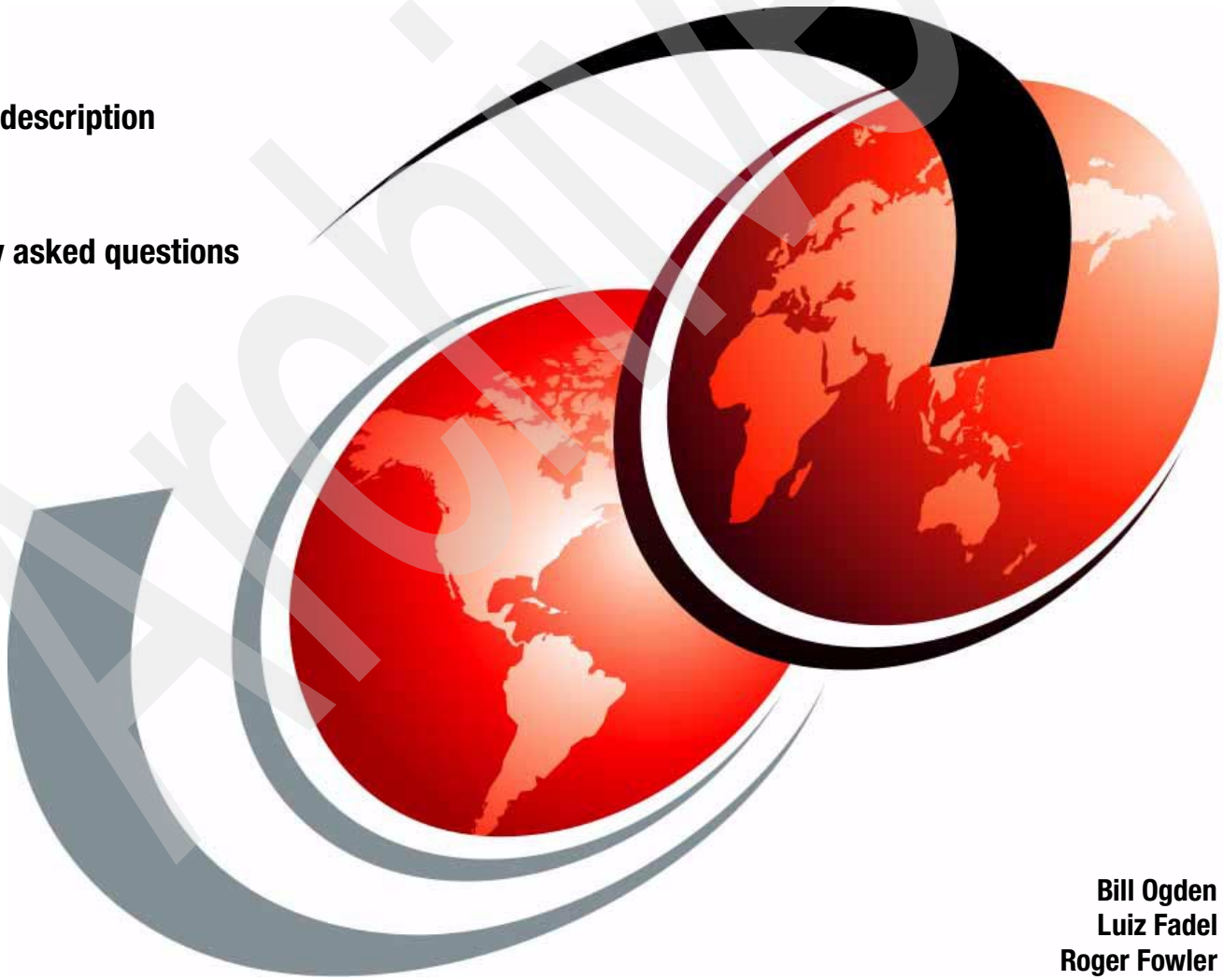
IBM

IBM @server zSeries 990 Technical Introduction

- System overview
- Hardware description
- Frequently asked questions



Bill Ogden
Luiz Fadel
Roger Fowler

Redbooks

ibm.com/redbooks

International Technical Support Organization

IBM @server zSeries 990 Technical Introduction

May 2003

SG24-6863-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

First Edition (May 2003)

This edition applies to the announcement of the IBM @server zSeries 990 system.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

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

@server ™	OS/390®	S/390®
@server™	Parallel Sysplex®	Sysplex Timer®
CICS®	PowerPC®	VSE/ESA™
DirMaint™	PR/SM™	VTAM®
DB2®	Redbooks™	3890™
ESCON®	Redbooks(logo) ™	z/Architecture™
FICON™	Resource Link™	z/OS™
IBM®	RACF®	z/VM®
MVS™	RMF™	zSeries®
OS/2®	S/370™	

The following terms are trademarks of other companies:

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

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

Preface

The IBM® @server™ zSeries® 990 provides major extensions to existing zSeries architecture and capabilities. The concepts of books and channel subsystems are added to the architecture, and the maximum number of LPARs is increased. These architectural extensions provide the base for much larger zSeries machines. This IBM Redbook provides an overview of these changes, and goes into more detail in selected areas.

By removing memory, processor, and channel constraints, z990 can be used for major server consolidation activities. More granular options for nondisruptive maintenance, growth, and alterations provide better 7 x 24 x 365 operation. In general, no changes are needed for existing applications—allowing immediate growth into z990.

Readers are assumed to have a general understanding of existing S/390® and zSeries hardware. Terms, acronyms, abbreviations, and concepts associated with existing systems are used without further introduction.

The authors

Bill Ogden is a retired Senior Technical Staff Member at the International Technical Support Center, Poughkeepsie. Bill joined the ITSO (then the WTSC) in 1978 and has worked with a large number of S/360, S/370™, S/390, zSeries, and associated systems during these years.

Luiz A Fadel is a Senior Certified Consulting IT Specialist with IBM Brazil. He is responsible for zSeries pre-sales and z/OS technical support for Americas South Region (LA Countries). He is also responsible for supporting zSeries or z/OS ESPs and first installation, as well as critical situations in LA.

Roger Fowler is a Consulting IT Specialist with IBM United Kingdom Ltd. He is responsible for technical leadership of z/OS™ and Parallel Sysplex® in EMEA Northern Region, and pre-sales zSeries technical support. He is also the IBM liaison officer for the Guide Share Europe Large Systems Group in the UK. Prior to joining IBM in 1988, Roger worked in large systems technical support for two large energy corporations and a financial institution.

Extensive contributions

The following people provided extensive help and expertise for this redbook.

Bill White, IBM Poughkeepsie, who helped us understand many of the details involved with recent I/O architecture, connectivity, and devices.

Franck Injey, on assignment from IBM France to the ITSO in Poughkeepsie, who helped with details about system setup and migration.

Parwez Hamid, IBM UK, who helped us understand the “big picture” (in detail!).

Additional help

We thank the following people for their contributions to this project:

Joe Turner and **Mike Scoba,** IBM Poughkeepsie, who helped organize this work.

Ken Trowell, recently retired from IBM Australia and now with IBM in Poughkeepsie.

Dale Riedy, IBM Poughkeepsie, who provided much information on z/OS compatibility and exploitation support.

