB  DD   DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD   SYSOUT=A
//SYSIN    DD   *
     DEFINEDS SYSPLEX(APPLEX01)
              DSN(SYS1.XCF.CDSPRI) VOLSER(NSA900)
              MAXSYSTEM(32)
              CATALOG
          DATA TYPE(SYSPLEX)
               ITEM NAME(GROUP) NUMBER(99)
               ITEM NAME(MEMBER) NUMBER(50)
     DEFINEDS SYSPLEX(APPLEX01)
              DSN(SYS1.XCF.CDSALT) VOLSER(NSA901)
              MAXSYSTEM(32)
              CATALOG
          DATA TYPE(SYSPLEX)
               ITEM NAME(GROUP) NUMBER(99)
               ITEM NAME(MEMBER) NUMBER(50)
     DEFINEDS SYSPLEX(APPLEX01)
              DSN(SYS1.XCF.CFRMPRI) VOLSER(NSA900)
              CATALOG
          DATA TYPE(CFRM)
               ITEM NAME(POLICY) NUMBER(6)
               ITEM NAME(CF) NUMBER(5)
               ITEM NAME(STR) NUMBER(32)
               ITEM NAME(CONNECT) NUMBER(32)
     DEFINEDS SYSPLEX(APPLEX01)
              DSN(SYS1.XCF.SFMPRI) VOLSER(NSA900)
              CATALOG
              DATA TYPE(SFM)
               ITEM NAME(POLICY)  NUMBER(9)  /* # OF ADMINISTRATIVE
                                                POLICIES THAT WILL
                                                FIT IN THE
                                                COUPLE DATA SET
                                                DEFAULTS TO 9       */
               ITEM NAME(SYSTEM)  NUMBER(32) /* # OF SYSTEMS ELEMENTS
                                                THAT WILL FIT
                                                IN EACH POLICY
                                                DEFAULTS TO 8.      */
               ITEM NAME(RECONFIG) NUMBER(4) /* # OF RECONFIG ELEMENTS
                                                THAT WILL FIT
                                                IN EACH POLICY
                                                DEFAULTS TO 0.      */
     DEFINEDS SYSPLEX(APPLEX01)
              DSN(SYS1.XCF.SFMALT) VOLSER(NSA901)
              CATALOG
     DATA TYPE(SFM)
               ITEM NAME(POLICY)  NUMBER(9)  /* # OF ADMINISTRATIVE
                                                POLICIES THAT WILL
                                                FIT IN THE
                                                COUPLE DATA SET.
                                                DEFAULTS TO 9.     */
               ITEM NAME(SYSTEM)  NUMBER(32) /* # OF SYSTEMS ELEMENTS
                                             THAT WILL FIT
                                             IN EACH POLICY
                                             DEFAULTS TO 8.        */
               ITEM NAME(RECONFIG) NUMBER(4) /* # OF RECONFIG ELEMENTS
                                                THAT WILL FIT
                                                IN EACH POLICY
                                                DEFAULTS TO 0.     */
/*
```

Figure 20. Creation of the Couple Data Set and Couple Data Set for CFRM

```
//STEP20   EXEC PGM=IXCMIAPU
//...
//SYSIN    DD   *
     DATA TYPE(CFRM) REPORT(YES)
     DEFINE POLICY NAME(PRODPOL) REPLACE(YES)
        CF NAME(CFC1)
               TYPE(009674)
               MFG(IBM)
               PLANT(02)
               SEQUENCE(000000040142)
               PARTITION(1)
               CPCID(00)
               DUMPSPACE(2000)
        CF NAME(CFC2)
               TYPE(009672)
               MFG(IBM)
               PLANT(51)
               SEQUENCE(000000065726)
               PARTITION(1)
               CPCID(00)
               DUMPSPACE(2000)
        STRUCTURE NAME(COUPLE_CKPT1)
               SIZE(8192)
               PREFLIST(CFC1)
               EXCLLIST(COUPLE2_CKPT1)
        STRUCTURE NAME(COUPLE2_CKPT1)
               SIZE(8192)
               PREFLIST(CFC2)
               EXCLLIST(COUPLE_CKPT1)
        STRUCTURE NAME(IRRXCF00_P001)
               SIZE(2304)
               PREFLIST(CFC1,CFC2)
        STRUCTURE NAME(IRRXCF00_B001)
               SIZE(2304)
               PREFLIST(CFC2,CFC1)
        STRUCTURE NAME(IXCPLEX_PATH1)
               SIZE(1024)
               PREFLIST(CFC1,CFC2)
               REBUILDPERCENT(1)
        STRUCTURE NAME(IXCPLEX_PATH2)
               SIZE(1024)
               PREFLIST(CFC2,CFC1)
               REBUILDPERCENT(1)
        STRUCTURE NAME(IXCPLEX2_PATH1)
               SIZE(1024)
               PREFLIST(CFC1,CFC2)
               REBUILDPERCENT(1)
        STRUCTURE NAME(IXCPLEX2_PATH2)
               SIZE(1024)
               PREFLIST(CFC2,CFC1)
               REBUILDPERCENT(1)
        STRUCTURE NAME(IMSPIRLM)
               SIZE(40960)
               PREFLIST(CFC2,CFC1)
               REBUILDPERCENT(1)
        STRUCTURE NAME(IMSPOSAM)
               SIZE(1024)
               PREFLIST(CFC1,CFC2)
               REBUILDPERCENT(1)
        STRUCTURE NAME(IMSPVSAM)
               SIZE(59392)
               PREFLIST(CFC2,CFC1)
               REBUILDPERCENT(1)
```

Figure 21. Definition of the CFRM Policy at ABSA

4.1.5 Examination and Modification of Support Procedures

STEP 23 (see page 92)

Many of the support processes have to be examined and modified. We touch on some of the major system support elements ABSA had to examine.

In the IMS system definition process, some system definition parameters have to be specified differently for each IMS in the data sharing group, and others have to be the same:

- **Sysgen parameters that have to be different**

  IMSID on the IMSCTRL macro

  ABSA did not provide an IMSID on the IMSCTRL macro, so the IMSID used as a default is the name of the VTAM access control block (ACB): IMSP for the production system, and IP02 for the production clone.

  DBRCNM on the IMSCTRL macro

  The names of the cataloged procedures to start the DBRC address space are V01DBRC and V02DBRC for the development systems and P01DBRC and P02DBRC for the production systems.

  DLINM on the IMSCTRL macro

  At ABSA the names of the cataloged procedures to start the DLI SAS separate address space (DLISAS) are P01DLIS and P02DLIS for the primary production IMS systems and V01DLIS and V02DLIS for the development test systems.

- **Other parameters that have to be different**

  Unique initiators have to be assigned for each IMS region so that job streams are directed to the correct processing location.

  Different time controlled operation (TCO) members must be set for each system. For example, a job to be initiated by TCO was developed to open databases such as DEDBs with SDEPs set at SHARELVL(1) that have affinity to one specific system.

- **Sysgen parameters have to be the same**

  ACCESS in the DATABASE macro

  The default for the ACCESS parameter is EX (exclusive). If any database is to be shared, the ACCESS parameter has to be changed from EX on all IMS generation stage 1 input decks. ACCESS=EX is not valid in combination with SHARELVL(3) in an IMS data sharing sysplex environment.

  IRLM=Y on the IMSCTRL macro

  When IRLM=Y is set, that will be the standard for all online and batch jobs.

  TRANSACT and APPLCTN macros

  Both the TRANSACT and APPLCTN macros had to be checked at ABSA to ensure they provided the same function in a shared environment as they did in a single IMS system.

  – TRANSACT macro

The MAXRGN parameter is enforced only within each single IMS. Therefore, even though a particular application is scheduled in only one dependent region per IMS system, it still could be scheduled in a parallel mode within the sysplex. This might be counter to the original application execution design specifications.

The SERIAL parameter is effective only within its own system. SERIAL=YES forces processing of transactions in the order of their arrival.

– APPLCTN macro

SCHDTYP=SERIAL is used to stop parallel scheduling of a program specification block (PSB) and is not effective across systems.

External scheduling or application layer changes might have to be investigated to ensure that unwanted parallel scheduling does not occur. Changes were not required for the ABSA applications.

4.2 Database and Application Considerations

It may be necessary to modify your IMS databases and applications to gain the most value from sysplex data sharing. This section describes what to look for in your applications, in preparation for the move to sysplex data sharing. We look at application affinities (where the application needs to be on the same system as some other resource), the conversion of DEDBs with SDEPs so they can be shared across the sysplex, conversion of MSDBs to HDAM databases, the performance of HDAM databases, disaster recovery plans, and the collection of diagnostic information.

STEP 6 (see page 85)

4.2.1 Identifying Affinities and Potential Deadlocks While Data Sharing

There are two major classifications for IMS subsystems:

Cloned

All applications and databases are available to all IMS systems in the data sharing group, and each IMS system is a clone of the others. When applications are cloned, they run on all IMS systems in the data sharing group.

Partitioned

Each IMS subsystem runs its own set of applications, and some of the databases can be shared.

ABSA implemented sysplex data sharing with a design of cloned applications, with the vast majority of databases being shared. Databases such as MSDBs and DEDBs with SDEPs that did not conform to the data sharing environment were converted to structure types that could be included in the data sharing environment.

Although ABSA attempted to share databases as much as possible, the opportunity to use ACCESS=EX was there to have one IMS system exclusively own a database. The database would then be registered with SHARELVL(O) in DBRC.

Some applications may have affinities to the system on which they run, and that will inhibit running multiple copies of the application in parallel. As well, database design might present performance problems when accessed by multiple IMS systems. ABSA performed a review of those applications and database structures it thought would be prone to problems operating in a data sharing mode.

The following two publications contain more information on where and how to identify affinities in applications and databases:

OS/390 Parallel Sysplex Application Considerations Presentation Guide, SG24-4743

IMS/ESA Sysplex Data Sharing, GG24-4303

Here are ABSA′s major targets for investigation:

- **Use of data in storage by systems software or applications**

  If an application keeps information in a storage resident table, that table will be available only to those applications operating on that MVS image. If the table is read only, multiple copies could be made available to each MVS image, but synchronization of table update (new rate levels, for example) would have to maintained. For very active updated tables, another approach could be conversion into a shared database, but that could lead to contentions under load.

- **Access to nonshared resources**

  In IMS/ESA Version 5, MSDBs, DEDBs with SDEPs, and DEDBs using the virtual storage option (VSO) cannot be shared among multiple IMS systems. A second example of access to nonshared resources is programs that interface directly to hardware associated with only one MVS image. At ABSA, an optical disk imaging application exhibited these characteristics.

- **Transaction workload balancing**

  Because IMS/ESA Version 5 does not support VTAM generic resources, terminal linkages are to a particular IMS system. This might overload the capacity of the specific machine running the "networking" IMS. With multiple system coupling (MSC) routing, however, workload balancing can be achieved. In addition, networks can be split, as is the case at ABSA where every successive terminal in each branch is assigned to a different IMS system, with one of two systems to choose from currently. Therefore ABSA divides the network workload between the two cloned IMS systems.

- **Small databases with few records**

  Frequent access or updating of the same segment range, CI, or block by many IMS programs running in multiple IMS systems leads to what is called *hot spot contention*. Databases with this design, and sometimes the applications that access them, may have to be modified. This was the case at ABSA, as we discuss in 4.2.5, "HDAM Performance" on page 54.

- **Databases with ascending or descending keys**

  New occurrences of accessed segments could be located next to the last segment in the same CI or block, causing contention between data sharing partners.

- **Excessive numbers of DL/I DB calls within applications**

The lack of the use of path calls, issuing multiple calls for the same segment, or use of unqualified get next rather than fully qualified get unique DL/I calls increases contention and locking rates. Applications experiencing problems in a data sharing environment could be candidates for programming review and possible redesign.

- **Evidence of many deadlocks**

  Deadlocks sometimes arise because of poor database design, and any more than a few deadlocks per day should be investigated. The ABENDU0777 code logged as a type 67FF record in the IMS log indicates that a deadlock has occurred. We review deadlocks in more detail in 4.2.7, “Ensure Diagnostic Procedures and Facilities Are in Place” on page 55 because ABSA did experience many deadlocks during production periods.

- **Lack of checkpointing in BMPs**

  The time interval between BMP checkpoints should be a few seconds at most, and the IMS facility to suppress the master terminal operator (MTO) message for checkpoints should, as ABSA did, be exercised.

  There are two ways of suppressing the DFS681I message:

  1. The dependent region parameter:

     CKPTID=′NOMSG′

  2. The systemwide OPTION in DFSVSMxx:

     OPTION ISSUE681=NONE or **xx**

     where xx is the number of checkpoint messages per second per program that can be issued before the messages are not shipped. The default is ISSUE681=ALL. Another message, MSGDFS683I, might be issued to indicate the number of DFS681I messages that were skipped. This will prevent any single region from flooding ECSA with DFS681I messages and indicate that a BMP region may be looping.

- **Processing option (PROCOPT) not used effectively**

  If processing is truly read only and the application can accept a “dirty read” associated with PROCOPT=GON, the locking rate decreases. Even using Procopt=G rather than Procopt=A decreases the scope and rate of locking.

- **Databases in need of tuning**

  Although each database type has unique characteristics, if contention associated with small CIs or blocks, limited free space, or serialized key chains is detected, redesign might be in order.

Besides the group of applications we discuss in this chapter, ABSA did not have to redesign many databases or applications to make them efficient in the data sharing environment.

4.2.2 Data Sharing Positioning Stage

STEPS 24 and 25 (see pages 92 through 92)

The fundamental goal of the ABSA data sharing migration was to be able to share as much of the corporate data resources as possible. In order for all if the data to be fully “shareable,” circumventions had to be implemented for those database types which are not yet shareable at the IMS/ESA Version 5 level, namely, DEDBs with SDEPs and MSDBs.

Figure 22 on page 51 is an overview of ABSA's database conversion efforts.

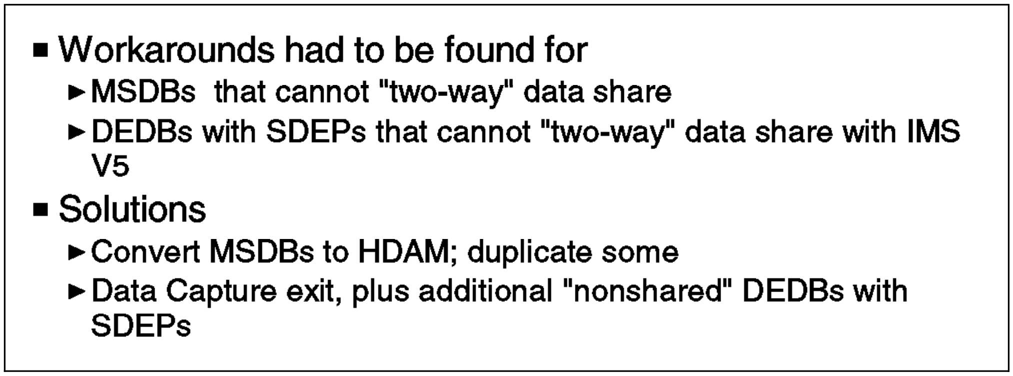


Figure 22. Overview of Database Conversion Activity

4.2.3 DEDBs with SDEPs: Conversion for Data Sharing

STEPS 26 to 27 (see pages 93 through 93)

Deciding on a solution for DEDBs with SDEPs (which are used at ABSA for journaling) was not an easy task. Three implementation solutions were used:

- Duplication of the DEDBs on the two systems
- Use of the Data Capture exit to copy the inserted data to the new SDEPs and delete the direct dependent segments (DDEPs) inserted by application programs
- Creation of new duplicated unshared DEDBs with SDEPs

4.2.3.1 SDEP Solution 1: New DEDBs with SDEPs

Ten DEDBs with SDEPs were targeted for this data sharing solution (Figure 23). These DEDBs were targeted because the SDEPs are used extensively and since the applications development department at ABSA was already making changes to the systems that use these databases.

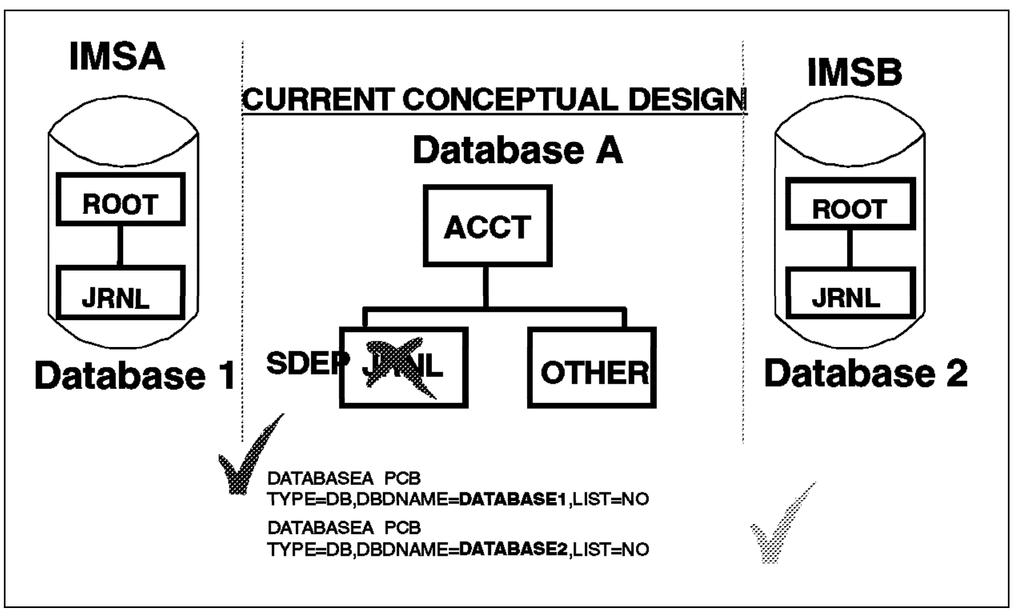


Figure 23. First DEDB (SDEP) Data Sharing Solution

- Database A is the existing DEDB that was defined with SDEPs.
- Database 1 is a new DEDB with SDEPs defined, but it is nonshared and used only by transactions running on IMSA.
- Database 2 is a new DEDB with SDEPs defined, but it is nonshared and used only by transactions running on IMSB.

The database definition (DBD) for Database A was first modified to remove the references to any SDEP. The rest of the DBD remained the same. This DEDB was now shareable between IMSA and IMSB.

When applications wanted to insert an SDEP (which originally had gone to Database A), the application interface block (AIB) interface was used, and, depending on where the transaction was running (either IMSA or IMSB), an SDEP was inserted on either Database 1 or 2. The inserted SDEP is under a root segment, which is keyed on "Region ID." The program communication block (PCB) labels for Database 1 and 2 are the same, but the database referred to differs depending on which subsystem it is run. This allowed the use of the same application for both IMS systems, with only different PSB and ACB libraries. Obviously all available data in Database A is shareable.

Two DEDB SDEP SCAN jobs were required to extract the SDEPs from Database 1 and 2, respectively. The output was combined (appended, or merged with a sort, depending on requirements).

4.2.3.2 SDEP Solution 2: Use of the Data Capture Exit

This solution was used for four DBDs (Figure 24). The database definitions were the same as in the previous solution, but ABSA was unable to modify the applications using these databases, so the Data Capture exit for the SDEP inserts to Database 1 and 2 had to be used. The coded exit and sample DBD input are listed in Appendix B.

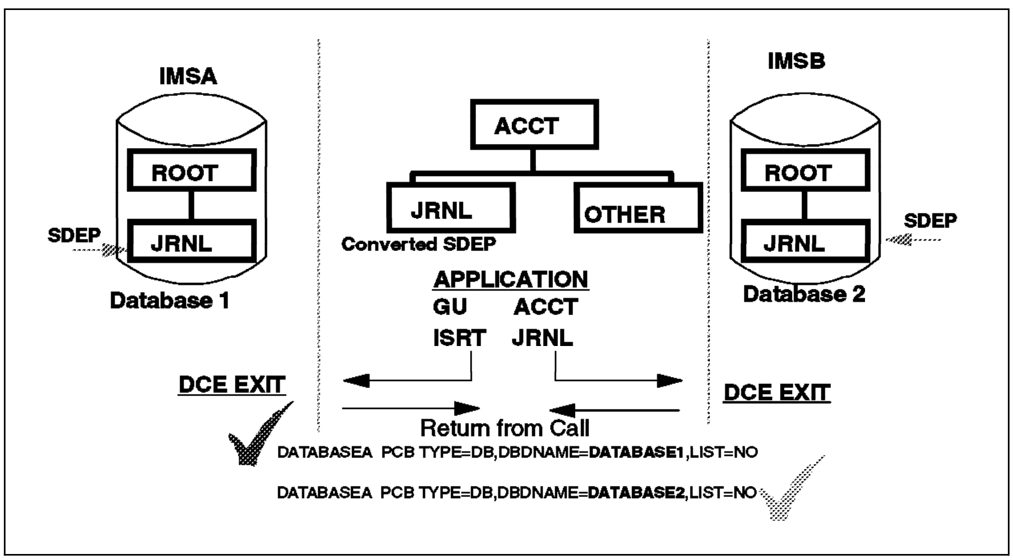


Figure 24. Second DEDB (SDEP) Data Sharing Solution

The SDEP for Database A was changed to a DDEP (referred to in Figure 24 as a *converted SDEP*) on the DBD. A Data Capture exit was also defined on the SEGM statement for this converted SDEP. When the application inserts a converted SDEP to Database A, the exit is invoked. It first inserts an SDEP on either Database 1 or 2 and then deletes the converted SDEP that was inserted on Database A, within the same unit of work (UOW). An application had to be written to merge these duplicated DEDBs with SDEPs for end-of-day processing.

These actions enable the application to operate as if it had moved the old SDEP to the new unshared DEDB area. The newly defined DDEP segment is not actually stored in the database except during the brief processing of these calls.

The second solution enabled ABSA to share Database A and did not require any changes be made to the applications. It is, however, not as efficient as the first solution.

4.2.3.3 SDEP Solution 3: Creation of New Duplicated Unshared DEDBs with SDEPs

This solution was used for the other 39 DEDBs with SDEPs (Figure 25). These DEDBs have simple hierarchical structures and were used solely for the SDEP capability. ABSA was therefore able to duplicate the DEDBs on the two systems. The DBD specified on the PCB label determines which database will be updated. The SDEP updates from both IMS systems were input to two SDEP extract jobs and then another sort/merge job was run to obtain the necessary information.

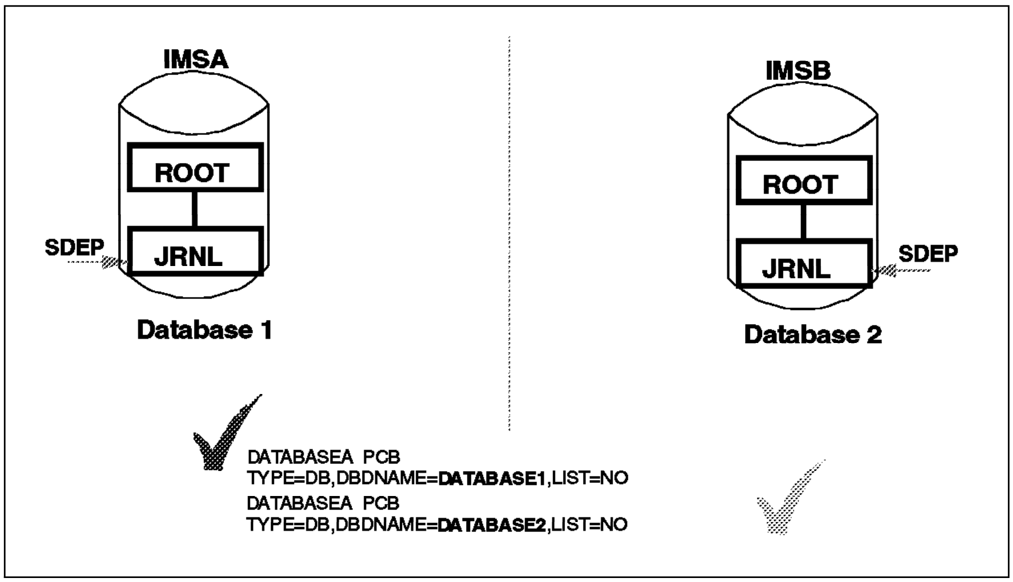


Figure 25. Third DEDB (SDEP) Data Sharing Solution

4.2.4 Conversion of MSDBs to HDAM Structures

STEPS 28 and 29 (see pages 93 through 94)

ABSA had only a few MSDBs that were relatively small. These databases where accessed frequently, so it was very important for them to keep the performance benefits of this database type. Therefore ABSA decided to duplicate the MSDBs on both IMS systems. It was not possible to duplicate one of the MSDBs, however, so ABSA converted it to a HDAM database, with its own dedicated VSAM buffer pool. When IMS initializes, a BMP processes the data sequentially, loading the data into the buffer pool to minimize I/O for subsequent online transaction access.

4.2.5 HDAM Performance

STEP 32 (See page 94)

As an example of the type of activity required to ensure that the newly modified database structures will perform well, let us review the MSDB to HDAM conversion.

MSDB MSDA15 was converted to a simple HDAM structure with a dedicated buffer pool assigned to it. When the conversion was completed and testing using the HDAM structure started, it soon became evident that there were considerable performance issues to resolve. The applications accessing this database ran very slowly when online and BMP applications processed the data concurrently.

To understand the performance issues, let us first examine the database organization:

- The database consists of 15 root segments containing control information.
- Segment with key of 2 has information that has to be reviewed per application access whether it is running in a message processing program (MPP), interactive fast path program (IFP), or BMP region.
- Specific BMPs have access to control information in segments for use in the restart process.
- Segment with key of 5 is used for restarting BMP OLP150.
- Segment with key of 6 is used for restarting BMP OLP141.
- Segment with key of 9 is used for restarting BMP OLP040.

Each BMP read the segment with key of 2 with PROCOPT=GO and then updated the restart segment associated with it. But each online transaction accessed the control segment with PROCOPT=G, thereby locking resources for a period. The ABSA project team and applications staff decided that a “dirty” read was acceptable for applications running within the online regions, so the PROCOPT was changed to GO.

The team also discovered that the BMPs had to physically update only one of the three checkpoint segments associated with the particular BMP at checkpoint time. Before, a checkpoint had been issued every 20 updates, but only the last update was really required to be committed to the database.

The team decided to make changes to both the PROCOPT values and some application code. After this, accessing the HDAM structure was not a performance bottleneck.

The tools used by the project team and the applications staff to review this performance issue were the IMS DC Monitor, IMS Monitor Summary and System Analysis utility (IMSASAP), and the IRLM lock trace.

4.2.6 Examine Existing Disaster Recovery Plans

STEP 35 (see page 95)

Regardless of the facilities and processes in place for existing disaster recovery plans, the introduction to sysplex configurations demands a re-review of disaster recovery environments. With the distribution of workload, the total available capacity of an offsite environment might not be able to manage the normal production workload, so decisions on core business emergency operational processes must be made.

ABSA began developing a sysplex-based emergency backup site that would also be used for production stress testing. For the stress test exercises, copies of production databases were made available, and, through the use of the restored image copies, relatively current access to production level data was consistently maintained.

4.2.7 Ensure Diagnostic Procedures and Facilities Are in Place

STEP 37 (see page 95)

To perform effective problem source identification in a sysplex data sharing environment, it is vital that the required documentation can be obtained and that tools that analyze problems associated with application and system code defects or system performance be understood and ready for use. In addition, to compare the performance of the current and sysplex data sharing environment, it is important to obtain monitor run output before implementation. There are many aspects to preparing your environment for effective problem source identification. We discuss some of the major tools.

- **IMS monitor reports**

  The IMS monitor (DFSUTR20) and the DB Monitor (DFSUTR30) reports lock wait times in the NOT-IWAIT field in the call summary report. This field should be compared for similar application-monitored runs taken in the current and target sysplex data-sharing environment. If increases are seen in the lock wait times, a locking situation is probably the cause. The field value is in microseconds, so a value of 222517 would be 0.22 seconds.

- **Deadlock analysis report**

  The deadlock report available from the IMS File Select and Print utility (DFSERA10) should be included in the toolkit used for the preparation of block level data sharing (BLDS) in a sysplex environment.

  Deadlocks always produce type X′67FF′ records on the victim′s log after an ABENDU0777 deadlock. IMS utility DFSERA10 with exit DFSERA30 specifying DEADLOCK is used to produce the report with the SYSUT1 DD statement pointing to the log from the system with the victim in the deadlock. The control statements input to DFSERA10 are documented in the *IMS/ESA Version 5 Utilities Reference: System*, SC26-8035.

  The following sample control statements provide a report showing the participating transactions and BMPs in any deadlocks, so that users can determine which applications are causing deadlocks and take corrective action before system performance is seriously compromised.

  ```
  OPTION PRINT O=5,FLDLEN=2,FLDTYP=X,VALUE=67FF,COND=M
  OPTION PRINT O=33,FLDLEN=8,FLDTYP=C,VALUE=DEADLOCK=E,EXITR=DFSERA30
  END
  ```

The report includes specific data identifying both the holder of the lock and the requester (who was denied access to the lock), details of the locking levels of each request, and, where relevant, identifying information such as the keys for database locks.

Figure 26 shows an example of a portion of a deadlock report obtained from ABSA.

```
DEADLOCK ANALYSIS REPORT - LOCK MANAGER IS IRLM
 ......................................................................
RESOURCE DMB-NAME LOCK-LEN LOCK-NAME     - WAITER FOR THIS RESOURCE IS VICTIM
01 OF 02 DSDF20A     08    00000350832A01C6
KEY FOR RESOURCE IS NOT AVAILABLE
       IMS-NAME TRAN/JOB PSB-NAME PCB--DBD PST#  RGN  CALL LOCK  LOCKFUNC STATE
WAITER IMSP     DIST     DSX000   DSDF20   00099 MPP  GET  GFPLL 904004F0  08
HOLDER IMSP     EPSS     EPX000   -------- 00128 MPP  ---- ----- --------  08
 ......................................................................
RESOURCE DMB-NAME LOCK-LEN LOCK-NAME
02 OF 02 DSDF20A     08    00000470832A01C6
KEY FOR RESOURCE IS NOT AVAILABLE
       IMS-NAME TRAN/JOB PSB-NAME PCB--DBD PST#  RGN  CALL LOCK  LOCKFUNC STATE
WAITER IMSP     EPSS     EPX000   DSDF20   00128 MPP  GET  GFPLL 904004F0  08
HOLDER IMSP     DIST     DSX000   -------- 00099 MPP  ---- ----- --------  08
DEADLOCK ANALYSIS REPORT - END OF REPORT

DEADLOCK ANALYSIS REPORT - LOCK MANAGER IS IRLM
 ......................................................................
RESOURCE DMB-NAME LOCK-LEN LOCK-NAME
01 OF 02 DSDF20A     08    00000350832A01C6
KEY FOR RESOURCE IS NOT AVAILABLE
       IMS-NAME TRAN/JOB PSB-NAME PCB--DBD PST#  RGN  CALL LOCK  LOCKFUNC STATE
WAITER IMSP     DIST     DSX000   DSDF20   00123 MPP  GET  GFPLL 904004F0  08
HOLDER IMSP     EPSS     EPX000   -------- 00128 MPP  ---- ----- --------  08
 ......................................................................
RESOURCE DMB-NAME LOCK-LEN LOCK-NAME     - WAITER FOR THIS RESOURCE IS VICTIM
02 OF 02 DSDF20A     08    00000470832A01C6
KEY FOR RESOURCE IS NOT AVAILABLE
       IMS-NAME TRAN/JOB PSB-NAME PCB--DBD PST#  RGN  CALL LOCK  LOCKFUNC STATE
WAITER IMSP     EPSS     EPX000   DSDF20   00128 MPP  GET  GFPLL 904004F0  08
HOLDER IMSP     DIST     DSX000   -------- 00123 MPP  ---- ----- --------  08

DEADLOCK ANALYSIS REPORT - END OF REPORT
```

Figure 26. Deadlock Analysis Report

The report shows that two transactions experience deadlocks. An IRLM lock for separate resources is obtained by each transaction, and each then waits on the resource held by the other. Transaction DIST is chosen to be abended. An analysis of the frequency and type of deadlocks in the current environment is necessary to determine whether application or database redesign is required.

- **IMS trace data**

  Familiarity with tracing tools and techniques is required to quickly and accurately collect trace data to analyze and interpret details of IMS system operation while setting up and testing sysplex functions.

  The MTO or an authorized remote terminal operator can use the /TRACE command to invoke several kinds of trace activity to selectively record details of IMS operation. Because trace or monitoring actions affect the performance of the online IMS system, they are not usually part of normal operations. However, we recommend selective use of IMS tracing to in-memory tables during all IMS operations. The benefits, in terms of quick and definite availability of data during problem diagnosis, far outweigh the small overhead involved. Because the operation of the tracing functions of IMS and the trace interval can be controlled by the /TRACE command, the operator has to be aware of the trace requirements and the timing over which tracing is to be active.

The default target of trace output are in-memory wraparound tables. Therefore, to request IMS tracing using the external trace data sets, operators would use this command:

/TRACE SET ON TABLE xxx OPTION LOG

where xxx are the table options.

When OPTION LOG is used, the trace tables are written to one of the following external data sets:

1. A DASD data set allocated by JCL, if that is specified
2. A dynamically allocated DASD data set, if that is specified
3. A dynamically allocated tape data set
4. If nothing else is specified, trace records are sent by default to the OLDS.

If it is necessary to accumulate traces over a period of time and it would be undesirable to lose any information when the traces are overwritten in the in-memory wraparound tables, we recommend that a dynamically allocated DASD data set be used for trace data so that there is minimal impact on the IMS logging subsystem.

Details of the use of the /TRACE command can be found in Chapter 15 of *IMS/ESA Version 5 Operations Guide*, SC26-8029, in the section entitled “Using the External Trace Facility.”

The following set of trace table options for use in initial testing and early production implementation can be set with the /TRACE command or in the OPTIONS statement in IMS.PROCLIB member DFSVSMxx or data set DFSVSMAP in batch. The trace tables should be activated for output to external data sets first and then in-memory as the system stabilizes. After that they can be left on producing in-memory trace table entries.

SCHED Trace all scheduling and/or termination events.

DISP Trace dispatching events.

LOCK Indicates that LOCK and PI tracing are to be activated. This will enable tracing for IRLM locking for sysplex data sharing.

Note that this level of tracing (that is, within an IMS system) shows only events causing ′777′ deadlocks and does not show full details of all IRLM lock requests. IRLM component tracing may be required to show internal details of lock requests to IRLM.

DL/I Trace events associated with the DL/I call interface and action modules.

Other trace tables can be turned on, but their use will vary according to the problem situation being investigated.

Because IMS tracing is performed under each IMS system, in a sysplex environment with multiple IMS systems it may be necessary to obtain traces from the sysplex members either contained in-memory in separate dumps, external traces, or OLDSs.

- **IRLM V2.1 tracing**

  IRLM V2.1 uses the MVS component trace (CTRACE) facility to trace lock activity at the IRLM level. IRLM tracing is therefore quite separate from IMS tracing. The TRACE CT command lets you start, stop, or modify the following sublevel traces:

  DBM Trace interactions with the identified database management system.

  EXP Trace any exception condition.

  INT Trace member and group events other than normal locking activity.

  SLM Trace interactions with the MVS locking component.

  XCF Trace all interactions with the MVS XCF services.

  XIT Trace only asynchronous interactions with the MVS locking component.

  The IRLM start and stop load module, DXRRL183, has to be placed in the MVS link list to allow the MVS TRACE CT command to operate with the IRLM diagnostic trace.

  Details of the control parameters for IRLM tracing can be found in the *IMS/ESA Version 5 Operator's Reference*, SC26-8030, and an example is included in Figure 27. In the example, the trace data is written to an external writer data set identified in procedure CTWTR.