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an Internet note to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYJ Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

1

Introduction

The original S/360 architecture has evolved in many ways since it was introduced in 1964. Two fundamental characteristics of this evolution have been:

- Compatible evolution

  IBM recognizes the total customer investment in data processing solutions and recognizes that application software (and the non-DP processes that are intertwined with these applications) represents the bulk of this investment. Therefore, the entire evolution of the system architecture has been planned with application compatibility in mind.

- Resolving constraints

  Evolving hardware capabilities and evolving application designs produce various stress points in system designs. Over the years these stress points have included CPU performance, usable memory size, addressable memory size, I/O speed, I/O management overhead, I/O attachment capacity, and so forth. System architecture has evolved to relieve constraints in many areas.

Examples of this evolution include:

- Single processor systems evolved to multiple processor systems.
- Real memory systems evolved to virtual memory systems.
- Memory addressing (real and virtual) evolved from 24-bit to 31-bit to 64-bit addressing.
- I/O channels evolved from 16 channels to 256 channels.
- I/O addressing evolved from 3 hex digits (S/360) to 4 hex digits (XA architecture) providing up to approximately 64,000 addresses (device numbers) for I/O devices.
- A single operating system in the machine has evolved to LPARs providing up to 15 system images in one physical system.
- I/O management evolved from complex operating system functionality to System Assistance Processors (SAPs) that handle much of the work.
- I/O performance has evolved from parallel channels to ESCON® channels to FICON™ channels.
- Communication between similar systems has evolved from simple shared DASD to interconnected channels (CTCs) to Coupling Facilities and Parallel Sysplex.
- Advanced I/O interfaces have evolved from intricately programmed control units (such as the IBM 2701, 2702, 2703 products) to front-end processors (such as the IBM 3745) to OSA Express designs.

- I/O programming has evolved from the basic SIO-type instructions of the S/360 to the SSCH-related instructions and subsystem of the S/390, and to the QDIO model available with the z/Series.
- System monitoring and tracing has grown from simple sampling routines (high overhead), to hardware-assisted tracing instructions and to hardware-assisted channel and device usage measurements.
- Instructions for small program models (with strict 12-bit displacements) have been expanded through the *relative and immediate* instructions added to the architecture a few years ago. These instructions, intended primarily for compilers, provide more efficient instructions for larger program models.

The z990 provides another major evolutionary step. Key architectural limits of existing systems are addressed without significant impact to the application compatibility that is so important to IBM's customers. Among other elements, the z990 addresses:

- Total system processing speed. The largest z990 has several times the processing capacity of the largest z900 system.
- Total system memory. A z990 can have 8 GB to 256 GB memory.
- Total number of I/O channels. A z990 is architected to have more than 256 channels.
- Number of system images. A z990 is architected to have more than 15 LPARs.
- Additional channel and device measurements are available.
- Enhanced concurrent maintenance and upgrade capabilities.
- More efficient large program models, using long-displacement instructions.
- More efficient LAN operation, using larger data blocks and offloading more processing from the main processor to the I/O adapters.

Not all of these extensions are available in the initial release of the z990, but they are present in the *architecture* of the system. For example, two Logical Channel Subsystems are implemented in the z990, while the architecture implemented in the z990 allows more.

1.1 Feature and function delivery schedules

The z990 announcement includes functions and features that are available with initial system delivery, and additional functions and features that are planned to become available later. This redbook discusses all the functions that have been announced. Refer to the primary IBM announcement materials to understand which functions and features will be available initially and which will be available later.

The z990 announcement also included several IBM Statements of Direction (SODs) about possible future enhancements. These are briefly discussed in this redbook.

1.2 zSeries system

The z990 is a member of the zSeries family and uses the zArch architecture (formerly known as ESAME Architecture) and instruction set (with some extensions) that is used in the z900 and z990 machines. (The zArch Architecture is commonly known as "64-bit" architecture, although it provides much more than 64-bit capability.)

Completely new with the z990 is the concept of a *book* within a zSeries machine. The z990 offers four models: A08 (with one book), B16 (with two books), C24 (with three books), and D32 (with four books).

A book contains processors (PUs in normal zSeries terminology), memory, and connections to the I/O *cages*. A z990 has space for 1 to 4 books.[1] Each book has:

- 12 processor units.
  - Within each book, a maximum of 8 processors may be used as CPUs. (Two are used as SAPs and two are reserved as spares.)
- 8 to 64 GB memory.
- Up to 12 connections to the I/O cages and/or ICB coupling channels.

While memory is resident in the book(s) it is used as a single SMP. All memory is addressable by any processor in any book. Stated another way, a given processor can address all the memory in the system and all the memory is coherent to programs.[2] L1 and L2 caches exist to improve performance, but they are not directly visible to programs. The memory structure is explored in more detail in “Memory” on page 12.

A z990 *book* might sometimes be labeled a *node*. IBM is not using the node terminology because it creates an incorrect impression. The term node is often used for an independent processor, or, in a sense, a loosely-coupled processor. This is not the case for the z990. All processors and memory, on all books, are part of a single symmetric multiprocessor (SMP) system.

The processors (normally known as Processor Units or PUs) are new. There are one or two processors on a chip (“single core” or “dual core” design), and each of the processors has the duplicate I and E units that are used for internal checking in all zSeries machines. Processor cycle time is .83 ns.[3] A fixed mixture of single and dual core chips is used in every system and, their positions are fixed by the design of the underlying ceramic carrier. Each processor provides superscalar operation, meaning that multiple instructions may begin execution in a cycle and some elements of an instruction may be executed out of sequential order.

z900 machines can have multiple instructions in process during a single cycle, while the z990 can have multiple instructions start and/or complete during a single cycle. Instruction completion is always in the order of the original instructions, but the z990 can perform memory accesses (required by an instruction) out of order. The net effect is improved performance. The degree of improved performance depends somewhat on the nature of the workload. The processor is discussed in more detail in “Processors and MCMs” on page 10. Performance considerations are discussed in “Performance factors” on page 29.

All z990 systems have two frames. These appear roughly the same as a two-frame z900 system and have the same frame names (A and Z frames). Some, but not all, I/O cards from a z900 can be used in a z990 system. The I/O cards are placed in I/O cages (*Cargo* cages[4]) that are similar to those in a z900. The A frame contains the processor/memory books and one I/O cage. The Z frame contains 0, 1 or 2 I/O cages and two ThinkPads® that are used as Support Elements.

The frames contain extensive power supplies, system controls, and a new hybrid cooling system. A more extensive description of the frames appears in “Power and cooling” on page 21.

z990 systems run only in *LPAR mode*, and there is no *basic mode*. Up to 30 LPARs may be defined. There are new rules associated with LPAR[5] definitions and a new type of

[1] Again, we stress that not all elements are available for initial z990 shipments. For example, only two books may be ordered for initial shipments.
[2] There may be minor exceptions to this statement for certain privileged state-changing instructions.
[3] This equates to a 1.2 GHz clock rate. However, we avoid this terminology because it leads to completely incorrect comparisons with clock rates in other architectures (in the same way that zSeries MIPS often are compared incorrectly with various metrics in other architectures).
[4] The IBM development name for these I/O cages was *Cargo*, and this name is still informally used.

definition—for Logical Channel Subsystems (LCSSs)—is required. This is discussed in more detail in “Channel subsystem” on page 32 and throughout this redbook. Briefly, the key rules include:

- Every LPAR must be associated with a Logical Channel Subsystem. Multiple LPARs (up to 15) can be associated with a single Logical Channel Subsystem.
- A Logical Channel Subsystem can define and use a maximum of 256 channels (CHPIDs).
- Some channel types, such as ESCON, can be associated with only a single Logical Channel Subsystem. Other channel types, such as HiperSockets, can be associated with multiple Logical Channel Subsystems.
- Logical Channel Subsystem definitions are part of the overall IOCDS definition. These definitions may be done through HCD or through direct IOCP statements in the operating system, Support Element or HMC.