```
TRACE CT,WTRSTART=CTWTR
TRACE CT,ON,COMP=IRLM,SUB=(DBM)
.
.

(MVS asks for a reply.)
.
.
R 15,WTR=CTWTR,END
TRACE CT,OFF,COMP=IRLM,SUB=(DBM)
.
.
(Wait a while to make sure
trace buffers are externalized)
TRACE CT,WTRSTOP=CTWTR
```

Figure 27. Sample IRLM CT Trace Command Sequence

- **IMS and IRLM V2.1 dump formatting**

  ABSA introduced the necessary IMS and IRLM support into MVS dump formatting and IPCS functions into most of its systems. However, the omission of IPCS from a few systems that produced problems and dumps caused delays because of the transmission of SVC dumps and stand-alone dumps to systems with IPCS facilities. We recommend that you install and test these facilities on all IMS systems involved in the data sharing sysplex migration project.

Chapter 5. Implementation of IMS Sysplex Data Sharing

Although we are into the implementation phase of the migration, there is no hard line between many of the steps described in Chapter 5 and the items discussed in this chapter. Some of the activities could occur in parallel and others, serially.

5.1.1 Schedule of Daily Status Meetings

STEP 39 (see page 97)

Because of the tight implementation deadlines and rate of change created from preparations for the sysplex data sharing migration, it became necessary to hold daily half-hour meetings every morning at ABSA with all key members of the team to tabulate and prioritize activities and review the status of problems. Both the IBM and ABSA project managers chaired the meeting.

5.1.2 DBRC and RECON Data Set Activities

STEP 40 (see page 97)

Because IMS/ESA 5.1 is the last release to support DBRC recovery control, you should be using DBRC at the SHARECTL and its inherent system-managed database integrity even if you are currently not using data sharing.

At ABSA, the RECON data sets are allocated to minimize contention and optimize performance as per items in Appendix A, Step 57, and in 5.1.11, "Ensure Optimal RECON Performance" on page 65. Also DBRC was set to FORCER to force DBRC authorization checking for all databases.

5.1.3 Change RECONs to SHARECTL

STEP 41 (see page 97)

ABSA was already running at SHARECTL with databases registered at SHARELVL(0) or (1) before the migration project started.

The SHARECTL status of the RECON is stored in the header record of the RECON and applies to all subsystems in the data sharing group. Share control is implemented by using:

- The /RMCHANGE IMS command
- The DBRC command utility DSPURX00 for ′INIT.RECON SHARECTL′ or ′CHANGE.RECON SHARECTL′.

5.1.4 Register Databases at SHARELVL(1)

STEP 42 (see page 97)

SHARELVL is a parameter in the database or AREA record in the RECON data set. For databases to take part in data sharing, they must be registered in the RECON data set. SHARELVL(3) is required for data sharing among multiple IMS subsystems in a data sharing group.

Although SHARELVL(1) is not compatible with the block level data sharing levels, it does permit two data sets in the same database to be image copied

© Copyright IBM Corp. 1997

simultaneously and concurrent image copy (CIC) to be run. ABSA used the CIC feature but did not image copy data sets within the same database at the same time.

Note: For SHARELVL(1, 2, or 3) to be effective, the RECONs must specify SHARECTL.

5.1.5 Register Databases at SHARELVL(3)

STEP 44 and 45 (see page 97)

To change the share level indicator, use the online DBRC command /RMCHANGE:

/RMCHANGE DBRC=′DB DBD(XXXXXXXX) SHARELVL(3)′

DBRC replies with this command input:

CHANGE.DB DBD(ORDERDB) SHARELVL(3)

Before this command is entered, stop the database by issuing the /DBRECOVERY GLOBAL command; otherwise IMS rejects the CHANGE command.

ABSA quickly changed the SHARELVL value through a global update to (3) and then reversed the SHARELVL to (1) for DEDBs with SDEPs and databases that were known to have affinity status.

At this point the earlier study on the shareability of databases will have proven worthwhile if there are no compatibility issues with the following items:

- DFSMDA macro
- PROCOPT parameter in the PCB source
- ACCESS parameter in the DATABASE macro

If it is found that an ACCESS=EX had not been changed earlier for a database that is required to be shared, the ACCESS=xx can be changed through a /START DATABASE dbn ACCESS=xx, where xx could be UP for update, RD for read, and RO for read-only access. But the /START command is local and must be entered in all IMS systems participating in the data sharing group. The final solution lies with modification of the ACCESS=xx value in the DATABASE macro.

If it is decided that a database is to be excluded from data sharing, ACCESS=EX should be set and the database registered with SHARELVL(0).

Step 46 is completed in conjunction with this activity because the VSAM SHAREOPTIONS must be (3 3).

5.1.6 Implement SHAREOPTIONS(3 3) and DISP=SHR for VSAM DBD

STEP 46 (see page 97)

ABSA set SHAREOPTIONS(3 3) and DISP=SHR before the migration. Unless both SHAREOPTIONS(3 3) and DISP=SHR are set, the open of a VSAM database data set will fail if SHARELVL of 1, 2, or 3 is used.

The RECON data set is also accessed as a key sequenced data set (KSDS) and requires SHAREOPTIONS(3 3).

5.1.7 Create a Clone of the IMS System

STEP 50 (see page 98)

This step was relatively easy as the system was built and tested in isolation. At this point it was important to involve operations so that they were aware of this cloned system. Both processors were attached to the coupling facility by this time. The RECON data set must be SHAREOPTIONS(3 3) with DISP=SHR, and any data sets that will be shared must be on shareable DASD.

At this point ABSA had two cloned IMS systems but without the structures of SCOPE=GLOBAL in their IRLMs, MSC to route workload between systems, and a network interface that would include both systems. They were still operating in a non-sysplex-data-sharing mode.

5.1.8 Multiple Systems Coupling Implementation

STEP 51 through 53 (see pages 98 through 99)

Now that the environment was set up, ABSA had to decide on a way of establishing workload on the "additional" cloned IMS system with transparency to their clients. These options were considered:

- Use Intelligent front-end systems
- Activate MSC routing
- Modify a delivery system signon for FBSS (Financial Banking System Services, now called LAN Distributed Platform (LANDP))
- Use the IMS Workload Router (WLR)

The option of an intelligent front-end system was chosen for a stand-in processor as well as for the routing of transactions to the back-end IMS systems. This would be intended for automated teller machines (ATMs) and point of sale (POS) terminals only. This solution was not going to be available in ABSA′s required time frame, so a solution still had to be found for the ATMs and POS terminals in the interim, as well as for the FBSS network (which constitutes the majority of logons).

ABSA decided to use the new IMS/ESA Version 5 DFSNPRT0 exit as well as change the code on its delivery system platform to establish workload on the cloned IMS.

All ATMs would logon to the original IMS subsystem, and all external links would remain defined on this system. Cryptography and the exchange of key information are used in the communications with external institutions. The cryptography functions are not shareable across the sysplex, so ATMs have to logon only to the original IMS system, and transactions that require cryptography have to have an affinity to the originating IMS system.

For the FBSS network, the logons were to be divided between the IMS systems within the sysplex. Code on the delivery system was changed such that terminals with even numbers would signon to one system, and terminals with odd numbers would signon to the other IMS system.

The MSC Input Message Routing exit, DFSNPRT0, was used to route certain transactions across MSC links, namely, those transactions with an affinity to an IMS subsystem. ABSA does not intend to route high volume transactions that are used to and require fast response. Fast path transactions would fall into this category, but they are not MSC eligible anyway. Also FBSS transactions that originate on the additional IMS system and are destined for the external links will be routed through DFSNPRT0 to the original IMS system for processing.

Figure 28 illustrates the use of DFSNPRT0 in routing transactions between systems.

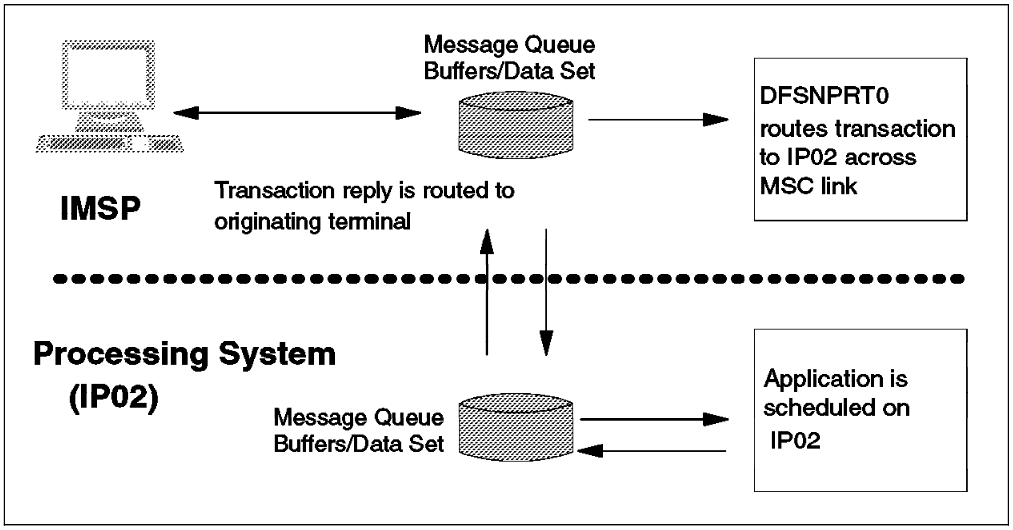


Figure 28. Use of Input Message Routing Exit (DFSNPRT0)

The IMS systems are cloned, except for REMOTE and LOCAL MSC definitions, which must be unique per sysgen. A set of logical paths is built, with a one-to-one relationship between LOGICAL PATH and LOGICAL LINK and PHYSICAL LINK.

- One MSNAME statement (with unique SYSID) per MSLINK statement (with unique PARTNER)
- Only one MSPLINK statement with the number of VTAM sessions equal to the number of MSLINK or MSNAME statements
- The logical set is subdivided through a program to facilitate workload balancing initiated from each partner.

For example:

- If the number of VTAM sessions between IMSA and IMSB = 16, the corresponding number of MSNAME or MSLINK statements = 16.
- The first eight logical paths are reserved for “round robin” workload balancing, initiated from both IMSA and IMSB.
- The last eight logical paths are reserved for round robin workload balancing, initiated from IMSB to IMSA.

Logical paths and links are monitored for availability and overhead. If a path or link is not available, transactions default to processing on the local system.

Routing decisions are based on information contained in a separately built user module, which has the following characteristics:

- The module contains tabular information only.
- The modules are dynamically refreshable. This is a facility developed by ABSA.
- Multiple exits in the same address space can use a single user module concurrently. Access to the user module is serialized during the refresh stage.

In summary, the IMS exits used to implement MSC routing across the sysplex are:

- **DFSINTX0: Initialization exit**
  - Preloads all user modules at IMS startup time
  - Assists in the setup of the lock manager environment
- **DFSNPRT0: Input Message Routing exit**
  - Decides on the routing on the basis of user module parameters
  - Monitors links for availability and overload
  - Balances link traffic on a round robin basis

At this point ABSA reviewed the functions provided by the IMS workload router (PID number 5697-074). See 1.3.2.3, "Multiple Systems Coupling at ABSA" on page 5 for a discussion of ABSA′s review of the use of the IMS Workload Router.

5.1.9 Activate

"One-Way" Data Sharing **STEP 54** (see page 99)

ABSA had to move from local to global locking even though other IMSs did not participate in the data sharing, so it activated "one-way" data sharing. The coupling facility had been installed and the VSAM cache, OSAM cache, and IRLM lock structures defined. ABSA did not use OSAM, but a small OSAM cache was defined for future use if needed. The IMS PROCLIB member DFSVSMxx was modified to reference the CFNAME, and the IRLM procedure changed. Figure 29 illustrates how ABSA modified its DFSVSMxx member in the IMS.PROCLIB data set.