1.3 New terminology

A number of new terms are associated with the z990 machines. Important new terms (and concepts) include:

- Books

  A book contains processors, memory, and connections to I/O cages. A z990 can have one to four books.

- Logical Channel Subsystems (LCSSs)

  A LCSS contains up to 256 CHPIDs. An LPAR (or several LPARs) is associated with a single LCSS. An LPAR (and the operating system running in it) sees a maximum of 256 CHPIDS, and all the I/O control functions are compatible with earlier systems which had a maximum of 256 CHPIDs.

  Note: LCSSs are not an optional feature; the capability and *requirement* of using LCSSs is inherent in z990 systems.[6]

- A Physical Channel ID (PCHID)

  This reflects the physical identity of a channel-type interface and has a range of X’000’ through X’6FF’.[7] A PCHID number is based on the I/O cage number, the adapter slot number, and the connector number of the I/O adapter. A unique PCHID exists for every potential channel interface connector in the system. See “I/O interfaces and identification” on page 18 for a table of PCHID numbers.

  A CSS.CHPID is assigned to a PCHID. The traditional CHPID numbers are limited to one byte or 2 hexadecimal digits. They are in the range 0 - 255. As announced, a z990 system may have up to two Logical Channel Subsystems. A Logical Channel Subsystem (LCSS) can have a maximum of 256 CHPIDs.

  New mapping specifications (in an IOCDS) are used to map PCHIDs to CHPIDs within an LCSS. A notation such as 1.C1 = 23F means CHPID C1 in LCSS 1 is assigned to PCHID 23E. PCHID 23E is connector 14 in the adapter in slot 22 in I/O cage 1. This is covered in more detail in “Channel subsystem” on page 32.

- System-used and customer-used PUs:
  - *System-used* PUs are used for basic SAPs or spares.

[5] Strictly speaking, “LPAR” is the name of a system mode that uses logical partitions (LPs). However, the use of “LPAR” to indicate a logical partition is so widespread that we continue such usage in this redbook.

[6] HCD and IOCP definitions will default to LCSS 0 if an LCSS is not specified, so it is possible to create a working IOCDS without LCSS parameters in HCD or IOCP source statements.

[7] Not all this range is used.

- *Customer-used* PUs may be used as CPUs, ILFs, ICFs, or additional SAPs.
- A customer may purchase a CP or IFL and leave it *unassigned*. This might be done for software cost control purposes. A later IBM MES action is needed to enable the PU.
- Otherwise unassigned PUs are use for sparing after the reserved spare PUs are used.

Do not confuse the hardware I/O subsystem with Logical Channel Subsystem. A z990, regardless of the number of books or PUs installed, has only one hardware I/O channel subsystem. However, it can have multiple Logical Channel Subsystems (LCSSs). The multiple Logical Channel Subsystems permit the system (typically via multiple operating systems in multiple LPARs) to address and use more than 256 channels (CHPIDs).

1.4 z990 models

The z990 is offered in four models, although only the first two models listed here are available for initial shipments. The model is directly related to the number of books included in the system. The number of processor units purchased for each book is selected by feature codes and is not reflected in the system model number. The models are:

- IBM 2084-A08 - (One book, with up to 12 processors, 12 STI connectors, with a maximum of 8 customer-used PUs and a maximum of 64 GB memory)
- IBM 2084-B16 - (Two books, with up to 24 processors, 24 STI connectors, with a maximum of 16 customer-used PUs and a maximum of 128 GB memory)
- IBM 2084-C24 - (Three books, with up to 36 processors, 36 STI connectors, with a maximum of 24 customer-used PUs and a maximum of 192 MB memory)
- IBM 2084-D32 - (Four books, with up to 48 processors, 48 STI connectors, with a maximum of 32 customer-used PUs and a maximum of 256 GB memory)

As can be seen, the A, B, C, or D in the model number indicates 1, 2, 3, or 4 books. The number after the A, B, C, or D indicates the *maximum* number of zSeries CPs/IFLs/ICFs that can be ordered and enabled. (Remember that additional processors are included for SAPs and spares.)

Note that the system model number no longer reflects the number of processors that are enabled for use. It reflects only the maximum number of customer-used processors that could be enabled (if purchased). The model number also reflects the number of books installed.

1.5 Maximum channels

The maximum number of channels for a z990 depends on a number of factors:

- The maximum number of *usable* CHPIDs is 256 x LCSSs; that is, 256 (the maximum number of CHPIDS that can be addressed by a Logical Channel Subsystem) times the number of Logical Channel Subsystems used. The z990 announcement provides for two Logical Channel Subsystems, allowing up to 512 CHPIDS to be used.[8]
- The number of STI connections available for I/O cages. A z990 model A08, for example, has 12 STI connections available and this limits the number of I/O slots that can be used.
- The number of I/O adapter slots available. There are three I/O cages, each containing 28 slots for I/O adapters.

[8] This number is reduced if spanned channels are used. Also, remember that a single z/OS image supports a maximum of 256 channels.

- The number of channels on various I/O adapters. For example, an ESCON adapter can provide up to 15 usable channels, while a FICON adapter provides two channels.
- Maximum limitations for various I/O adapters. For example, a maximum of 60 FICON adapters (providing 120 channels) can be installed.[9]

It is possible to install more physical channels than can be used at any one time. Channel adapters are discussed in more detail in “I/O cages and adapters” on page 14.

1.6 Migration

The z990 introduces significant new architectural changes. Software exploitation of these changes is expected to be introduced in phases. A migration from an existing zSeries system to a z990, based on the software available with initial z990 shipments, should be relatively simple.

z/OS functions available with initial z990 shipments use only a single Logical Channel Subsystem. This provides an environment very similar to that of current zSeries machines. The most significant migration activity at this level might be an IOCP (or HCD equivalent) move. A z990 contains more information in an IOCP. A CHPID Mapping Tool is available and should ease this migration. The new PCHID addressing level can be used to make the CHPID numbers for the z990 generally match the CHPID numbers in an existing zSeries machine. This can considerably reduce the effort in migrating a complex IOCDS.

These aspects of migration are discussed in more detail in “Migrating the I/O configuration definitions with HCD” on page 98.

1.7 Statements of direction

IBM included a number of Statements of Direction (SODs) with the z990 announcement. Refer to the formal announcement material for the complete list and exact formulation of these statements. In this section we briefly mention several of the SODs.

- IBM intends to support four Logical Channel Subsystems on the z990.

  The announced support is for two LCSSs. This redbook discusses multiple Logical Channel Subsystems in general, although specific examples use only two LCSSs. The maximum number of channels supported on a single z/OS image will be 256.
- IBM intends to support up to 60 LPARs on the z990.