```
 .
 .
POOLID=A15,FIXINDEX=YES,FIXDATA=YES,FIXBLOCK=YES
VSRBF=512,30
DBD=MSDA15(1,A15)
OPTIONS,INSERT=SEQ
OPTIONS,VSAMPLS=LOCL
OPTIONS,BGWRT=(YES,30)
OPTIONS,STRINGMX=80
OLDSDEF OLDS=(01,02,03,04,05,06),                    X
 BUFNO=255,MODE=DUAL
WADSDEF WADS=(1,2,3,4)
CFNAMES,CFIRLM=IMSPIRLM,CFVSAM=IMSPVSAM,CFOSAM=IMSPOSAM
```

Figure 29. DFSVSMxx Definitions for CFNAMES

The structures with the names referenced in the DFSVSMxx member were already defined in the coupling facility policy. When IMS starts, the names specified in the CFNAMES statement are used to connect IMS to these previously defined structures.

Figure 30 lists activities associated with the "one-way" data sharing implementation.

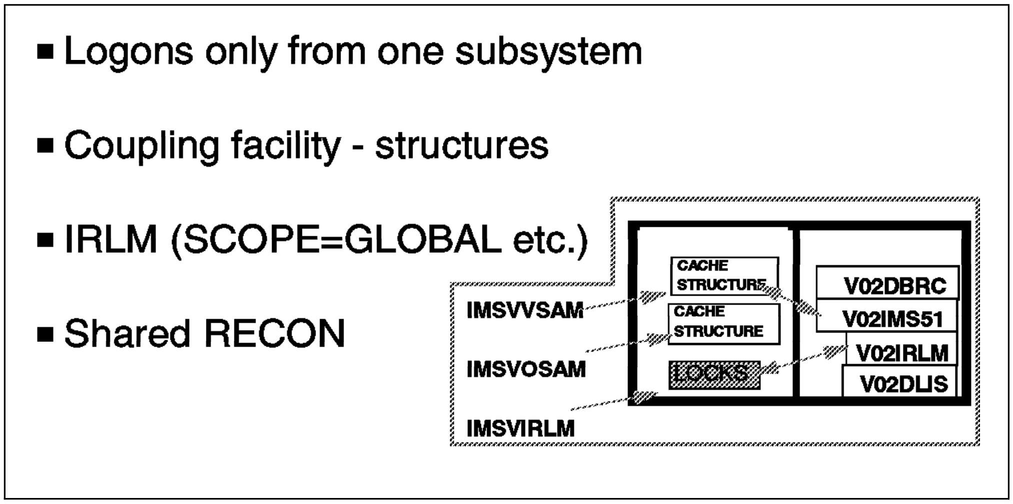


Figure 30. Activities Associated with "One-Way" Data Sharing

At ABSA, the coupling facility uses a logical partition (LPAR) on the 9672 processor in the development environment. Production uses two coupling facilities, and the structures are placed such that the load is balanced and availability maintained if recovery is required. The IRLM structure is 40 MB, and the VSAM buffer invalidate structure, 58 MB. Figure 31 completes the review of the "one-way" data sharing phase.

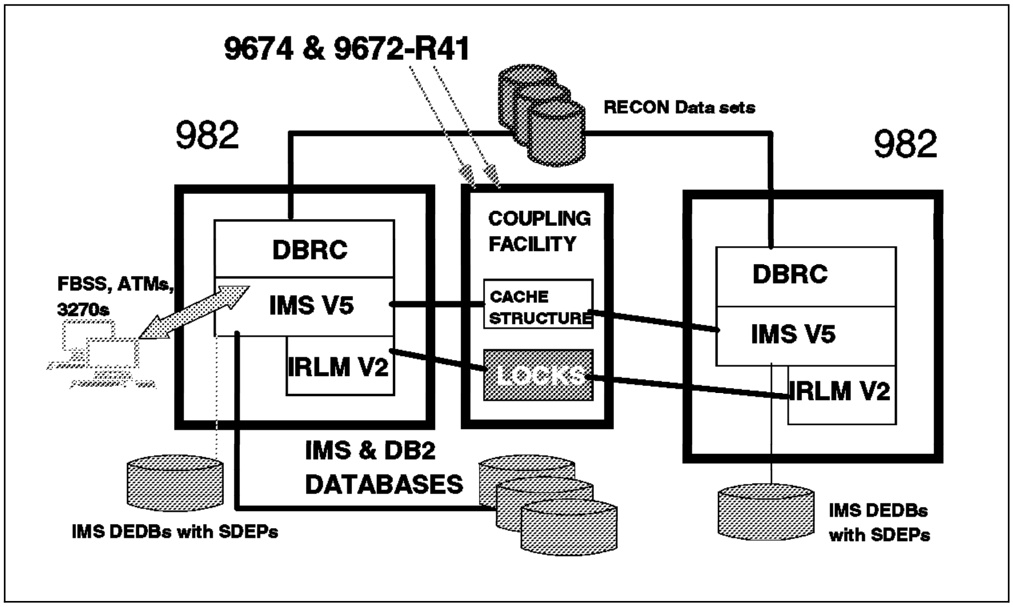


Figure 31. "One-Way" Data Sharing

At this stage, users were only allowed to logon to the "original" IMS system. The second IMS was able to share the data, even though no online transactions were processed on this IMS.

5.1.10 Test the Functionality of the Sysplex

STEP 55 (see page 99)

In Appendix A under Step 55 there are 63 suggested items related to testing the IMS system and application functionality in the data sharing test environment. There are also a few reminders associated with required procedure changes.

5.1.11 Ensure Optimal RECON Performance

STEP 57 (see page 104)

Because the RECON data set must be shared between systems, it is important to improve the efficiency of access and minimize contention as much as possible. This is not a data sharing effort alone. RECON performance is vital to all IMS installations; as a shared resource for the data sharing group, however, access to the RECON data sets must be optimized. The activities associated with RECON performance improvements are listed below. Those activities undertaken by ABSA are indicated with a #.

- The WRITECHECK parameter in the VSAM cluster definition is not specified. The default of NOWRITECHECK is used.
- The RECON local shared resource (LSR) buffers are increased to 48 Index and 192 data buffers. #
- Each RECON data set is cataloged in unique (separate) catalogs. Each catalog is on a separate device and is also on the same volume as the RECON data set it describes. This ensures that a catalog failure does not cause multiple RECONS in the set to be lost. #
- An unique alias per RECON data set is created.
- Each RECON data set is defined on a different DASD volume, behind different cache control units, each supporting DASD fast write (DFW) on different channels. #
- The RECON data sets are excluded from GRS management. #

  GRS can be used to convert hardware reserve requests to SYSTEMS ENQ requests, which are propagated to all CPUs or MVS images in the sharing sysplex. The entire volume is not locked out from access by other CPUs or MVS images when the system enqueue method is used. However, a GRS converted reserve using software is slower than the hardware reserve, so the process is slower.
- RACF is used to ensure that no unauthorized IMS subsystem can access the RECON data sets.
- Whenever possible, the /DBRECOVERs commands for full function databases are grouped together.
- The size of the RECON logical record is to be increased as the number of systems in the sysplex grows. Currently the RECON logical record size at ABSA is 32KB. It was 64KB in the development system, but it was left at 32KB on the production system. ABSA has two Change Accumulation groups, so the smaller size is sufficient. Also, BACKUP.RECON uses the facilities of IDCAMS, which would run into the restriction of backing up only 32KB of spanned records to tape.
- To keep activity to a minimum on the RECON data sets:

– The data management block (DMB) pool is kept large enough to prevent databases from closing

– A dummy program to read or update all databases to be accessed during the current IMS session is run at IMS startup time.

5.1.12 Test the Failure Scenarios

STEP 62 (see page 105)

In Appendix A under Step 62 there are eight examples of failure tests and suggestions for extending them in your environment. It is necessary to undertake this testing, as it will give your staff practical experience with system recovery in the sysplex and enable you to ensure that operations procedures have been fully tested.

5.1.13 Activate Two-Way Data Sharing

STEP 65 (see page 110)

ABSA defined the MSC links, implemented the routing exit and then made the changes to the Delivery System application, which enabled logons from FBSS terminals to both IMS systems. IRLM was changed to enable global locking. Figure 32 provides a general overview of the two-way data sharing environment.

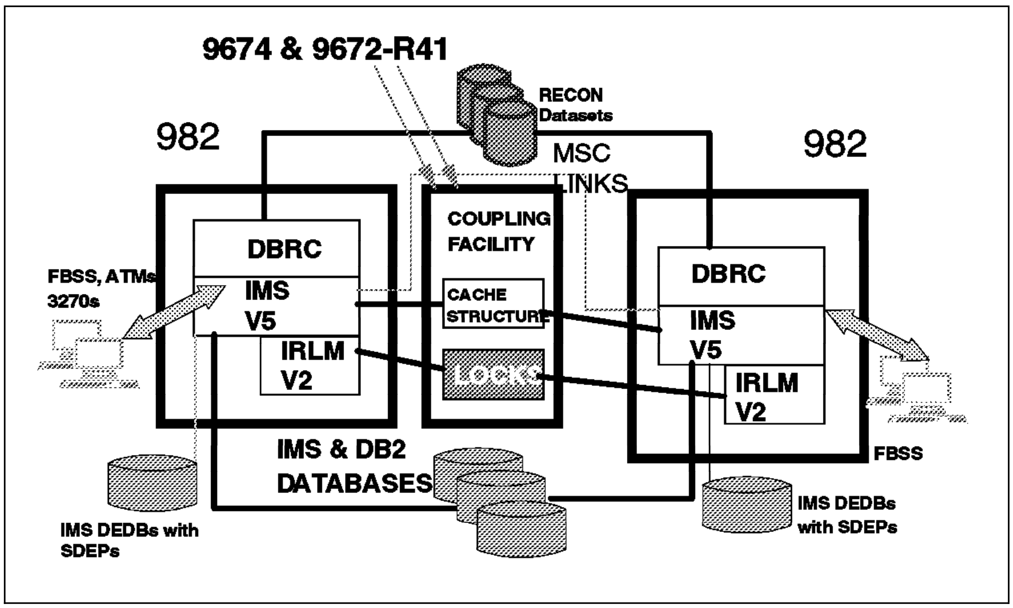


Figure 32. Two-Way Data Sharing

By October 1996, all banking sites had been linked into the data sharing complex and full production level two-way data sharing had been successfully implemented.

5.1.14 Future Activities Associated with Data Sharing

What does ABSA plan to do to maximize the benefits of IMS data sharing in a sysplex environment? Some of its planned activities are:

- Migrate to IMS/ESA Version 6 as soon as possible, to make use of the data sharing support for DEDBs with SDEPs and VSO DEDB structures. This will be the next step to implement full data sharing, without having to resort to the circumventions needed for IMS/ESA Version 5 operation (such as the SDEP sharing).

- Implement mechanisms to effect dynamic load balancing in the sysplex, including the use of VTAM generic resources and IMS capabilities of dynamically sharing message queues rather than using MSC.
- Use a front-end or stand-in processor for routing and backup processing when required.
- Use DASD remote dual copy as part of the disaster recovery process.
- Move to a four-way sysplex and use third-generation CMOS technology.
- Full conversion to the ABSA2 application environment, which will provide ABSA with a single image system for all its banking systems.

Chapter 6. IMS Sysplex Operations and Automation

The sysplex is a group of several MVS systems and their related subsystems working together to form a single logical data processing environment. This single systems image concept can be managed from one MVS console for a single point of control. Messages from more than one system are routed to one physical console, and operator commands affecting more than one system can be issued on the same MVS console.

The sysplex brings a substantial amount of change to the way in which computer systems are run and managed. It is vital that operations staff be fully aware of these changes and familiar with all of the new messages and commands.

In this chapter we discuss the effect on the computer operations department of moving to a sysplex environment. We discuss new commands, messages, address spaces, and the changes required to existing operating procedures. We also discuss the extra functions that should be automated in an IMS Parallel Sysplex.

One important lesson learned during the project was the necessity of having the operations staff involved at all times. When system programmers issue the new sysplex commands, the operations department can miss out on the practical education, and this can lead to operational problems in the production environment.

6.1 New Components

The most visible change to operations is the number of new components, all of which require management. These components include:

- IMS data sharing group
- Internal resource lock manager (IRLM)
- Coupling facility

6.1.1 IMS Data Sharing Group

The single production IMS system is replaced by multiple IMS systems running in a sysplex data sharing environment. The IMS workload is spread across more than one IMS system, and the IMS systems are linked to and have access to the same set of physical databases. The IMS operations staff must thus manage several IMS systems, coordinating their actions across the Parallel Sysplex.

6.1.2 Internal Resource Lock Manager

IRLM is a new and critical part in the sysplex data sharing environment. Each subsystem participating in the data sharing group has its own IRLM running as a separate MVS address space. The IRLMs manage data sharing with the use of global locks held in the coupling facility.

© Copyright IBM Corp. 1997

6.1.3 Coupling Facility

The coupling facility is the key technology that makes high-performance data sharing possible. It is a combination of hardware and software services.

Within the coupling facility, storage is dynamically partitioned into structures that MVS manages. Each structure has a unique function:

- Cache structure

  MVS/ESA Version 5 uses the cache structure to keep track of the buffers held in each IMS across the sysplex. If a buffer is updated in one IMS, IRLM uses the cache structure for buffer invalidation. The cache structure can also be used as a high-speed buffer for storing shared data with common read and write access.

- List structure

  The list structure enables authorized applications to share data that is organized in a set of lists so that they can implement functions such as shared work queues and maintain status information.

- Lock structure

  The lock structure supplies shared and exclusive locking capability for serialization of shared resources.

The operations staff must be familiar with both the MVS commands that are used to manage the coupling facility, as well as the messages the system generates regarding the structures in the coupling facility.

It is necessary to specify the size of these structures to IMS. For information about how to size these structures, see the paper entitled "Calculating IMS and DB2 Locking Rates for Parallel Sysplex," which is available from your local IBM Software Specialist (it is the PTSLOCK package on the IBM marketing tools disk).

6.2 MVS Commands

Three new MVS commands are used in the data sharing sysplex environment:

- *ROUTE*
- *DISPLAY XCF*
- *SETXCF*

6.2.1 Preparation for MVS Sysplex Commands

Before you can use the new MVS commands across the sysplex from a single system, you must complete two tasks:

1. Define the CONSOLxx member in SYS1.PARMLIB

   Two of the parameters in this member are important in terms of the messages received on the MVS console in a sysplex environment:

   - MFORM=S

     This parameter causes MVS to prefix each message with the system name, enabling an operator to associate the message with a particular MVS image in the sysplex.

   - MSCOPE=*ALL

This parameter allows messages from all MVS images in the sysplex to be received at the MVS console.

2. Set up the MVS Command Prefix Facility (CPF)

   The MVS CPF allows any operator or authorized application to enter a command and route that command to the appropriate system in the sysplex for execution. A subsystem such as DB2, JES2, or IMS Database Control (DBCTL) can create unique command prefixes for each copy of the subsystem in the sysplex and control which systems can accept the subsystem commands for processing. The response to the command is sent back to the originating console. To display the command prefixes in use, issue this MVS command:

   ```
   DISPLAY OPDATA,PREFIX
   ```

6.2.2 ROUTE

The MVS route command tells MVS to execute the operator command on:

- All systems in the sysplex
- A subset of the systems
- One particular system

The command responses are aggregated and sorted by sysnames, if received within a timeout interval (default 30 secs). Only one command at a time can be routed. The ROUTE command is an example of the use of XCF signaling, a function of MVS for communicating among the various systems in a sysplex. Some examples of the ROUTE command are:

- Display all active jobs (*D,A,L* command) on system *MVS2*:

  RO MVS2,D A,L

- Display the time (*D T* command) on each of the systems in the sysplex:

  RO *ALL,D T

- Start IRLM (*S IRLM* command) on the *OTHER* system:

  RO *OTHER,S IRLM

- Start A&SYSNAME (*S A&SYSNAME* command) on all systems in the sysplex.

  RO *,S A&SYSNAME

  Note: &SYSNAME will be replaced by the name of the system on which the command is executed.

  The use of &SYSNAME is a new function of MVS Version 5.1. In the above example, the route command is interpreted according to the name of the MVS images and the number of MVS images started in the sysplex. Refer to member IEASYMxx in SYS1.PARMLIB to see the association between the &SYSNAME value and the IEASYSxx parameters.

Refer to *OS/390 MVS System Commands*, GC28-1781, for more details on the ROUTE command, including restrictions and the timeout interval calculation.

6.2.3 DISPLAY XCF

You can display the following resources, using the DISPLAY XCF command:

- The XCF environment, which runs as an address space under MVS
- XCF communications links

  The communication paths between the XCF address spaces can be either channel-to-channel or through the coupling facility. We recommend that both facilities be used.
- Coupling facility

  The coupling facility can be seen as an extended XCF signaling path and as common storage shared among all the MVS systems in the sysplex.
- Couple data sets

  The couple data sets contain the common rules that apply sysplexwide. An example is the CFRM couple data set, which contains the CFRM policies that define how the various coupling facility structures are to be defined.

Some examples of the DISPLAY XCF commands are:

- Display the names of the systems connected to the coupling facility and the names of the structures currently allocated in the coupling facility:

  D XCF,CF
- Display the CFRM policy in use:

  D XCF,COUPLE,TYPE=CFRM.
- Display the XCF structure information (see Figure 33).

```
D XCF,STRUCTURE

IXC359I  12.43.03  DISPLAY XCF 389
STRNAME           ALLOCATION TIME  STATUS
COUPLE_CKPT1      --       --     NOT ALLOCATED
IMSPIRLM        11/14/95 15:11:08 ALLOCATED
IRRXCF00_B001   11/20/95 09:09:56 ALLOCATED
IRRXCF00_P001   11/20/95 09:09:55 ALLOCATED
IXCPLEX_PATH1   11/14/95 14:19:34 ALLOCATED
IXCPLEX_PATH2   11/14/95 14:19:17 ALLOCATED
IMSPOSAM        11/20/95 08:46:04 ALLOCATED
IMSPVSAM        11/20/95 08:46:05 ALLOCATED
```

Figure 33. Sample MVS DISPLAY XCF STRUCTURE Command

- Display information about one or all of the structures used in the current CFRM policy, such as size, connection name, number of systems connected (see Figure 34 on page 73)

```
DISPLAY XCF,STRUCTURE,STRNAME=IMSPIRLM

    STRNAME: IMSPIRLM
     STATUS: ALLOCATED
      POLICY SIZE    : 40 000 K
      REBUILD PERCENT: N/A
      PREFERENCE LIST: CFC1     CFC2
     EXCLUSION LIST IS EMPTY

    ACTIVE STRUCTURE
    ----------------
     ALLOCATION TIME: 06/17/96 07:50:26
     CFNAME         : CFC1
     COUPLING FACILITY: 009672.IBM.02.000000040252
                        PARTITION: 2   CPCID: 00
     ACTUAL SIZE    : N/A

     STORAGE INCREMENT SIZE: 256 K
     VERSION        : AD07D5D8 DF0AE200
     DISPOSITION    : KEEP
     ACCESS TIME    : 0
     MAX CONNECTIONS: 7
     # CONNECTIONS  : 2


    & AMPERSAND DENOTES CONNECTOR WHO LOST CONNECTIVITY TO STRUCTURE
    CONNECTION NAME  ID VERSION  SYSNAME  JOBNAME  ASID STATE
    ---------------- -- -------- -------- -------- ---- ----------------
     IRLMV$$$$LP01001 01 000100C3     ASYS     P01IRLM   00EE   ACTIVE
     IRLMV$$$$LP02002 02 000200BD    AP02       P02IRLM  025E   ACTIVE
```

Figure 34. Sample MVS DISPLAY XCF STRUCTURE, STRNAME Command

- Display the XCF policy information (see Figure 35).

```
DISPLAY XCF,POLICY

 IXC364I  12.47.01  DISPLAY XCF 405
 TYPE: CFRM
      POLNAME:       TESTPOL1
      STARTED:       11/14/95 13:52:32
      LAST UPDATED:  11/14/95 13:44:28
 TYPE: SFM
 POLICY NOT STARTED
```

Figure 35. MVS DISPLAY XCF POLICY Command

6.2.4 SETXCF

The MVS SETXCF command can be used to manipulate certain data sharing sysplex resources. Examples of the SETXCF command are:

- Activate a CFRM policy to take new structure definitions into account:

  SETXCF START,POLICY,TYPE=CFRM,POLNAME=policy

- Rebuild CF structures in the same or a different coupling facility:

SETXCF START/STOP,REBUILD,STRNAME=xxxx

- Free or clear a structure from the coupling facility:

  SETXCF FORCE,STRNAME=xxxx

  Use this command with caution. It could cause database integrity problems if there are retained locks in the structure.

6.3 IMS Commands

The most apparent change in a data sharing sysplex environment, for operations staff, is that the production IMS system now consists of more than a single image of IMS. Thus more than one IMS control region and their related address spaces must be managed. It is mandatory for each IMS image to have a unique IMSID. The IMS systems can be a clone or copy of each other, and the same resources are defined to each system (as at ABSA), or they can share only a limited number of resources among them.

The IMS MTO is not shared among systems, and one must exist for each IMS system. There is no integrated console, as there is with the MVS system.

It is up to operations to coordinate commands across the sysplex. If, for example, a change to a PSB requires an online change, operations must execute a /MODIFY PREPARE MODBLKS and a /MODIFY COMMIT on all IMS systems that take part in the data sharing sysplex. Operations must also verify these commands have completed successfully on all systems in the sysplex, as there is no coordinated command processor. Operations staff must do the coordination manually. The best approach is to automate your IMS operations across the sysplex as much as possible. This subject is discussed further in 6.5, "IMS Sysplex Automation" on page 77.

To display the status of the sysplex, a /DISPLAY command must be issued on each individual IMS system.

To extend the transaction workload beyond a single IMS system, transactions can be routed between IMS systems by using MSC. MSC links the IMS systems together and passes transactions that originate from a terminal connected to one IMS system to a second IMS system for processing, as shown in Figure 36 on page 75. The links must be defined to the various IMS systems and managed by the operators.

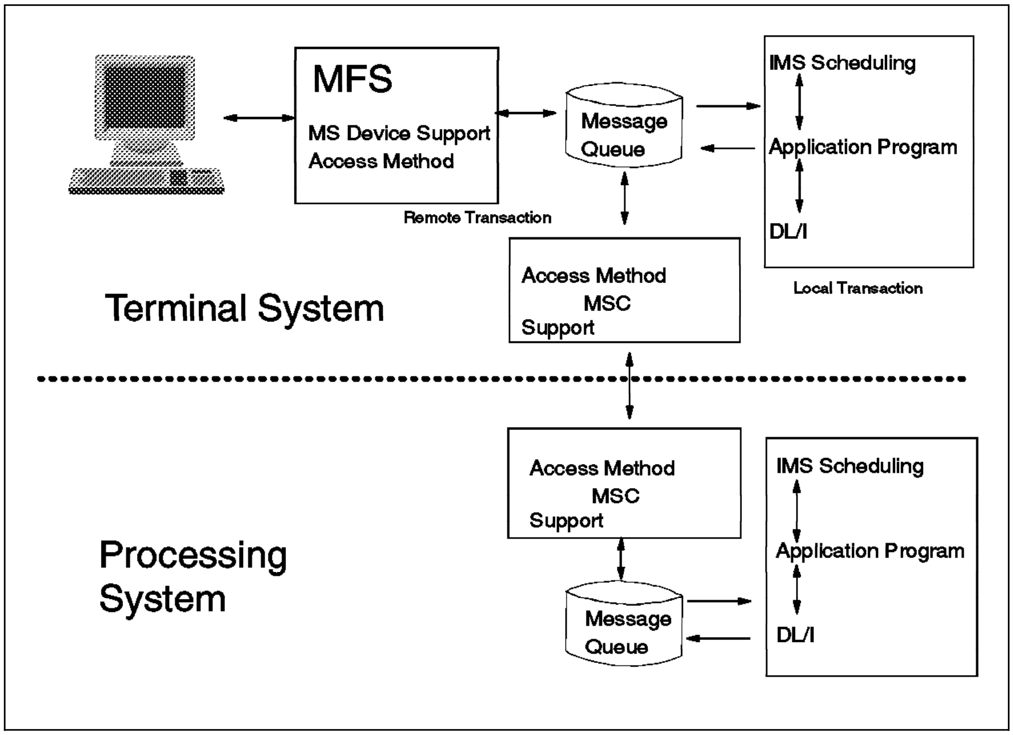


Figure 36. MSC Transaction Flow across Systems

Operations staff can manage the links with a number of commands:

- Display the MSC links and their respective queue counts, as shown in Figure 37.

```
/DISPLAY MSNAME ALL

    MSNAME    ENQCT   DEQCT    QCT
    MSC1          0       0      0
    MSC2         12      12      0
    MSC3          0       0      0
    *95325/130404*
```

Figure 37. DISPLAY MSNAME ALL Command

- Display the physical link and associated logical links, as shown in Figure 38.

```
/DISPLAY ASMT LINK ALL

   LINK PLINK    SIDR SIDL MSNAME
      1 LINK        2    1  MSC1
                    4    3  MSC2
                    6    5  MSC3
    *95325/130521*
```

Figure 38. DISPLAY MSC ASSIGNMENT LINK Command

- Display the status and queue counts for the logical MSC link, as shown in Figure 39 on page 76.

```
/DISPLAY LINK ALL
    LINK PARTNER  RECD  ENQCT  DEQCT  QCT  SENT
       1 LP         12     12     12    0    12 IDLE ACTV
    *95325/130618*
```

Figure 39. DISPLAY MSC LINK Command

- Display the queue counts for the logical MSC link path, as shown in Figure 40.

```
/DISPLAY MSNAME MSC1
    MSNAME    ENQCT   DEQCT    QCT
    MSC1          0       0       0
    *95325/130714*
```

Figure 40. DISPLAY MSNAME Command

6.4 IRLM Commands

The IRLM address space must be active before IMS is started and must remain active until IMS has terminated. At ABSA, the startup of IRLM is initiated by MVS as part of the MVS initialization. If for some reason IRLM is not started before IMS is started, IMS message DFS039a is issued to the MVS master console. IRLM must then be started by operations by using the S IRLMproc command.

ABSA has two IRLM address spaces per MVS image in the data sharing sysplex, one for IMS, and one for DB2.

All IRLM messages are prefixed with the characters DXR, for example, DXR101i.

The various IRLM address spaces are connected to a lock structure inside the coupling facility. The IRLMs use this structure to manage global locks across the data sharing sysplex. ABSA uses two coupling facilities to allow for a certain level of backup. Should the coupling facility that contains the IRLM lock structure fail, the structure will be automatically rebuilt in the second coupling facility.

The following commands are used when operating IRLM:

- Start the IRLM address space:

  S IRLMproc

- Tell IRLM to shut down normally:

  P IRLMproc

  If an IMS system is still connected to this IRLM, an error message is issued.

- Display information about the subsystems:

  F IRLMproc,STATUS

  Currently, IMS automation at ABSA issues this command every 30 minutes to provide a picture of the typical number of locks held, waiting, and retained. A runaway BMP, for example, quickly becomes apparent.

- Display information about subsystems in the data sharing group.

  F IRLMproc,STATUS,ALLD

  Figure 41 shows an example of this command after IRLM IP02 has failed.

```
F V02IRLM,STATUS,ALLD

DXR102I LP01 STATUS IRLMID=001 963
SUBSYSTEMS IDENTIFIED              PT01
 NAME     STATUS    RET_LKS   IRLMID  IRLM_NAME
 IMSP         UP          0      001      LP01
 IP02      SFAIL         49      002      LP02
```

Figure 41. IRLM STATUS Command Issued at ABSA

  IRLM IP02 is in a "failed" state and is holding 49 retained locks. Transactions on the surviving IMS that want to access the held information will be abended with the following messages:

```
DFS0781A ABEND 3303 IN DFSECP20+S203+SP51+510+06/06/96 IP01
DFS554A ABJL001X 00014 REGION   SYP914  (2)  000,3303  96/163 17:16:41 IP01
```

- Display information about the IRLMs in the data sharing group:

  F IRLMproc,STATUS,ALLI

- Abnormally stop IRLM, whether or not any IMSs are connected to this IRLM:

  F IRLMproc,ABEND

The MVS ROUTE command can be used to send a command to a different MVS system. The return message from the following command summarizes all display information about the subsystems identified to IRLMs on the different MVS images:

- ROUTE *ALL,F IRLM,STATUS

For more information about IRLM operator commands, please refer to the *IMS/ESA Version 5 Operator's Reference Guide*, SC26-8030.

6.5 IMS Sysplex Automation

The complex nature of the sysplex, particularly the synchronization of the various IMS systems taking part in the data sharing complex, makes many of the operational tasks ideal candidates for automation.

We recommend that automation be used to assist the operation of the IMS subsystems. Should one of the MVS images that runs the automation system fail, automation must be reactivated in another MVS image to ensure continuity.

ABSA implemented IMS automation with NetView and assigned the IMS MTO to a NetView terminal access facility (TAF) session. Both production IMS systems have their MTO assigned to the same TAF terminal, so all system messages for both systems come to the same point.

In addition to the existing IMS automation, ABSA automated the following tasks:

- Synchronized dumping of required address spaces

In the event of a wait state occurring in any of the IMS systems in the sysplex, it is necessary to take dumps of all address spaces involved. In a stand-alone environment, it is easy to identify which address space is causing the problem. In a sysplex data sharing environment, it is necessary to take dumps of the IMS control region, DLI separate address space, IRLM, and possibly DB2, on *all* systems simultaneously. Dump commands should be automated because they are long and complex and could be mistyped if entered manually. If a wrong dump command is entered, it may not be possible to capture all of the necessary diagnostic information. The dump data sets should be dynamically allocated to avoid partial dumps being taken because the dump data sets are full. See also 3.4, "Dump Management" on page 27 for a discussion of dump management.

- Synchronization of the online change process

  The online change process is used to dynamically change the ACBLIB, FMTLIB, and MODBLKS data sets, as well as the RACF profiles. In an environment where more than one IMS system is sharing resources, these commands must be coordinated across all systems to ensure that every IMS system accesses the same copy of the data set. This is particularly important if the IMS systems are cloned from a single base. The indicator as to which copy of the particular data set is in use is kept in the IMSx.MODSTAT data set. It is vital that the MODSTAT data set reflect the same information across the sysplex.

- Procedures to manage the allocation of databases across the sysplex

  When a database is to be reorganized, it must not be open by any of the IMS systems in the sysplex. The /DBRECOVERY command is used for each IMS, so that IMS stops and deallocates the database data set and thus prohibit any access to the database from the online system. It is imperative that the /DBRECOVERY command be executed on all IMS systems in the data sharing sysplex. To ensure that the databases have in fact been deallocated by each IMS system in the data sharing sysplex, the RECON data set must be queried.

  If a /START command is to be issued against a database or area, a procedure must be in place to ensure that the command is propagated across the entire data sharing sysplex.

- Monitoring and restarting MSC links

  It is advisable to monitor the MSC links that connect the various IMS systems. Monitoring can be done by automatically issuing /DISPLAY commands, and then acting on a link failure, for example, by restarting the link if it is down.

- Reconnection of IMS to IRLM after failure

  If the IRLM fails, automation can be used to restart the IRLM and connect the IMS to IRLM. Automating restart and reconnect reduces the duration of any outages caused by the IRLM failure.

- Consolidation of abends for the data sharing group

  In a data sharing sysplex, abends from more than one IMS system have to be kept. Automation processes must be in place to keep the records from abends across the various systems in one data set.

Chapter 7. Performance Considerations

The performance of the IMS environment in Parallel Sysplex depends primarily on the configuration of the sysplex hardware. But taking this a step further, we need to examine the components of the sysplex environment that are different from the pre-sysplex environment and understand how these affect the performance of the system.

IMS Version 5 and the Parallel Sysplex introduces n-way data sharing, which clearly adds an extra level of function to the IMS system. As with all software, when additional function is added, and especially if it is a sophisticated new feature, there is a natural increase in the amount of resources consumed. In the case of n-way data sharing, the main resource that is consumed is CPU processing time.

7.1 Sysplex Performance at ABSA

ABSA monitors the performance of its application system very closely and has found the online response times were not affected by the move to sysplex data sharing. This of course depends on the resources used by the application, as well as the way in which they are used. With the mix of database types used at ABSA (mostly fastpath), locking is the same as previously. However, with full function, the locking scope changes, and this may have an impact on application performance.

ABSA prefers to keep the processor utilization to a maximum of 70%. In the event of a single CPU failure, the rest of the system will still be able to process the load. ABSA has therefore moved from a single ES/9021-982 for its production system, to two of these systems sharing the load across the sysplex. ABSA is planning to add an IBM 9672-RX4 as the third system in the sysplex.

The workload processing characteristics can change when the speed of the processors change. This will happen when your system moves from running on a bipolar system to one based on CMOS technology, and again as you move from one generation of CMOS to another. Similarly, the change in numbers of processors in the MVS will potentially have an impact. Also, when using PRISM to partition large bipolar machines, you must remember that some MVS images were only running on a smaller number of engines than were in the hardware. Therefore, you can see increased parallelism on your CMOS system. ABSA has never been memory constrained, but memory size may also have an impact in other organizations.

CPU cost increased significantly for full function, although application performance was much the same.

The speed of the processor used for the coupling facility will have an impact on overall system performance. When ABSA moved its coupling facility from CMOS generation 1 to generation 3, the coupling facility utilization dropped from 20% busy with lots of conversion to 11% busy with no conversions (from synchronous processing to asynchronous processing).

ABSA experienced better performance of its DB2 applications when the size of the DB2 lock structures was increased.

© Copyright IBM Corp. 1997

Initially ABSA used a single JES initiator class for its batch workload and ran all this batch workload on a single MVS system to reduce the contention to databases across multiple BMPs. Later, when affinities were better understood, ABSA used two JES initiator classes across two MVS systems in the sysplex.

It is easy to move workload across systems in the Parallel Sysplex, in order to keep the CPU down under 100% busy.

PROCOPTs were reviewed for correctness in a Parallel Sysplex environment when performance of the application was noticably affected.

Database buffer space was effectively increased with the move to data sharing, as there were fewer programs accessing the buffers for each database in each IMS in the sysplex.

RECON placement is important in the Parallel Sysplex. RECONs need to be placed in GRS, to reduce contention to these shared data sets.

Appendix A. ABSA IMS Parallel Sysplex Project Plan

The ABSA IMS data sharing project plan is present in this appendix. There are some definite assumptions that have to be understood before review it:

- We are moving from one IMS Control Region environment to a data sharing environment by effectively cloning the production system to another IMS system.
- All the activities that are listed here have been executed against development system resources. For example starting IRLM proc V01IRLM associates that IRLM to a development system.
- In order to create the production Sysplex environment most of the steps listed here would have to be repeated.
- There are a few steps like teleprocessing network simulator (TPNS) testing that would not be attempted on the final production environment.

© Copyright IBM Corp. 1997

Table 3 (Page 1 of 3). IMS SYSPLEX DATA SHARING PROJECT PLAN - Planning

Step	Description	Reference
1	**Understand the Current Environment**	**2.1, “Understand the Current Environment” on page 10**
	Document the core business functions performed by the computing services group in your organization	
	Map out the current hardware configuration and system software levels in use	
	List the external links and types of links that connect IMS to external subsystems	
	List the type of workload that is currently present in the major applications	
	Prepare a current view of the enterprise computing environment with a view to processors, major application systems, and databases.	
	Identify the types of users which require the most resources	
	Map out the service level commitments for systems in your organization that will be migrated	
	Document the on-line and batch schedules for systems in your organization that will be migrated	
	Identify any application backouts because of database lockouts	
	Identify the performance and tuning tools and processes available presently	
	Document the usermods currently on your systems	
	Create a list of the vendor products associated with the current environment	
	List the Databases and applications that currently present affinity status (eg, Fast Path DEDBs with SDEPs before IMS/ESA V6.1, optical disk applications that might store tables in common storage).	
	Document the user exits and usermods that have hard coded affinities to particular systems	
	List any IMS log processing programs that review only one system′s logs per run	
	Determine if you have the necessary skillset in house to perform the migration of the IMS, IRLM and MVS components into a BLDS Sysplex environment	
	Review your change and problem control processes and document them	
2	**Architect an overall component image of the Sysplex environment**	**2.2, “Architect an Overall Component Image of the Sysplex Environment” on page 16**
	List the business reasons associated with migrating to a Sysplex environment	
	List the technical reasons associated with migrating to a Sysplex environment	
	Map the selection of the hardware elements in the Sysplex that the IMS environment will use	
	Define the software package elements and levels in the Sysplex that the IMS environment will use	

Table 3 (Page 2 of 3). IMS SYSPLEX DATA SHARING PROJECT PLAN - Planning		
Step	**Description**	**Reference**
	Map how the network will be connected to the Sysplex for use with the IMS environment	
	Identify and define the final MVS/LPAR sysplex configuration. • System and LPAR names • MVS and SYSNAME etc. • JES2 MAS/NJE configuration • MVS mastercat/usercat configuration • The GRS/MIM reserves and global reserves expected in the target environment	
	Define the migration strategy for the use of processors and LPARs from current to final configurations including interim phases	
	Identify how the work will be directed to processors and LPARs for Batch, TSO, IMS and CICS	
	Define the migration strategy for IMS from the current environment to the final including interim configurations for systems programming test, application development, integration and production	
	Determine the dependencies that an IMS migration to extended Parallel Sysplex mode might be subject to: DB2 installation, fully functional base environments including MVS/JES2, TSO/ISPF, program products.	
	Identify any complicating factors that might exist at the end of this project for the following workloads: IMS, TSO, CICS, BATCH. Examples could be the current lack of process for the following situations in environments where cloned applications are running in multiple systems or LPARS. • Scheduled shutdowns of a processor or LPAR for hardware or software maintenance • Unscheduled outages of MVS or its subsystems • Subsystem maintenance rippled across systems. List the scheduling and restart factors for applications when the above events occur	
	List the IMS resources that will be able to be cloned (transactions, data sets etc..)	
	Identify the IMS resources that will have affinities to one specific IMS system within the Sysplex	
	Identify remote system connections to IMS in the new Sysplex configuration	
	Develop a preliminary view of the target environment showing processor, LPAR, coupling facility, major applications, target databases etc.	
	Identify what the security processes will be for the IMS environment in the Sysplex	
3	**Plan for running the Sysplex in degraded mode**	**2.2.1, “Plan for Running the Sysplex in Degraded Mode” on page 19**

Table 3 (Page 3 of 3). IMS SYSPLEX DATA SHARING PROJECT PLAN - Planning

Step	Description	Reference
	Define what degraded mode means based on the above Sysplex architecture design	
	Identify what core business functions must continue to operate when the Sysplex is in degraded mode	
	Map how the network would interface with a Sysplex running in degraded mode	
4	**Develop the Sysplex project plan**	**2.2.2, “Develop the Sysplex Project Plan” on page 19**
	Obtain necessary executive and financial approvals	
	Ensure that there is an effective process to communicate project status to executive management	
	Assign a project coordinator	
	Identify general Customer and Vendor staff and their responsibilities	
	Select the project management tools to use	
	Define the education required	
	Estimate changes associated with hardware/software/personnel resource requirements for the planning, preparation, implementation and ongoing phases of the project	
	Ensure that the necessary resources (weekend migration and testing time and system access) are available and booked in advance	
	Develop a risk assessment plan and review with all involved functions and management levels as required	
	Develop this project plan assigning ownership and start and expected completion dates to items	

Table 4 (Page 1 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
5	**Develop naming conventions.** The naming conventions will need to include subsystems, data sets, IMS region job names, CF structures, started task names for IMS regions, IMS data sharing group names, MVS system names, (the selection of names will occur latter in the process)	**4.1.1, "Develop Naming Conventions" on page 37**
	Items to consider • Some name changes will mean application JCL changes • Some names will be tied to the MVS system name • The staging of name changes in all environments	
6	**Document how databases and applications will be accessed and used in the Sysplex environment**	**4.1.3, "Migrate Lock Manager from Program Isolation to IRLM 2.1" on page 40**
	Partitioned or cloned databases	
	Partitioned or cloned applications	
	Which resources such as applications, databases, data sets, terminal network elements etc. can′t be shared	
	Which applications or databases will have to modified to be able to join the sharing environment	
7	**Naming Standards** Change naming standards on IMSV to Sysplex naming standards so that shared/non-shared resources can be identified. (The names used in this project plan are obviously unique to the ABSA environment)	**4.1.2, "Naming Standards" on page 38**
	1. Define alias for high level qualifier - IV01.*	
	2. Define user catalogue for IV01.**	
	3. Define RACF for IV01	
	4. Create SYSM.IV01.RESLIB library	
	5. Copy Generated Reslib into SYSM.IV01.RESLIB	
8	**Data Set Sharing** Determine and document what IMS system data sets will be shared or unique. (This list maps out the choices that ABSA made within the scope of what data sets are under user control for shared or unique status)	**4.1.2.1, "Shared IMS Data Sets" on page 38, 4.1.2.2, "Nonshared IMS Data Sets" on page 38**

Table 4 (Page 2 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	Naming Standards: IMSV** • Shared Data sets IV01.** • Non-Shared Data sets	
	Shared Data sets - Prefixed by IMSV.** • DBDLIB • FORMAT/A/B • PGMLOAD • PGMLOAD • LOADERS • PROCLIB • RECON1/2/3 • REFERAL • TFORMAT • USOURCE • USOURCEN • MATRIX/A/B (Prefixed by SYSM. and then IMSV.) • MODBLKS/A/B (Prefixed by SYSM. and then IMSV.) • URESLIB - Prefixed by SYSM. then IMSV. • UPARMLIB	

Step	Description	Reference
	Non-Shared Data sets - Prefixed by IV01.*, IV02.* etc. • POLDS01/2/3/4/5 • SOLDS01/2/3/4/5 • WADS1/2 • IMSMON • DFSTRA01/2 • QBLKS • SHMSG • LGMSG • MODSTAT • MSDBCP1/2 • MSDBDUMP • BACKUP. • MSDBINIT (0) • RDS • TCFSLIB • SYS01/2/3/4/5/6/7/8/9/10/11/12 • RESLIB This will be prefixed by SYSM. and then IV01. or IV02. etc. • JOBS • JCLLIB • ACBLIB/A/B • PSBLIB	
9	**Implement the Data Set Name Changes**	**4.1.2.2, “Nonshared IMS Data Sets” on page 38**
	1. Rename IMS system data sets	
	2. Adjust JCL for system DBA and application jobs	
	3. Modify naming conventions when necessary for auto operation execs	

Table 4 (Page 3 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Table 4 (Page 4 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	4. Modify overrides in DFSPBxxx members	
	5. Change required data set names for audit control and accounting	
	6. Change data sets in the following procs in IMSV.PROCLIB • REGBMPNP • REGFPP • REGFFP • REGDLIP • REGBMPP • INTRDP	
	7. Change all Procs using Reslib library (Run search on IMSV.PROCLIB)	
	8. Change members in library IV01.JCLLIB • ARCHJCL • CAJCL	
	9. Change TCO library and member IV01.TCFSLIB(DFSTCF)	
	10. Ensure that all shared data sets are on shared DASD	
	11. Build GDG base for files • IV01.MTOLOG.MONTH.* • IV01.BACKUP.MSDBINIT.* • DCMON	
10	**Prepare for Region name changes**	**4.1.2.2, "Nonshared IMS Data Sets" on page 38**
	1. Change the following members in IV01.JOBS • VREG • VFF100A • VFF101A • ..... • VFF110A	

Table 4 (Page 5 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	2. Delete old dynamic allocations from SYSM.IMSV.URESLIB • DFSWADS1/2/3 • DFSOLP01 - 05 • DFSOLS01 - 05 • DFSTRA01/02	
	3. Run dynamic allocation to SYSM.IV01.RESLIB	
	4. Automation Changes: Change region names - now 8 characters	
	5. Operational Changes: • Communicate with Operations on new region names • Update Scheduler with new naming standards	
	6. IMS Department Procedure Changes • Gen Procedures – Copy new Reslib – Set-up Dynamic Allocation with every Gen – Communicate to all departments about changes	
	7. DBA Department Changes: Add &IMSID parm into procedures	
11	**Started Task Name Changes**	
	1. Change the following names • IMSV51 to V01IMS51 • DBRCV to V01DBRC • DLISASV to V01DLIS	**4.1.2.3, "Started Tasks" on page 39**
	2. Add the following to the Started Task Table • V01DBRC • V01IMS51 • V01DLIS	

Table 4 (Page 6 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	3. Set-up procedures in Proclib for • V01DBRC • V01IMS51 • V01DLIS	
12	**Implement procedure changes for V01DBRC, V01IMS51, V01DLIS**	**4.1.2, “Naming Standards” on page 38**
	1. Change in IMSV.PROCLIB(DFSPBxxx): • DBRCNM= DBRCV to V01DBRC • DLINM = DLISASV to V01DLIS	
	2. Start V01IMS51.IMSV	
	3. Communicate changes to operations group	
	4. Communicate changes to DBA/IMS departments	
13	**Review and update JCL where required**	**4.1.2, “Naming Standards” on page 38**
	For general support JCL such as DBRC skeletal JCL, statistics JOB JCL, Application JCL.	
14	**Get IRLM installed and used instead of program isolation (PI) as the lock manager for local locking** This could be considered an implementation segment line item but ABSA decided to use to implement IRLM as the lock manager with SCOPE=LOCAL before changing naming standards which is considered to be within the preparation segment of the project plan.	**4.1.4, “Design the Coupling Facility IMS Environment” on page 43**
15	**Prepare for the install of IRLM 2.1 via SMP/E on BSYS**	**4.1.4, “Design the Coupling Facility IMS Environment” on page 43**
	1. Check with IBM for PSP Buckets, Hipers and PE (PTF in error) maintenance or perform the research at the account using tools such as Servicelink.	
	2. Order current level of IRLM	
	3. Receive/Apply/Accept FMIDs	
	4. Run JCLIN as per installation (SMP Library)	
	5. Build system libraries for IRLM 2.1 (ADRXRLOAD etc.)	
	6. Copy these libraries to Reslib	
16	**Migrate IRLM 2.1 to IMSV**	**4.1.4, “Design the Coupling Facility IMS Environment” on page 43**

Table 4 (Page 7 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	1. Set-up IRLM name in SYS1.PARMLIB(IEFSSN01) - LV01	
	2. Add IRLM to Program Properties Table (PPT)	
	3. Add CMMND02 to start IRLM with IPL	
	4. Specify IRLM=Y in IMSV.PROCLIB (DFSPBxxx)	
	5. Specify IRLMNM=LP01 in IMSV.PROCLIB (DFSPBxxx)	
	6. Workload Manager - dispatching priority (15,15)	
	7. Create startup JCL with parms (SCOPE=LOCAL, DEADLOK=2,1, PC=YES)	
	8. Start V01IRLM	
	9. Start V01IMS.IMSV	
	10. Remove values for PIINCR and PIMAX from DFSPBxxx (for Program Isolation support)	
	11. Add JCLIN for IRLM to system generation process	
17	**Formulate the manner that security will be implemented within the Sysplex environment**	
18	**Review all the user modifications and exits that will be running in the Sysplex environment for compatibility**	
19	**Design a process in which remote systems will be connected and interface into the Sysplex**	
20	**Produce the necessary plan to connect the IMS network into the Sysplex environment**	
21	**Design the environment for the coupling facility**	**4.1.5, "Examination and Modification of Support Procedures" on page 47**
	1. Determine IRLM Lock Structure size	
	2. Determine the size of cache structure needed for VSAM buffers (ABSA didn′t define OSAM structures)	
	3. Determine where the IMS and IRLM structures will be placed	
	4. Set up procedures to change CF structure sizes and placements	
	5. Develop a set of procedures for the recovery of CF structures	
	6. In member DFSVSMxx (in DFSVSAMP for BATCH BLDS jobs) determine the names to use for the CFNAMES statements CFIRLM, CFVSAM and CFOSAM (ABSA didn′t use OSAM structures)	
22	**Examine current operational procedures and update as necessary**	
	1. MVS and IMS start-up and shutdown steps	
	2. The process to recover should systems and subsystems fail	

Table 4 (Page 8 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	3. How DBs are reorganized or recovered currently	
	4. The manner that Jobs will be scheduled in the Sysplex environment - examine use of unique initiator classes for each system	
23	**Examine and modify installation support procedures such as**	**4.1.5, “Examination and Modification of Support Procedures” on page 47**
	1. IMS systems generation in Sysplex environments	
	2. Maintenance of multiple sharing IMS systems in a Sysplex	
	3. Security maintenance for the Sysplex resources	
	4. DB definition utilities: DBD, PSB, ACB	
	5. On-line Change utilities	
	6. Management of the logs from multiple data sharing systems	
	7. Database reorganization and recovery in Sysplex environments (review naming standards for CA and Image copy data set)	
	8. Application changes and maintenance in a cloned and non-cloned environment	
24	**Develop plan to enable Data Sharing**	**4.2, “Database and Application Considerations” on page 48**
	1. Schedule time for a workshop with all interested parties present	
	2. Review IMS system definition parms associated with data sharing	
	3. Understand changes to IRLM, IRLM association to XCF, DBRC, Proc and JCL changes.	
	4. Investigate the most effective test bed for application and systems software in the Sysplex environment	
	5. Examine scope of work to update operating procedures	
	6. Understand the requirements for automation changes	
25	**Database Conversion Planning** This is an optional step but one that ABSA was required to take in order to allow applications that had been accessing DEDBs with SDEPs or MSDBs to now join into the data sharing environment.	**4.2, “Database and Application Considerations” on page 48**
	1. Determine DBs to be converted	
	2. Set up DB conversion strategy plan	
	3. Draw up naming standards	

Table 4 (Page 9 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	4. List procedures to be changed	
26	**Database conversion activity - to enable sharing of DEDBs with SDEPs**	**4.2.2, "Data Sharing Positioning Stage" on page 50**
	1. Write Data Capture Exits in Assembler	
	2. Test Data Capture Exits	
	3. Obtain performance statistics for DC Exits	
	4. Obtain space requirements for new 'SDEP' DEDBs	
	5. Request use of additional packs for new 'SDEP' DEDBs	
	6. Update DB macros for new 'SDEP' DBs	
	7. Update DB source for 'DDEP' converted DBs	
	8. Create DB source for new 'SDEP' DBs	
	9. DBD/ACBGEN for updated 'DDEP' DBs	
	10. DBDGEN for new 'SDEP' DBs	
	11. Create new SDEP databases - Check control roots	
	12. Set up DB unload /reload procedures	
	13. Set up FPSCAN jobs for existing SDEP DBs	
	14. Set up FPINIT jobs for new SDEP DBs	
	15. Set up image copy jobs for new SDEP DBs	
	16. Set up Analyzer jobs for new SDEP DBs	
27	**PSB Activities associated with the above DB conversion**	**4.2.2, "Data Sharing Positioning Stage" on page 50**
	1. Determine PSBs to be changed	
	2. Automate procedure to update affected PSBs	
	3. Update PSB source with 'SDEP' DBs	
	4. PSB/ACBGEN for updated PSBs	
28	**Conversion of MSDB to HDAM structures**	**4.2.3, "DEDBs with SDEPs: Conversion for Data Sharing" on page 51**
	1. Review DB bufferpool usage	

Table 4 (Page 10 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation

Step	Description	Reference
	2. Update DBD source for MSDBs to HDAM	
	3. Set-up conversion JCL for MSDB databases	
	4. Define dedicated buffer pools for new HDAM DBs	
	5. DBD/ACBGEN for converted MSDBs	
	6. Convert MSDBs to HDAM databases	
	7. Set up jobs which will load HDAM data into buffer pools at startup or when necessary.	
	8. Review procedures to add new terminals	
29	**IMS Start-up JCL Changes**	**4.2.3, "DEDBs with SDEPs: Conversion for Data Sharing" on page 51**
	Jobs to open individual SDEP DBs on their ′own′ IMS (TCO)	
30	**End-of-Day JCL Changes** (This step is unique to the ABSA environment and may not be required at other Customer locations)	
	1. List all ′Marker Drop′ jobs	
	2. Update ′Marker Drop′ applications	
	3. Set up duplicate ′Marker Drop′ steps	
	4. Test ′Marker Drop′ jobs	
31	**RECON activities associated with the above conversions of DB structures**	
	1. Define converted MSDBs	
	2. Define sets of new SDEP DBs	
32	**Review of potential performance issues in the Sysplex environment associated with above conversion of DB structures**	**4.2.4, "Conversion of MSDBs to HDAM Structures" on page 53**
	1. Examine new HDAM databases for potential "hot spots"	
	2. Review potential bottlenecks in DEDBs with SDEPs	
	3. Do general application reviews for possible contentions in the Sysplex application mix	
33.	**Determine how to implement processing in degraded modes** This step was not part of the ABSA project plan specifically but included for completeness.	
	1. Finalize the choice of the core business functions that must available regardless of the loss of processing capacity	

Table 4 (Page 11 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation		
Step	**Description**	**Reference**
	2. Develop procedures associated with the lost of various Sysplex components to restore functionally for the above defined core business functions	
34	**For the stress testing phase, examine the TPNS requirements**	
35	**Examine existing disaster recovery plans and re-work them to fit into the Sysplex environment**	**4.2.5, "HDAM Performance" on page 54**
36	**Examine automated operation programs and procedures and modify as much as possible during the Preparation Segment of the project based on the systems, operational and application changes that have occurred during this period. (Some examples are listed below)**	
	1. If DATA SHARING STOPPED message received (connection loss to CACHE Structure in CF) use `D XCF,STRUCTURE,STRNAME=` command to identify what failed.	
	2. Watch for IRLM messages on MVS console, prefixed by DXR: `DXR027A irlm SESSION LOST, SHARING STATE IS IN DOUBT ACTION REQUIRED` Automate an alert	
	3. Check in automation for message: `DXR136I irlmnm HAS DISCONNECTED FROM THE DATA SHARING GROUP` and issue an alert.	
37	**Ensure that all necessary diagnostic procedures and facilities are in place**	**4.2.6, "Examine Existing Disaster Recovery Plans" on page 55**
	1. Turn on the dispatcher, scheduler, DL/I and Lock traces on in core via the OPTIONs statement in the DFSVSMxx member (or the DFSVSAMP data set if running batch)	
	2. Modify the SDATA parms associated with SDUMPs so that necessary storage is obtained	
	3. Examine the dump data set allocations and the process to manage them	
	4. Add DXRRLI83 to link list (for IRLM use of the CTRACE facility)	
	5. Test the MVS Component Trace facility which is used for IRLM 2.1 diagnostic tracing	
	6. To run IPCS to review IRLM dumps provide a JOBLIB or STEPLIB DD statement pointing to the library with the IRLM PRDUMP formatting routine DXRRLM50	
	7. Ensure that module DXRRLM50 and DFSOFMD0 are both included in the print dump print control table in Sys1.Parmlib member BLSCECT.	
38	**Education and Documentation**	

Table 4 (Page 12 of 12). IMS SYSPLEX DATA SHARING PROJECT PLAN - Preparation		
Step	**Description**	**Reference**
	Provide documentation of pending changes to Applications, Operations and End-Users. Present all parties involved with the details of the projected changes and how they will be affected by them.	

Step	Description	Reference
39	**Schedule daily half hour morning meetings with all affected staff at this point within the project** This might not be necessary but pressing schedules required that staff at ABSA communicate on and prioritize activities and problems and this was the most effective method to further that process.	**5.1.1, "Schedule of Daily Status Meetings" on page 59**
40	**RECON data set activities (see STEP 57 for information on ensuring optimal RECON performance)**	**5.1.2, "DBRC and RECON Data Set Activities" on page 59**
	1. Determine placement of RECON data sets	
	2. Create separate user catalogue for each RECON	
41	**Change RECONs to SHARECTL**	**5.1.3, "Change RECONs to SHARECTL" on page 59**
42	**Register Databases at SHARELVL(1)**	**5.1.4, "Register Databases at SHARELVL(1)" on page 59**
43	**Modify the IMS sysgen definition with parameter changes necessary for data sharing** ABSA plans to specify DBRC=FORCE in the IMSCTL macro but that will be latter into the data sharing implementation cycle.	
44	**Adjust ACCESS= parameter on the DB macro if necessary to UP for shared databases.**	**5.1.5, "Register Databases at SHARELVL(3)" on page 60**
45	**Register Databases at Sharelevel(3) in DBRC**	**5.1.5, "Register Databases at SHARELVL(3)" on page 60**
	1. Update SHARELVL to (3) globally	
	2. Reverse SHARELVL to (1) for ′SDEP′ DBs and other Databases with affinity status	
	3. Update RECON to ′FORCER′	
	4. Remove default ′SSID′	
46	**VSAM Shareoptions must be defined as (3,3) for shared databases.**	**5.1.6, "Implement SHAREOPTIONS(3 3) and DISP=SHR for VSAM DBD" on page 60**
47	**Ensure that backup and recovery utilities specify DBRC=Y**	
48	**Make the necessary changes to the network to supply the IMS data sharing partners with transaction input**	
49	**GRS setup**	
	1. Add all IMS data set names to SYSTEM Exclusion RNL	
	2. Do not include the DBRC RECON or the OLDS or WADS names in the RESERVE conversion RNL	

Table 5 (Page 1 of 7). IMS SYSPLEX DATA SHARING PROJECT PLAN - Implementation

Table 5 (Page 2 of 7). IMS SYSPLEX DATA SHARING PROJECT PLAN - Implementation

Step	Description	Reference
50	**Create a clone of the IMS system**	**5.1.7, “Create a Clone of the IMS System” on page 61**
	1. Develop the series of unique IMS system data sets for the clone environment	
	2. Sysgen the clone environment	
	3. Create procs and execution time parms for clone system	
	4. Develop the necessary support procs and DBRC skeletal JCL for cloned or unique data sets and functions	
51	**Multiple Systems Coupling Implementation between IV01 and IV02**	**5.1.8, “Multiple Systems Coupling Implementation” on page 61**
	1. Identify list of transactions which use facilities from other vendors which might place them in affinity status (transactions that use DB2 V3.1 or use EMH buffering)	
	2. Define MSC Links for IV01 (MSLINK, MSNAME, MSPLINK macros)	
	3. Define MSC links for IV02 (MSLINK, MSNAME, MSPLINK macros)	
	4. Set-up Gen Procedures	
	5. Establish VTAM links between systems	
	6. Customize Input Message Routing Exit DFSNPRT0 (evaluate use of IBM Workload Router product also at this point)	
	7. Include DFSNPRT0 in authorized library in JOBLIB, STEPLIB or LINKLIST library concatenated in front of IMS.RESLIB	
52	**Establish Operating Procedures for MSC**	**5.1.8, “Multiple Systems Coupling Implementation” on page 61**
	1. Automate /RSTART LINK command (TCO)	
	2. Link termination /PSTOP LINK	
	3. Test trace command which can be used if link problems suspected: `SET ON LINK X LEVEL 3 MODULE DDM`	
	4. Test commands to be used in MSC environment: pg.275 IMS/ESA Version 5 Administration Guide, SG????????????? **Note:** GGGGGGGGGGGGGGGGGGGGGGG	
	5. Test message recovery when IMS goes down and is then restarted.	

Table 5 (Page 3 of 7). IMS SYSPLEX DATA SHARING PROJECT PLAN - Implementation

Step	Description	Reference
	6. Test events when a transaction is stopped - establish automation procedures to ensure excessive queuing does not develop etc.	
	7. Test application program abends - sending of error messages to other system	
	8. Protect the use of the `/DEQUEUE MSNAME PURGE` command - be aware of its consequences	
53	**Establish Monitoring Procedures for Multiple System Environment**	**5.1.8, "Multiple Systems Coupling Implementation" on page 61**
	1. Coordinate DCMON monitoring period between the two systems	
	2. Get familiar with IMS Monitor Report Print Program • MSC Traffic Report • MSC Summaries • MSC Queueing Summary Report	
	3. Extract Multiple System transaction Statistics by running Log Merge Utility and then Log Transaction Analysis Utility. Get familiar with the information contained in this report.	
	4. SLR monitoring, set up procedures to merge logs.	
54	**IRLM Changes to Activate Global Locking instead of Local**	**5.1.9, "Activate " on page 63**
	1. Change IRLM procedures to SCOPE=GLOBAL, Maxusers, etc.	
	2. Set the XCF transport class for the IRLM group at a message length of 395 bytes	
	3. Set the PATHIN and PATHOUT statements in the MVS COUPLExx member	
55	**Test Sysplex system to ensure functionality of all components**	**5.1.10, "Test the Functionality of the Sysplex" on page 65**
	1. /STA DC	
	2. Test all IMS commands	
	3. Run Log Transaction Analysis Utility (DFSILTA0)	
	4. Run Fast Path Log Analysis (DBFULTA0)	
	5. Run Statistical Analysis Utility DFSISTS0	
	6. Test Off-line Dump Formatter	
	7. ERE testing	

Table 5 (Page 4 of 7). IMS SYSPLEX DATA SHARING PROJECT PLAN - Implementation		
Step	**Description**	**Reference**
	8. Test Usermods	
	9. Test MFS	
	10. Test Interactive Dump Formatter	
	11. Test RACF signon	
	12. Test Loadlist	
	13. Test Saswitch, Paynet, Multinet, Videotex, CICS links (these are the external links)	
	14. Test IMS Monitor	
	15. OLDS recovery	
	16. SLDS recovery	
	17. Test archives	
	18. MTO Spool	
	19. APPC Testing	
	20. Test DB2	
	21. Test Change Accumulation	
	22. Test Randomizers	
	23. Test database recovery: • DEDB • MSDB • HIDAM • HDAM • HISAM • GSAM	
	24. Exercise Database Updates	
	25. Test all DBRC commands	
	26. Test Reconclr	
	27. Test DFSDDLT0 to all DB types	
	28. DCMON Extensions (Fast Path Monitor)	

Table 5 (Page 5 of 7). IMS SYSPLEX DATA SHARING PROJECT PLAN - Implementation		
Step	**Description**	**Reference**
	29. On-line image copy	
	30. Concurrent image copy	
	31. DEDB Reorganization	
	32. Program DBA025J (DBRs DB or Area)	
	33. High Speed DEDB Reorg (DBFUHDR0)	
	34. DBTOOLs DEDB Unload/Reload	
	35. Test PSBGEN	
	36. Test DBD gen - all types	
	37. Test ACB gen	
	38. Test on-line change	
	39. MSDB Maintenance	
	40. Dynamic Allocation	
	41. Test DEDB initialization	
	42. Test DEDB SDEP Scan	
	43. Test Nodemon	
	44. Test Automation	
	45. Crypto (Hardware and Software, performance)	
	46. Concurrent copy (Use Global commands to /DBR databases ensure coordinated properly)	
	47. Hardware compression	
	48. HSSP	
	49. Test all global commands	
	50. Test dynamic change of database access intent with `/START` command	
	51. Test changing of sharelevel of database via `/RMCHANGE DBRC=` command	

Table 5 (Page 6 of 7). IMS SYSPLEX DATA SHARING PROJECT PLAN - Implementation

Step	Description	Reference
	52. Test Other IBM products • DB2 • SLR • IMS Monitor Summary and System Analysis utility (IMSASAP) • IMS Performance Analysis and Reporting utility (IMSPARS) • IMS Database Tools	
	53. Check transaction macro for MAXRGN= and SERIAL=YES specifications to ensure same effect in data sharing environment if necessary.	
	54. Check Application macro for SCHDTYP=SERIAL as above.	
	55. Co-ordinate the /MODIFY commands for controlling on-line change • Stop affected database in all online systems using the DB before initiating a sequence of /MODIFY commands • If ACBLIB, FORMAT or MODBLKS are shared, make sure all systems use the same active library. • Ensure DFSUOCU0 updates only inactive libraries	
	56. Establish procedures to protect databases from other system access during DB Reorgs. (/DBR GLOBAL, or DBRC CHANGE.DB NOAUTH, or MTO procedure to stop allocations that would be active against the DB)	
	57. Lockmax option - implement to reduce chance of “run away” programs	
	58. Establish procedures to protect databases from update access from other subsystems whilst an Online Database Image Copy is in progress (DFSUICP0).	
	59. Establish procedures for Change Accumulation, using merge function of DFSUCUM0.	
	60. Establish procedures for database forward recovery. System logs for both systems need to be merged first using the CA utility DFSUCUM0. Then the database Recovery utility can be run.	
	61. INIT STATUSGROUPB to cater for “BA” “BB” status codes	
	62. PROCOPTs in PSBs (update when only read needed)	
	63. Test OEM Products	
56	**Establish Monitoring Procedures for Sysplex Environment**	

	1. Automate use of following MVS commands so info is displayed on log at regular intervals:
	`DISPLAY XCF,STRUCTURE`
	`DISPLAY XCF,STRUCTURE,STRNAME=`	

Table 5 (Page 7 of 7). IMS SYSPLEX DATA SHARING PROJECT PLAN - Implementation

Step	Description	Reference
	2. Use `MODIFY irlmproc,STATUS, irlmx` to show status of IRLM in DS group.	
	3. Test IRLM tracing using MVS TRACE CT command - Ensure that DXRRLI83 is in MVS link list	
	4. Use RMF III CF Activity report to analyze true/false contention situation for locks.	

Table 6 (Page 1 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance

Step	Description	Reference
57	**Ensure Optimal Recon Performance**	**5.1.11, "Ensure Optimal RECON Performance" on page 65**
	1. Do not specify WRITECHECK in VSAM Recon definitions use the default of NOWRITECHECK	
	2. Increase the number of RECON LSR buffers (defaults are 6 and 12), try 48 and 192 index and data buffers respectively. (DSPBUFFS CSECT in IMS Customization Guide.) **Note:** Correct the reference	
	3. Place each RECON data set on a separate volume behind a separate cache control unit and channel. Use the DASD Fast Write feature.	
	4. Place each RECON in a separate user catalogue with the BCS for the catalog on the same pack as the RECON.	
	5. Add DSPURI01 to the SYSTEMS Exclusion RNL (if all of above have been done). Do not include the DBRC RECON in the RESERVE conversion RNL	
	6. Execute a "database opening" program on each system before the end users start using IMS.	
	7. Group the full function databases. One /DBR command with 10 databases will process faster within DBRC than 10 /DBR commands with one database.	
	8. Increase the RECON logical record size as the number of systems in the SYSPLEX grows.	
58	**DMB Pool size review and modification if necessary**	
	Check DMB pool usage /DIS POOL DMBP (only applicable for non-resident databases.)	
59	**VSAM BUFFER POOL Hit Ratios**	
	1. Calculate the hit ratios for the current system (from DC Monitor Reports).	
	2. Calculate hit ratios for new systems in the SYSPLEX - adjust buffers if necessary.	
	3. If buffers are adjusted ensure that the CF VSAM cache structures are adjusted accordingly.	
60	**IRLM Lock Contention**	
	Use `F IRLM,STATUS` command to display IRLM waiters - monitor and try to reduce WAITING value if applicable	
61	**General Monitoring**	

Table 6 (Page 2 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance

Step	Description	Reference
	1. Ensure that sufficient monitors of non-data sharing environment are collected with a transaction mix similar to that used in data sharing environment exists in order to do valid throughput comparison.	
	2. Monitoring of I/O subsystem for data sets with potential contention.	
62	**Testing of Failure Scenarios**	**5.1.12, "Test the Failure Scenarios" on page 66**
	1. Test on-line transaction and BMP abends	
	2. Test IMS failure (e.g. IV01) • Note potential U3303 abends on IV02 • Do /ERE to ensure backout and lock releases • Try using different IRLM (any one in DS group to do /ERE)	
	3. Test Cold Start after IMS subsystem abend • Ensure Batch backouts done for each inflight PSB	

	4. Test IRLM IDENTIFY failure
	• If structure names in CF at the time of the IDENTIFY do not match those in the CFNAMES control card, the IDENTIFY will abend and IMS will issue a message. Correct the error and retry.
	• Use
	`DISPLAY XCF,STRUCTURE`
	and
	`F IRLMx, STATUS`
	commands to determine actions.	

	5. Test IRLM failure • Note that inflight UOWs will be abended and backed out • PSBs will not be scheduled • Keep IMS up (to preserve network) • Note effect on other IMS - U3303 abends for retained locks • Operator command to reconnect to the IRLM must be used once IRLM is started (test this)	

Table 6 (Page 3 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance

Step	Description	Reference
	6. Test MVS failure • Note the effect on the other IMS - U3303 abends for retained locks • /ERE the failed IMS • DB Buffers for failed IMS should be invalidated on restart • Test restart of IRLM, and failed IMS on different MVS image.	

Table 6 (Page 4 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance

Step	Description	Reference
	7. Test CF Failure (two CFs required to test automatic rebuild) • Cache Structure Effects – Rebuild attempted automatically – All buffers invalidated on ALL systems – Rebuilt structure is empty – Data sharing should be resumed – If rebuild fails - SHARELVL 1,2,3 DBs stopped - DB calls fail (U3303) - Failed trans go on suspend queues - IMS may USTOP trans – When rebuild completes - IMS reconnects to structure - IMS starts databases - IMS releases trans from suspend queue • IRLM Lock Structure – IRLM attempts rebuild on alternate (successful only if all IRLMS survive) – Repopulates structure – If rebuild fails (manual recovery required) - IMS quiesced - IMS unauthorises all shared DBs - New PSBs not scheduled – CF must be fixed or new policy invoked • Following successful rebuild – IMS reconnects to IRLM – IRLM reconnects to structure – IMS taken out of quiesced state	

Table 6 (Page 5 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance

Step	Description	Reference
	8. Test CF Link to IRLM Structure Failure (treated like IRLM failure) • IMS quiesced (IMS that failed) • Inflight UOWs abended • Dynamic backout invoked • IMS unauthorises and stops all shared DBs – Only non-shared DBs can be accessed on this system • New PSBs not scheduled • Surviving IRLMs read retained locks (U3303 abends) If CF link repaired • IMS reconnects to IRLM • IRLM reconnects to structure • IRLM “wakes” IMS • User must restart the fastpath areas	

Table 6 (Page 6 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance		
Step	**Description**	**Reference**
	9. Test Failure of CF Link to Cache Structure • Buffers (in this system) invalidated • IMS quiesces data sharing • SHARELVL= 2 or 3 DBs stopped • DB calls fail (U3303) • Failed trans on suspend queue – IMS may USTOP trans • Other IMSs continue data sharing • Data Sharing Stopped message will be received `DFS3384I Data Sharing Stopped` When link re-established • IMS reconnects to structure • IMS starts DBs – User must restart the fastpath areas • Releases transactions from suspend queue • Data would be read into buffers as CIs requested **Note:** HDAM databases which have been converted from MSDBs need to run job to read this data into buffer pool in this case to ensure performance)	
	10. Test out-of-storage conditions for IRLM and IRLM lock structure (775 and 3307 abends)	
63	**RACF**	
	Test /MODIFY PREPARE RACF	
64	**TPNS driven stress test with production level system images**	
	1. Determine Requirements	
	2. Investigate FBSS	
	3. Identify transactions	
	4. Build scripts	
	5. Build tables for variable data	
	6. Build TEST	

Table 6 (Page 7 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance

Step	Description	Reference
	7. Do simulation	
	8. Simulation	
	9. Produce report with findings	

	10. TPNS Test
	1. Take down IMSP IP02 (P01IMS51, P02IMS51)
	2. Implement Gen
	3. Rename P01IRLM to P01IRLMG
	4. Rename P01IRLML to P01IRLM (local locking)
	5. Start P01IRLM
	6. Start V01IMS51 and check if Gen is in
	7. /SWI OLDS
	8. Start TPNS test with 20 terminals
	9. Start TPNS with 50 terminals
	10. Issue a /CHECKPOINT
	11. Turn IMS Lock trace on
	`/TRACE SET ON TABLE DL/I OPTION LOG`
	`/TRACE SET ON TABLE LOCK OPTION LOG`
	12. Run an IMS Monitor
	13. /DIS POOL ALL regularly during the interval, as well as F xxxxIRLM, STATUS
	14. /TRACE SET OFF TABLE DL/I OPTION /TRACE SET OFF TABLE LOCK OPTION
	15. At end of interval /SWI OLDS
	16. Run a Deadlock report: DFSERA10 with DFSERA30 exit
	17. Format the IMS lock trace with DFSERA10 and DFSERA40 exit
	18. Analyze X'4521' and X'4522' log records for IRLM statistics
	19. Run a DBFUALT0 Fast Path Log Analysis	

65	**Activate two way data sharing**	**5.1.13, "Activate Two-Way Data Sharing" on page 66**
66	**Test performance of production system in BLDS mode and tune where necessary**	

Table 6 (Page 8 of 8). IMS SYSPLEX DATA SHARING PROJECT PLAN - Performance		
Step	**Description**	**Reference**
67	**Workload Manager and IMS service classes introduction into the Sysplex environment**	

Appendix B. Sample Data Capture Exit and DBD Source

A sample of the DBD source containing the EXIT operand used for one of the DEDBs follows:

```
DBD   NAME=RTDB54,ACCESS=DEDB,RMNAME=DBFHDC44
AREA  DD1=RTDB54A,DEVICE=3380,SIZE=4096,
      UOW=(060,6),ROOT=(5,1)
SEGM  NAME=RTSB541,PARENT=0,BYTES=(200,10)
FIELD NAME=(SEQNB54,SEQ,U),BYTES=8,START=3
SEGM  NAME=RTSB542,PARENT=RTSB541,BYTES=(200,4),
      EXIT=((DBAXDC00))
DBDGEN
FINISH
END
```

In this example, segment RTSB542 is the converted SDEP on the original DBD. The DBAXDC00 exit is called when the application inserts this converted SDEP to database RTDB54. The exit inserts a real SDEP on a copy of the database on whatever system the transaction is running. Then the exit deletes the converted SDEP that was originally inserted on database RTDB54. During end of day processing another application is used to merge these duplicated DEDBs with SDEPs.

The following is the assembler code for the exit:

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * *  DESCRIPTION : DATA CAPTURE EXIT FOR ALL DEDB'S THAT HAS    * * *
* * *              A SDEP AND USE RANDOMIZER IMM068A AND THE     * * *
* * *              ROOT SEGNAME IS STANDARD AND THE ROOT         * * *
* * *              KEYNAME IS "SEQN@##" WHERE "@##" IS THE DB    * * *
* * *              UNIQUE IDENTIFIER.        (26 DATABASES)      * * *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
         PRINT NOGEN
         EQUREG
         EJECT
*
DBAXDC00 CSECT
         STM   R14,R12,12(R13)           SAVE CALLER'S REGISTERS
         LR    R12,R15                   USE R12 AS MAIN BASE
         LA    R11,2048(R12)             USE R11 AS 2ND BASE
         LA    R11,2048(R11)             USE R11 AS 2ND BASE
         USING DBAXDC00,R12,R11          SET ADDRESSABILITY
         USING XPCB,R10                  SET ADDRESSABILITY
         USING XSDB,R9                   SET ADDRESSABILITY
         USING ENV,R8                    SET ADDRESSABILITY
         USING PCBMASK,R7                SET ADDRESSABILITY
*
         L     R10,0(R1)                 POINT TO XPCB
         L     R9,XP_S_P                 POINT TO DATA XSDB
         L     R8,XP_INQY                POINT TO "INQY" OUTPUT
*
         L     R2,XP_WRK_P               POINT TO SUPPLIED WORK AREA
         ST    R13,4(R2)                 BACKWARD POINTER
         ST    R2,8(R13)                 FORWARD POINTER
         LR    R13,R2                    R13 = OWN SAVE AREA
```

```
         USING WSAREA,R13                   SET ADDRESSABILITY
*
INIT     EQU   *
         CLC   XP_PFUNC,ISRT                PHYSICAL FUNCTION = ISRT
         BNE   ENDBAD                       NO
         MVI   XP_RSN+1,X'01'               SET XPCB REASON = 01
         LTR   R9,R9                        DATA XSDB ADDRESS GIVEN
         BZ    ENDBAD                       NO
*
         MVC   LBNUM,XP_DBD+3               MOVE THE DB UNIQUE NUMBE
         L     R2,ENV_RID                   GET REGION ID
         CVD   R2,DW                        CONVERT TO DECIMAL
         MVC   WSRID,EDITFLD                MOVE EDIT PATTERN
         ED    WSRID,DW+4                   CONVERT TO ZONED DECIMAL
         MVC   QSSAKEY,WSRID+3              MOVE REGION ID
         MVC   DQSEG,XP_DBD                 MOVE THE DATABASE NAME
         MVI   DQSEG+2,X'E2'                CHANGE INDICATOR TO SEG
         MVI   DQSEG+6,X'F1'                SAY IT'S A ROOT
         MVC   DQNUM,LBNUM                  MOVE THE DB UNIQUE NUMBER
         L     R2,XP_CK_P                   GET ADDRESS OF ROOT KEY
         MVC   DQKEY,0(R2)                  MOVE THE KEY
         MVC   DUSEGD,XP_SEG                MOVE THE DDEP SEGNAME
         EJECT
*
INSERT   EQU   *
         LA    R3,ISRT                      POINT TO ISRT FUNCTION
         L     R4,XS_SEG_P                  POINT TO SEGMENT DATA
         LA    R5,QSSAROOT                  POINT TO QSSA FOR ROOT
         LA    R6,USSASDEP                  POINT TO USSA FOR SDEP
         MVC   AIB_RNM,LABL                 MOVE PCB LABEL
         MVC   AIB_OLEN,FW0000              NO OUTPUT AREA
*
         BAS   R14,AIBCALL                  INSERT THE SDEP
         CLI   FLAG,X'00'                   SUCCESSFUL ?
         BE    DELETE                       YES - GO DELETE THE DDEP
         CLI   FLAG,X'FE'                   BAD STATUS CODE ?
         BE    CHKSTA                       YES
         MVI   XP_RSN+1,X'02'               SET XPCB REASON = 02
         B     ENDBAD
*
CHKSTA   EQU   *
         CLC   PCB_STA,GE                   STATUS CODE = "GE"
         BNE   ENDBAD                       NO - THEN END BAD
*
         MVI   WSROOT,X'00'                 BUILD LENGTH
         MVI   WSROOT+1,X'06'               SET LENGTH = 06
         MVC   WSRKEY,QSSAKEY               MOVE KEY
         LA    R4,WSROOT                    POINT TO SEGMENT DATA
         LA    R5,USSAROOT                  POINT TO USSA FOR ROOT
         MVI   XP_RSN+1,X'05'               SET XPCB REASON = 05
         BAS   R14,AIBCALL2                 INSERT THE ROOT
         CLI   FLAG,X'00'                   SUCCESSFUL ?
         BNE   ENDBAD                       NO
INSDEP   EQU   *
         L     R4,XS_SEG_P                  POINT TO SEGMENT DATA
         LA    R5,USSASDEP                  POINT TO USSA FOR SDEP
*
         MVI   XP_RSN+1,X'06'               SET XPCB REASON = 06
         BAS   R14,AIBCALL2                 INSERT THE SDEP
```

```
         CLI   FLAG,X'00'                  SUCCESSFUL ?
         BNE   ENDBAD                      NO
         EJECT
*
DELETE   EQU   *
         LA    R3,GHU                      POINT TO GHU FUNCTION
         LA    R4,IOAREA                   POINT TO I/O AREA
         LA    R5,DQSSA                    POINT TO QSSA FOR ROOT
         LA    R6,DUSSAD                   POINT TO USSA FOR DDEP
         MVC   AIB_RNM,XP_PCBNM            MOVE PCB LABEL
         MVC   AIB_OLEN,FW4096             SIZE OF OUTPUT AREA
*
         MVI   XP_RSN+1,X'03'              SET XPCB REASON = 03
         BAS   R14,AIBCALL                 GET THE DDEP
         CLI   FLAG,X'00'                  SUCCESSFUL ?
         BNE   ENDBAD                      NO
*
GOTDDEP  EQU   *
         LA    R3,DLET                     POINT TO GHU FUNCTION
         LA    R4,IOAREA                   POINT TO I/O AREA
         LA    R5,DUSSAD                   POINT TO USSA FOR DDE
*
         MVI   XP_RSN+1,X'04'              SET XPCB REASON = 04
         BAS   R14,AIBCALL2                DELETE THE DDEP
         CLI   FLAG,X'00'                  SUCCESSFUL ?
         BE    ENDGOOD                     YES
*
ENDBAD   EQU   *
         MVI   XP_RC+1,X'10'               SET XPCB RC = 16
         B     ENDOFF
*
ENDGOOD  EQU   *
         MVI   XP_RSN+1,X'00'              SET XPCB REASON = 00
*
 ENDOFF   EQU   *
          L     R13,SAVE+4                  LOAD BACKWARD POINTER
          LM    R14,R12,12(R13)             RESTORE REGISTERS
          XR    R15,R15
          BR    R14
*****************************************************************
         EJECT
*
AIBCALL  EQU   *
         ST    R14,SAVE14                  SAVE R14
         MVI   FLAG,X'00'                  SET FLAG = OK
*
         CALL  AIBTDLI,((R3),AIB,(R4),(R5),(R6)),VL
*
        CLC   AIB_RC,FW0000               RC = 00 ?
        BE    AIBEND                      YES
        CLC   AIB_RC,FW2304               CHECK PCB ?
        BE    CHKPCB                      YES
        MVI   FLAG,X'FF'                  SET FLAG = BAD RC
        B     AIBEND
*
CHKPCB   EQU   *
         L     R7,AIB_RES                  GET PCB ADDRESS
         CLC   PCB_STA,FW                  STATUS CODE = "FW"
         BE    AIBEND                      YES
```

```
         MVC   PCBSTAT,PCB_STA            MOVE STATUS
         MVI   FLAG,X'FE'                 SET FLAG = BAD STATUS
         MVI   XP_RSN+1,X'0B'             SET XPCB REASON = 11
*
AIBEND   EQU   *
         L     R14,SAVE14                 RESTORE R14
         BR    R14
```