  The announced support is limited to 30 LPARs.
- IBM intends to support greater than 16 engines in a single image on appropriate releases of z/OS and z/VM® on the z990.
- IBM intends to support an optional Smart Card Reader, attached to a TKE workstation, permitting secure storage, convenient transport, and rapid re-entry of key parts for the z990.

  The PCIXCC adapter is planned to be available later.
- IBM intends to provide z/VM guest support of the PCIX Cryptographic Coprocessor (PCIXCC) adapter.

Except for a few tables listing maximum values (which correspond to the announced z990) the discussions in this redbook should be compatible with these statements of direction.

[9] The number of OSA-Express adapters installed and other factors can reduce this number.

1.8 Limitations and considerations

There are several key z990 considerations when migrating from earlier systems. These include:

- Parallel ("bus and tag") channels are not supported. Converters (which connect an ESCON channel to a parallel control unit) are supported. The details are the same as for the z800 systems. IBM 9034 converters (commonly known as *Pacer* boxes) may be used, as well as converters from Optica Technologies, Incorporated. See "Parallel channel planning" on page 70 for more details.
- Hardware cryptographic functions are different:
  - The CMOS cryptographic coprocessors are not available.
  - The PCICC cryptographic adapters are not available.
  - The PCICA cryptographic adapter *is* available.
  - New cryptographic-assist instructions are included in *every* processor unit.
  - A new PCIX Cryptographic Coprocessor (PCIXCC) adapter will be available October 31, 2003.

  More details about the cryptographic changes can be found in "Sysplex coupling considerations" on page 49.
- Older channel and communications adapters cannot be used. These include:
  - FICON adapters; FICON Express adapters are used instead.
  - OSA2 adapters; OSE-Express adapters are used instead.
    - There is no replacement for FDDI adapters.
  - ATM adapters (which are OSA-Express adapters). Customers using ATM functions should consider using ATM interfaces in outboard switches and routers.
- Existing OSA-Express Fast Ethernet adapters on z900 systems can be moved to a z990, but new orders must be for the new 1000BaseT Ethernet adapter.
- A z990 IOCDS must contain PCHID numbers that assign CHPIDs to specific hardware channels. These assignments must be added to existing HCD (and IOCP) definitions.
- The z/OS versions available with initial z990 shipments can use only the first LCSS.
- If appropriate, you must use a new IOCP program that includes support for z990 functions.

2

Hardware details

This chapter provides an overview of the z990 hardware, with some comparisons to earlier systems. Many of the features, functions, and considerations briefly described here are further discussed in “Discussion topics” on page 29.

A simplified view of the building blocks of a z990 system is shown in Figure 2-1. This figure is not intended to depict the operation of a z990 system. It simply lists the general elements discussed in this chapter and roughly indicates how they are related to each other.

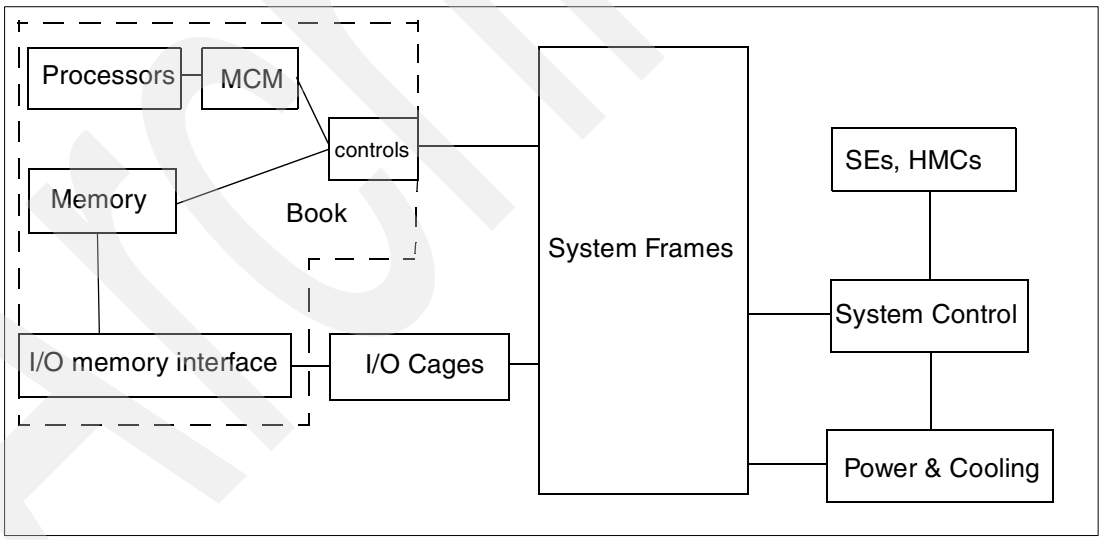


Figure 2-1 Organization of descriptions in this chapter

2.1 System frames

All z990 systems have two frames. The general content of the frames is shown in Figure 2-2 on page 10. The actual design (especially for power and cooling components) is much more complex than indicated, of course, but this diagram serves as a starting point.

The Z frame is always present, even if no I/O cages are installed in it. It will always have the two ThinkPads used as Support Elements and basic power components. The ThinkPads are mounted somewhat differently than on z900 or z800 machines, but they fold out for use in the same general manner as with a z900 machine.

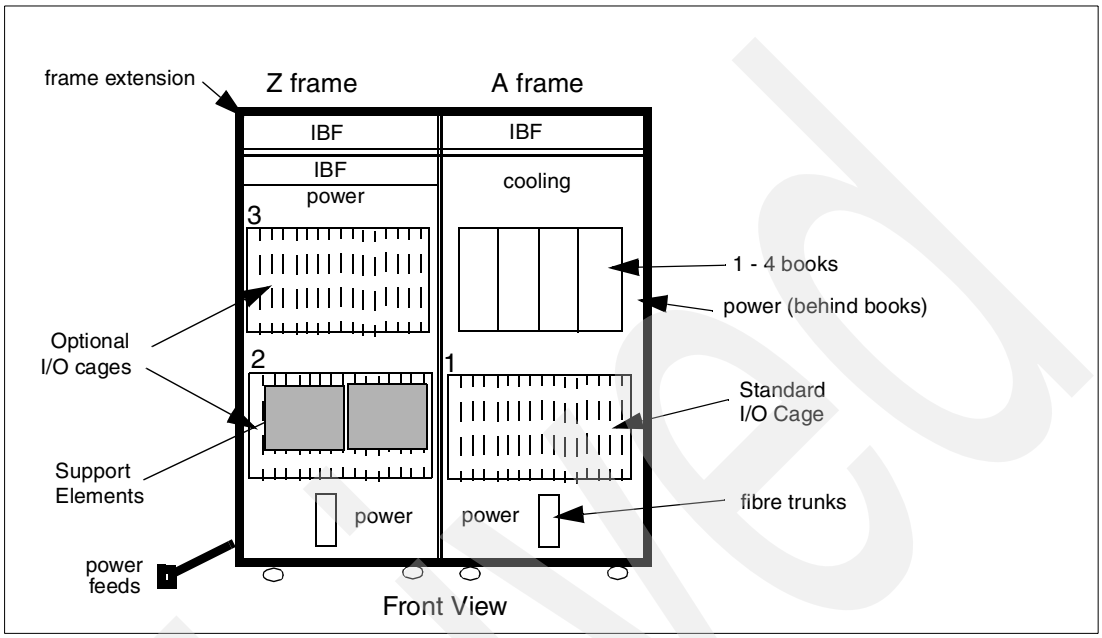


Figure 2-2 General frame arrangement

The minimum system is a model A08 (which has one book). The books contain the processors and memory of the system.

The basic frames are 40 EIU units high, with a breakpoint at 38 EIUs. I/O cage 1 is in the bottom of the A frame. I/O cage 2 is in the bottom of the Z frame, and I/O cage 3 is in the top of the Z frame.

The approximate dimensions are:

- 158 cm deep (62 inches), including the covers
- 194 cm high (76 inches), including the casters
- 154 cm wide (61 inches), both frames together
- 705 kg (1550 pounds) for the A frame (without the internal battery feature)
- 790 kg (1738 pounds) for the A frame (with the internal battery feature)
- 682 kg (1500 pounds) for the Z frame (without the internal battery feature)
- 767 kg (1688 pounds) for the Z frame (with the internal battery feature)

These weights assume a full load of I/O adapters, but do not include the covers. The weights may change slightly depending on the exact mixture of I/O cards used. The top sections (used for the battery feature) can be removed, if necessary, to move the system through a smaller opening.

2.2 Processors and MCMs

There are eight customer-available PUs on each MCM. Each resides on a separate processor chip, using all of the eight processor chip sites available. In addition, two SAP and two spare PUs are needed. These are implemented with *dual core* chips containing two complete PUs. Each PU has dual instruction and execution units (I and E units) that are used

to verify internal operation. These additional I/E units, used for internal checking within a PU, are not further mentioned in this discussion. Do not confuse them with the two full-function PUs that are on a dual-core processor chip.

The processors have a superscalar design, meaning that several instructions may be processed, started, and/or ended at once and that some processing steps within an instruction may be out of sequential order. Each processor (and there are 12 per book, in a mixture of single-core and dual-core chips) can decode up to 3 instructions per cycle and complete up to 3 instructions per cycle. Some instructions complete in a single cycle, while others involve a pipeline of internal steps. Storage accesses may be out of sequential order, but instruction completion is in sequential order. (In these comments, *sequential order* means the order in which the instructions appear to be issued by the program.)

Each processor has a dual L1 cache, meaning that separate caches exist for instructions and data. Both I and D caches are 256 KB. Both the I and D caches have their own Translation Lookaside Buffer (TLB) with 512 entries. In addition, there is a new second-level TLB with 4096 entries. There is also a Branch History Table (BHT) with 8 K entries. All of these caches, TLBs, and BHT are four-way associative.

An L2 cache exists in each book and is 32 MB. The L2 caches of all books are connected through a ring structure, and it is the L2 cache operation that provides the unified, coherent memory seen by all books.

Floating point operation has been redesigned to enhance Binary Floating Point (BFP) performance, bringing it to at least the same level as traditional Hexadecimal Floating Point (HFP) performance.

New functions have been added to assist cryptographic operations. These are new (synchronous) instructions for several symmetric (clear key) cryptographic operations. Every PU has these new functions. (The separate cryptographic coprocessors that were associated with a small number of PUs in earlier systems no longer exist.)

The processor chips are mounted on a Multiple Chip Module (MCM), along with a number of other chips required at this level, as roughly illustrated in Figure 2-3. MCMs are the building blocks for all recent S/390 and zSeries machines. The MCM itself is glass-ceramic with many internal layers of connections. The z990 MCM is 93x93 mm (smaller than the z900 MCM) and is built without internal thin-film layers. (This provides simpler and more robust construction.)

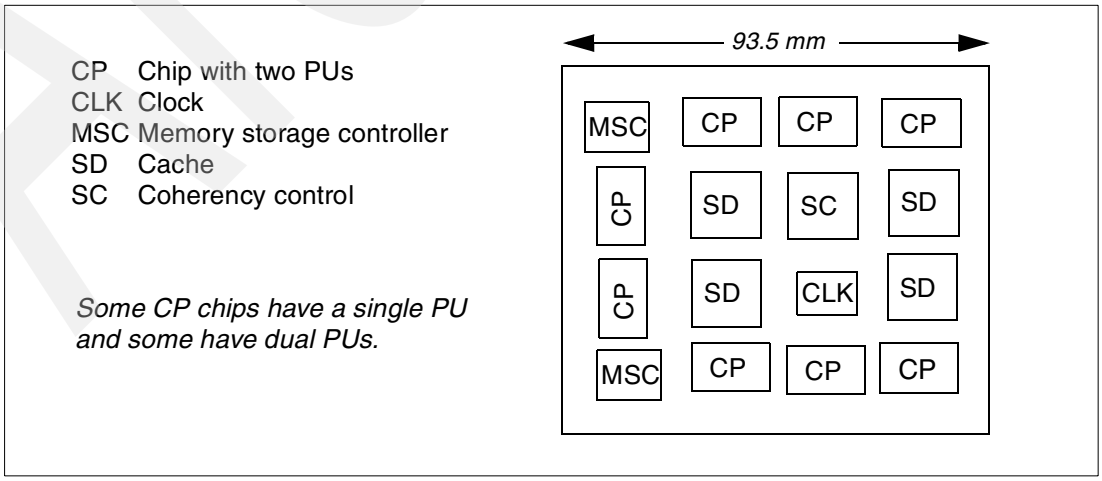


Figure 2-3 General MCM layout

The chip notation in the figure is sometimes written as SCD (instead of SD) for cache chips and SCE (instead of SC) for coherence and control chips. Also remember that some processor chips have two PUs on them. There are a total of 12 PUs on the MCM, of which a maximum of 8 can be ordered for customer use as CPs, IFLs, or ICFs.

The user cannot select specific PUs as spares, SAPs, or CP/IFL/ICFs, or disabled. This selection is made by the system control code and can change due to configuration changes or sparing actions.

The memory interface chips (MBAs) are not on the MCM, but on separate riser cards in the book. This is a design difference from z900 machines, where the MBAs were on the MCM.

2.3 Book package

Books are sometimes described as *cards*, but a book physically appears as a box. A book plugs into one of four slots in the processor cage of the z990. A model A08 system has at least one book and a model D32 has four books. One or two memory "cards", the MCM, and various other chips and connectors are mounted in each book. A conceptual view of a book is shown in Figure 2-4. A book is fairly large, about 56 cm high and 14 cm wide (about 22 inches high and 5.5 inches wide) and weighs about 32 kg (70 pounds) with both memory cards fully populated.