Appendix C. Special Notices

This publication is intended to help IMS/ESA systems programmers, database and system designers and application programmers and designers plan and implement IMS/ESA sysplex data sharing environment based on a case study of the experiences from one account. The information in this publication is not intended as the specification of any programming interfaces that are provided by IMS/ESA, IRLM or MVS/ESA. See the PUBLICATIONS section of the IBM Programming Announcement for IMS/ESA, IRLM and 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 pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

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:

AS/400	AT
BookManager	CICS
CICS/ESA	CT
DB2	DFSMS
ES/9000	IBM
IMS	IMS/ESA
LANDP	MQ
MQSeries	MVS
MVS/ESA	OS/390
Parallel Sysplex	RACF
Resource Measurement Facility	RMF
RS/6000	S/390
SP	System/390
VTAM	

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

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

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

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

Other trademarks are trademarks of their respective companies.

Appendix D. Related Publications

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

D.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see "How to Get ITSO Redbooks" on page 121.

- *IMS/ESA Multiple Systems Coupling in a Parallel Sysplex*, SG24-4750.
- *IMS/ESA Sysplex Data Sharing*, SG24-4303
- *MVS/ESA Version 5 Sysplex Migration Guide*, SG24-4581
- *OS/390 Parallel Sysplex Application Considerations Presentation Guide*, SG24-4743
- *System/390 MVS Parallel Sysplex Continuous Availability Presentation Guide*, SG24-4502
- *System/390 MVS Parallel Sysplex Continuous Availability SE Guide*, SG24-4503
- *System/390 MVS Parallel Sysplex Performance*, SG24-4356

D.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
Lotus Redbooks Collection	SBOF-6899	SK2T-8039
Tivoli Redbooks Collection	SBOF-6898	SK2T-8044
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
RS/6000 Redbooks Collection (PDF Format)	SBOF-8700	SK2T-8043
Application Development Redbooks Collection	SBOF-7290	SK2T-8037

D.3 Other Publications

These publications are also relevant as further information sources:

- *IMS/ESA Version 5 Customization Guide*, SC26-8020
- *IMS/ESA Version 5 Diagnosis Guide and Reference*, LY27-9620
- *IMS/ESA Version 5 Installation Volume 2: System Definition and Tailoring*, SC26-8024
- *IMS/ESA Version 5 Operations Guide*, SC26-8029
- *IMS/ESA Version 5 Operator's Reference*, SC26-8030
- *IMS/ESA Version 5 Utilities Reference: System*, SC26-8035

© Copyright IBM Corp. 1997

- *MVS/ESA System Modifications*, GC28-1831
- *OS/390 MVS Planning: Workload Management*, GC28-1761
- *OS/390 MVS Programming: Sysplex Services Guide*, GC28-1771
- *OS/390 MVS Setting Up a Sysplex*, GC28-1779
- *OS/390 MVS System Commands*, GC28-1781
- *OS/390 Parallel Sysplex Application Migration*, GC28-1863
- *OS/390 Parallel Sysplex Hardware and Software Migration*, GC28-1862
- *OS/390 Parallel Sysplex Overview*, GC28-1860
- *OS/390 Parallel Sysplex Systems Management*, GC28-1861
- *System/390 MVS Sysplex Overview*, GC28-1208

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 the following command:

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

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

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1998
  ```

  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.ibmlink.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.ibmlink.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.html). 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:

	IBMMAIL	Internet
In United States:	usib6fpl at ibmmail	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	IBM Publications	IBM Direct Services
Publications Customer Support	144-4th Avenue, S.W.	Sortemosevej 21
P.O. Box 29570	Calgary, Alberta T2P 3N5	DK-3450 Allerød
Raleigh, NC 27626-0570	Canada	Denmark
USA		

- **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.ibmlink.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.ibmlink.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.html`). 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.

List of Abbreviations

ABSA	Amalgamated Banks of South Africa
ACB	access control block (VTAM)
	application control block (IMS)
ACBLIB	access control block library data set
AIB	application interface block
APAR	authorized program analysis report
APPC	advanced program-to-program communication, also advanced peer-to-peer communication
ASAP	automatic software alert process
APPLID	VTAM application identifier
ATM	automatic teller machine
BLDS	block level data sharing
BMP	batch message program
CF	coupling facility
CFRM	coupling facility resource manager
CI	control interval
CIC	concurrent image copy
CICS	Customer Information Control System
CMOS	complementary metal oxide semiconductor
CPF	command prefix facility
CPU	central processing unit
CSA	common storage area
DAE	dump analysis elimination
DASD	direct access storage device
DBA	database administrator
DBCTL	database control
DBD	database definition
DBDGEN	database definition generation
DBDLIB	database definition library data set
DBMS	database management system
DBRC	IMS/ESA database recovery control feature
DB2	DATABASE 2
DDEP	direct dependent segment
DEDB	data entry database
DFW	DASD Fast Write
DLI	Data Language 1
DLISAS	DLI separate address space
DMB	data management block
ECSA	extended CSA (common storage area)
EMH	expedited message handler
EPVT	extended private
ERE	emergency restart
ESA	enterprise systems architecture
ETO	extended terminal option
FBSS	financial branch systems service
FMTLIB	format library data set
GRS	global resource serialization
HDAM	hierarchic direct access method
HIPER	high impact or pervasive APAR
HSM	hierarchical storage manager (now part of DFSMS)
I/O	input/output
IBM	International Business Machines Corporation
IFP	interactive fast path program
IMS	Information Management System
IMSASAP	IMS monitor summary and system analysis utility
IMSID	IMS identifier
IMSPARS	IMS performance analysis and reporting utility
IPCS	interactive problem control system
IRLM	IMS resource lock manager
ISC	intersystem communications
ITSO	International Technical Support Organization

© Copyright IBM Corp. 1997

JCL	job control language
JCLLIB	job control language library data set
JES2	job entry subsystem 2 (part of MVS)
KSDS	key sequenced data set
LANDP	IBM LAN distributed platform
LGMSG	long message queue data set
logrec	log record
LPAR	logically partitioned mode
LSR	local shared resources
LU	logical unit (VTAM)
LU0	logical unit type 0
LU6.1	logical unit type 6.1 (also known as ISC)
LU6.2	logical unit type 6.2 (also known as APPC)
MFS	message format service
MPP	message processing program
MSC	multiple systems coupling
MSDB	main storage database
MSDBCP1	main storage database checkpoint data set 1
MSDBCP2	main storage database checkpoint data set 2
MSDBINIT	main storage database initialization data set
MSLINK	multiple systems coupling link
MSNAME	multiple systems coupling name
MTO	master terminal operator
MVS	Multiple Virtual Storage
NetView	network observation tool
OLDS	online log data set
OSAM	overflow sequential access method
OTMA	open transaction manager access
PARMLIB	MVS parameter library data set
PC	personal computer
PC	program call
PCB	program communication block
PE	PTF in error
PGMLOAD	program load library data set
PI	program isolation locking
POS	point of sale terminal
PPT	program properties table
PROCLIB	procedure library data set; IMS procedures are in IMSESA.PROCLIB and MVS procedures are in SYS1.PROCLIB
PROCOPT	processing option
PSB	program specification block, made up of database PCBs and I/O PCBs
PSBLIB	program specification block library data set
PSBGEN	program specification block generation
PSI	problem source identification
PSP	preventive service planning
PTF	program temporary fix
QPP	quality partnership program
RACF	resource access control facility
RECON	recovery control data set
RMF	resource measurement facility
ROI	return on investment
SDEP	sequential dependent segment
SHMSG	short message queue data set
SLC	service level contract
SLR	service level reporter
SLUP	secondary logical unit program
SMF	system management facilities
SMS	system managed storage
SNA	System Network Architecture
SVC	supervisor call instruction (IBM System/390)
SYSLOG	system log
TAF	terminal access facility
TSCFLIB	target system control facility library
TCO	time controlled operations
TPNS	teleprocessing network simulator
UOW	unit of work
VSAM	virtual storage access method

VSO virtual storage option

VTAM virtual telecommunications access method

WADS write ahead data set

XCF cross-system coupling facility

9021 System/390 bipolar processor

9672 System/390 CMOS processor

9674 System/390 CMOS processor

Index

Special Characters

Numerics

A

B

C

© Copyright IBM Corp. 1997

L

M

N

O

P

U

V

W

X

ITSO Redbook Evaluation

IMS/ESA Sysplex Data Sharing: An Implementation Case Study
SG24-4831-00

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.ibm.com
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.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!)**

____

____

____

____

____

© Copyright IBM Corp. 1997

SG24-4831-00
Printed in the U.S.A.