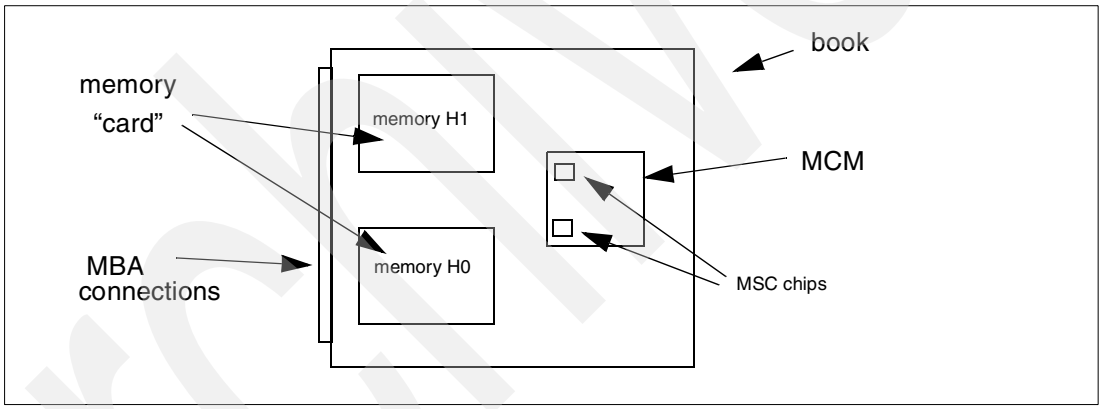


Figure 2-4 Physical book layout (with the external "box" cover removed)

The book is physically large for several reasons. The size makes cooling easier. It also matches the size and connectors on several power supplies that are placed in the back of the same frame cage that holds the books. (The books are in the front of this cage.)

2.4 Memory

A z990 book contains an MCM (with processors) and two memory cards. These cards also might more accurately be described as *boxes*. They are enclosed units containing DIMMs (strips with memory chips). A memory card is removable as a unit. Each card physically contains 8, 16, or 32 GB of memory[1]. Two cards are always present for every book and must have the same amount of memory. Two 32 GB cards provide 64 GB, which is the maximum memory for a book.

[1] These numbers are the amount of usable memory. Considerably more memory is actually present and is used for checking, redundancy, and sparing purposes.

Different books may have different memory sizes. The system memory is the total of the book memories. A book may have more memory installed than is enabled. IBM LICCC[2] specifications determine how much memory is available for use. Customers may order memory in increments of 8 GB per book. IBM controls the distribution of purchased memory across the system books.

Each card provides two memory ports. Each port can process four *fetch* and four *store* operations concurrently. Each access operation provides 128 bits (16 bytes) of usable data. (The storage path is actually 140 bits wide and includes a sophisticated sparing function.) While theoretical rates are higher, a practical workload may transfer up to about 7 GB/second for each port, thus providing up to approximately 28 GB/second for memory access in a book.

Spare memory chips are not used for main memory. Spare memory elements, used automatically as needed, are present in a distributed fashion. A description of the sophisticated sparing process is beyond the scope of this redbook.

Separate circuitry is used for storage protect keys and provides triple redundancy with a voting mechanism in case of errors. An additional spare set of memory for storage protect keys is present and is used to replace a failing set. System control circuitry keeps track of protect key memory errors and decides when to switch to the spare memory set.

Figure 2-5 provides a high-level conceptual illustration of the memory connectivity structure.

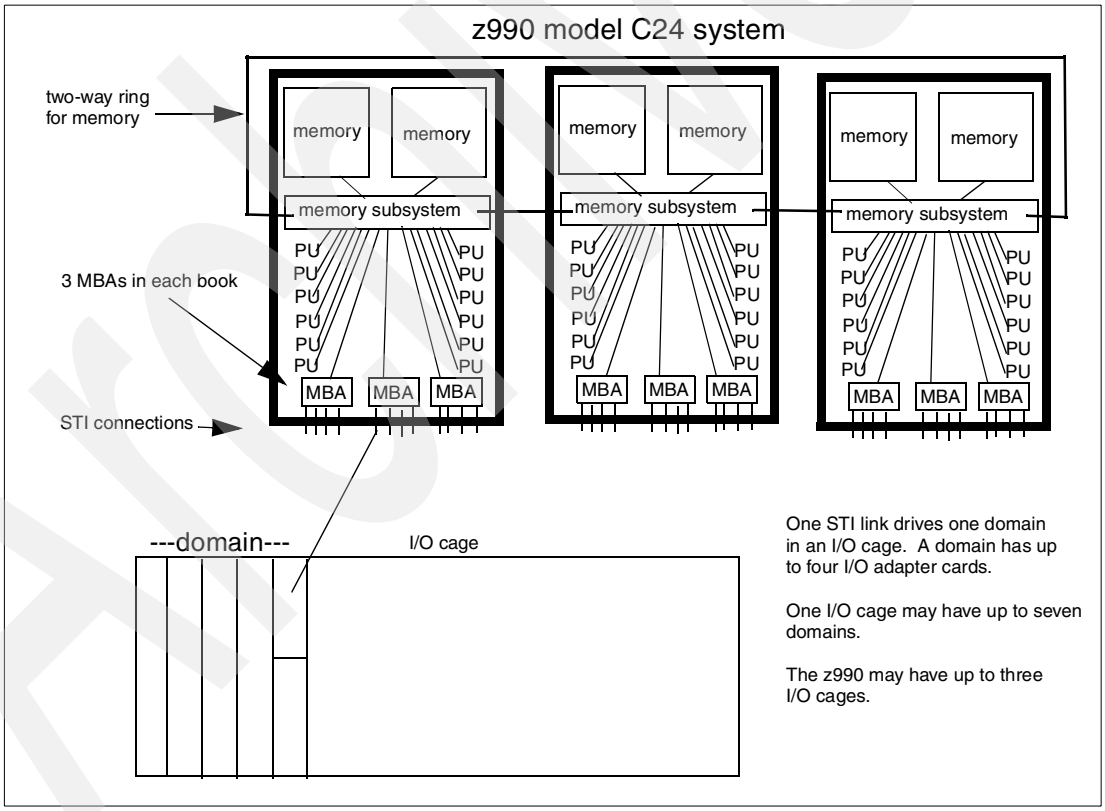


Figure 2-5 Overview of memory access for the z990 with three books

[2] IBM Licensed Internal Code Controlled Configuration.

2.5 z990 models

A z990 system has one to four books. Each book has 0 - 8 customer-used PUs, although a book with no customer-configured PUs would be unusual. (The system must have at least one configured CP, IFL, or ICF.) Each book has 8 - 32 GB memory, and each book has 12 STI connections. These interfaces are the links to external I/O devices and to specialized adapters.

Figure 2-5 on page 13 illustrates a model C32 system with three books. Note the *ring* that connects the memory subsystems of the three books together. This is a key element of the z990. It is actually two rings, one running clockwise and one running counterclockwise. Both loops are normally used as closed loops for maximum bandwidth and minimum latency; if necessary, they can run in open modes with reduced performance. This memory subsystem provides the L2 cache.

The memory ring is used to cause the memory in all the books to function as a single memory system. For example, if each book in the system shown in Figure 2-5 contains 32 GB, then the total system functions with a single 96 GB memory. The memory subsystem coordinates cache, STI accesses, memory ring functions, and local memory access such that the total system memory appears as a single coherent memory. Any PU or I/O connection in any book can access memory anywhere in the system. Software does not see memory as separated into books.

Different books can contain different amounts of memory. The system memory is the total of the books' memory.

Two oscillator cards (for redundancy) and two External Time Reference (ETR) cards (for redundancy) are shared by all books. ETR functions are discussed in "External time reference" on page 25.

2.6 PU characterization

PUs are ordered in single increments. A system must have one PU enabled as a CP, IFL, or ICF. Otherwise, the number of PUs ordered is not tied to the z990 model that is ordered. You could, for example, order a model D32 (four books with a maximum of 32 PUs enabled as CP/IFL/ICFs) and order only a single CP.

PUs are *characterized*. That is, they are assigned to be a CP, an IFL, an ICF, or an additional SAP. (There are also options to order unassigned PUs. This is discussed in "Concurrent system upgrades" on page 38.) You can place an order (an MES order) for IBM to change the characterization of your PUs.

2.7 I/O cages and adapters

The I/O cages are the same as used with z900 machines, except that the DCAs (power supplies and cage controllers) and STI cards are new. A cage is roughly outlined in Figure 2-6 on page 15.

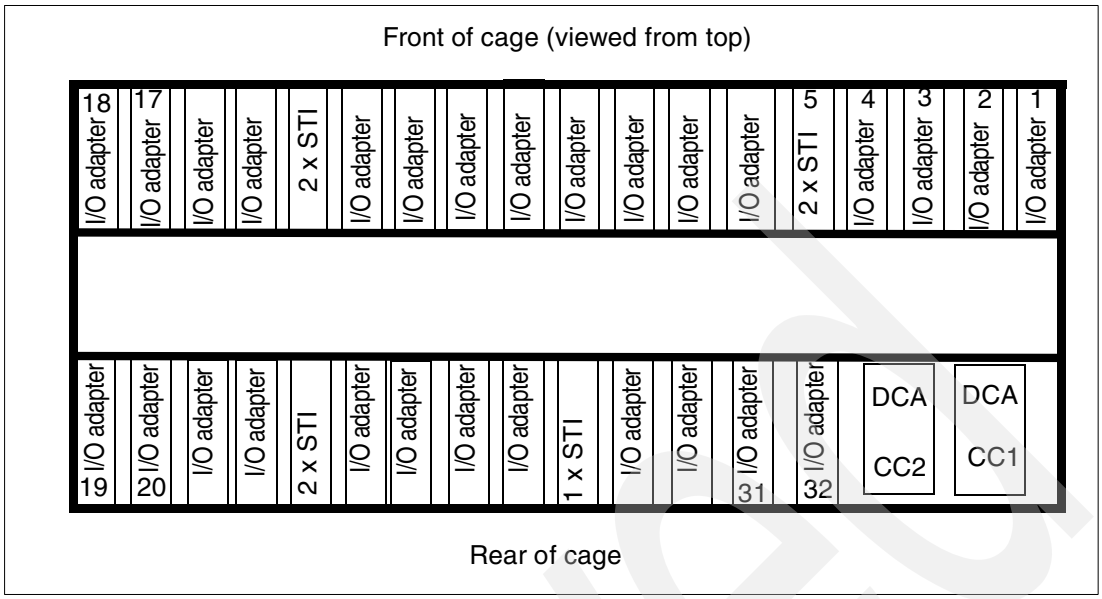


Figure 2-6 I/O cage

Notice that slots 5, 14, 23, and 28 are used for STI connections. There are two such connections in slots 5, 14, and 23, while slot 28 has a single STI connection. (The “other half” of slot 28 may be used for a power sequence control (PSC) card.) IBM selects which slots are used for I/O adapters and supplies an appropriate number of cages and STI cables. An STI cable goes directly from an I/O cage to an STI connector on a book. Each STI connection controls a *domain* of four I/O slots.

The assignment of slots to domains for each I/O cage is as follows:

Domain	I/O Slots in domain
0	1, 3, 6, 8
1	2, 4, 7, 9
2	10, 12, 15, 17
3	11, 13, 16, 18
4	19, 21, 24, 26
5	20, 22, 25, 27
6	29, 30, 31, 32

2.7.1 I/O adapters

The I/O adapters initially available for the z990 are a subset of the adapters used with the z900, plus one new adapter. The subset excludes parallel channel, FDDI, ATM, and OSA-2 adapters. The various adapters that may be used are discussed at greater length in Chapter 4, “Software considerations” on page 79. Table 2-1 provides a brief summary:

Table 2-1 Summary of I/O adapters

Description	I/O Cage slot?	Increment (ports)	Ports per card	Maximum cards	Max ports (CHPIDs)	Notes
ESCON channels	Yes	4	15 max	69	1024	1
FICON Express	Yes	2	2	60	120	2, 6
Fast Ethernet	Yes	2	2	12	24	3, 4, 6
Gigabit Ethernet	Yes	2	2	24	48	3, 6

Description	I/O Cage slot?	Increment (ports)	Ports per card	Maximum cards	Max ports (CHPIDs)	Notes
High Speed token ring	Yes	2	2	24	48	3, 6
1000BaseT Ethernet	Yes	2	2	24	48	3, 6
PCICA crypto	Yes	2 engines	0	6		
PCIXCC crypto	Yes	1 engine	0	4		
ICB-4 (CF link)	No	1	---	---	16 links	5
ICB-3 (CF link)	Yes	1 link	2	8	16 links	5
ICB (ICB-2) (CF link)	Yes	1 link	2	4	8 links	5
ISC-3 (CF link)	Yes	1 link	4	8	32	5
HiperSockets	No	----	-----	-----	16	
IC channel (CF link)	No	---	---	---	32	

Note 1. Remember that a maximum of 512 channels are available initially.
Note 2. FICON Express adapters are also used for FCP channels.
Note 3. The total number of OSA Express cards is limited to 24.
Note 4. Existing OSA-Express Fast Ethernet adapters on z900 systems may be carried forward, but new adapters cannot be ordered.
Note 5. A maximum of 32 external CF links plus 32 internal CF links may exist.
Note 6. The total number of FICON Express Adapters plus OSA-Express adapters plus PCICA/PCIXCC cards is limited to 20 in a single I/O cage.

The maximums listed in this table assume that sufficient I/O cages and books (to provide STI connection) are installed.

2.7.2 New I/O adapters

The OSA-Express 1000BaseT Ethernet Adapter is new and replaces the OSA-Express Fast Ethernet adapter. (As noted, existing OSA-Express Fast Ethernet Adapters may be carried forward from z900 systems, but new ones cannot be ordered.) The new adapter uses copper connections and offers 1000 Mbps, as well as 100 Mpbs and 10 Mpbs, operation. This means that two gigabit Ethernet adapters are available:

- The OSA-Express Gigabit Ethernet Adapter uses fiber connections.
- The OSA-Express 1000BaseT Ethernet Adapter uses copper connections.

As with other OSA-Express adapters, two ports (channels or CHPIDs) are installed in each adapter and are used independently.

A new OSA-Express Gigabit Ethernet adapter is available and provides new fiber connectors. Both new adapters (1000BaseT and Gb) provide a checksum offload function that will be supported by z/OS 1.5 for use with IPv4 (but not IPv6). This function reduces CPU overhead during IP transfers.

The PCIXCC adapter (cryptographic adapter) is new and requires an I/O slot for each adapter.

2.7.3 I/O adapter configuration rules

The following general rules apply for z990 systems:

- A maximum of three I/O cages, with 28 adapter slots per cage, may be used. Each domain (4 slots) within an I/O card requires an STI connection.
- OSA-Express cards include:
  - High Speed token ring (HSTR)
  - Gigabit Ethernet (GbE)
  - Fast Ethernet (FE) for adapters carried forward from a z900 system
  - 1000BaseT Ethernet, which replaces the OSA Express Fast Ethernet adapter
- OSA-Express features are ordered in increments of two ports (equal to one adapter card)
  - There is maximum of 24 OSA-Express cards per system.
- For FICON Express, OSA-Express, PCICA, and PCIXCC cards, taken together, the maximum is 20 of these cards per I/O cage.
- There is a maximum of 60 FICON Express cards (120 channels) per system, less any restrictions imposed by the number of OSA-Express, PCICA, and PCIXCC cards installed.
- There is a maximum of 6 PCICA cards per system and 2 per I/O cage.
- There is a maximum of 4 PCIXCC cards per system and 2 per I/O cage.
- ESCON adapters
  - Only the 16-port ESCON adapters may be used.
  - The first ESCON channel(s) require two adapters.
  - After the first two adapters, single adapters are added.
  - ESCON channels are ordered in increments of 4.
  - IBM will determine the number of adapters needed, based on the number of ESCON channels ordered. All the potential channels on an adapter might not be enabled, depending on the total number of channels ordered.
  - A maximum of 15 of the 16 ports on an adapter will be enabled. The 16th port is a spare.
- ICB links
  - ICB-4 links are for a z990-to-z990 connection and operate at 2.0 GB/second. These links use direct connections to a book and have no connections to I/O cages. There maximum number of links is 16 per system, but may be further limited by the number of free STI connections on the installed books. A special cable is needed for ICB-4 connections.
  - ICB-3 links are for connections to zSeries machines and operate at 1 GB/second. An STI-3 adapter card is used in an I/O cage, and this provides two ICB-3 ports. There is a maximum of 8 STI-3 cards (providing 16 ICB-3 links) per system.
  - ICB (also known as ICB-2) links are for connections to IBM 9672 G5/G6 machines (and can be used for connections to other zSeries machines) and operate at 333 MB/second. The STI-2 adapter card plugs into an I/O cage and provides two ICB-2 ports. A maximum of 4 adapter cards (providing 8 links) may be ordered.
- ISC-3 links
  - ISC-3 links are for connections to z900, z800, and 9672 processors, and run at 100 or 200 MB/second. A maximum of 32 ISC-3 links may be used. These are packaged on I/O adapter cards, and each card can have two daughter cards. Each daughter card has two ISC links.
- ICB and ISC links
  - The maximum number of external coupling links (ICB-2, ICB-3, ICB-4, active ISCs) is 32 per system. In addition, a maximum of 32 internal links (ICs) can be used.

The complete I/O configuration rules are more complex than outlined here. Consult your IBM representative for more specific and timely information.

You can order and install more physical channels than you can address. You can address 256 CHPIDs (that is, channels) with a Logical Channel Subsystem. As announced, two Logical Channel Subsystems may be used with the z990, allowing up to 512 CHPIDs (channels) to be addressed.[3] Each CHPID can be associated with a single channel. The

[3] There are additional limitations if you use spanned channels.

physical channel (which, in practical terms, is a connector jack on an I/O adapter) is specified by its PCHID number. You can have more installed PCHIDs (physical channels) than CHPIDs. Your IOCDS definitions connect each CHPID to a PCHID. You must have all your defined CHPIDs connected to PCHIDs to have a valid IOCDS. However, you need not define all your installed PCHIDs (physical channels) in your IOCDS.

In theory, for example, you could install 69 ESCON adapters and enable 1024 channels,[4] each with a PCHID number. Using two Logical Channel Subsystem definitions to their fullest extent, you could only address 512 of these channels while using a given IOCDS. (Of course, you could define another IOCDS that used a different set of 512 channels.) If the z990 were to implement four Logical Channel Subsystems, for example, you could address a maximum of 1024 channels. That is, the *architecture* of the system allows more Logical Channel Subsystems than are implemented in the z990, and this architecture allows a large number of channels to be addressed as CHPIDs.

Logical Channel Subsystems are, in a sense, entities provided by the system firmware. They are discussed in more detail in “Channel subsystem” on page 32.

2.8 I/O interfaces and identification

The I/O design and implementation is very similar to that of the z900. The primary difference is that the 1 GB/second STI links of the z900 have been replaced with 2.0 GB/second STI links. The general data flow structure is conceptually illustrated in Figure 2-5 on page 13, and works like this:

- Each book has 12 STI links, regardless of the number of PUs active in the book.
- An STI link (from a book) is connected to an STI-M card in an I/O cage. Two STI-M cards fit in a single slot in the I/O cage. These use slots 5, 14, 23, and 28 in the I/O cage. Slot 28 can hold only one STI-M card. This provides for a maximum of 7 STI-M cards in an I/O cage.
- Each STI-M card (in the I/O cage) is connected to four slots. The four slots driven by an STI-M card are its *domain*.
- Each book has 12 STI links. An I/O cage can have up to 7 STI connections (which means 7 domains). Many connection possibilities exist and some of the limitations are obvious. For example, a z990 model A08 cannot drive two fully-loaded I/O cages because it has only 12 STIs and the two cages need 14 STI connections.
- A z990 model D32 has 48 STI links. Three full I/O cages (the maximum possible) use 21 STI links.
- Directly connected ICB-4 links bypass the I/O cage and occupy a book STI connection.
- STI-2 and STI-3 adapters are used for ICB-2 and ICB-3 connections.
- Non-I/O adapters that occupy a slot in an I/O cage (such as a cryptographic adapter) are considered I/O adapters for this discussion.

Each possible “physical” CHPID port/connector has a PCHID number that is fixed by the physical location of the adapter and connector. An IOCDS associates a CHPID number to a PCHID number. An adapter in an I/O cage slot can provide a maximum of 16 CHPIDs. (The ESCON adapter does this, although only 15 can be used at one time.) There are a maximum of three I/O cages.[5] Each cage has a maximum of 28 slots. PCHID numbers are assigned as follows:

	PCHID numbers		
I/O cage slot	I/O Cage 1	I/O Cage 2	I/O Cage 3

[4] Initially, the IBM configurator will not allow more than 512 channels to be ordered.
[5] The numbering of the I/O cages is shown in Figure 2-2 on page 10.

```
1                100 - 10F       300 - 30F       500 - 50F
2                110 - 11F       310 - 31F       510 - 51F
3                120 - 12F       320 - 32F       520 - 52F
4                130 - 13F       330 - 33F       530 - 53F
6                140 - 14F       340 - 34F       540 - 54F
7                150 - 15F       350 - 35F       550 - 55F
8                160 - 16F       360 - 36F       560 - 56F
9                170 - 17F       370 - 37F       570 - 57F
10               180 - 18F       380 - 38F       580 - 58F
11               190 - 19F       390 - 39F       590 - 59F
12               1A0 - 1AF       3A0 - 3AF       5A0 - 5AF
13               1B0 - 1BF       3B0 - 3BF       5B0 - 5BF
15               1C0 - 1CF       3C0 - 3CF       5C0 - 5CF
16               1D0 - 1DF       3D0 - 3DF       5D0 - 5DF
17               1E0 - 1EF       3E0 - 3EF       5E0 - 5EF
18               1F0 - 1FF       3F0 - 3FF       5F0 - 5FF
19               200 - 20F       400 - 40F       600 - 60F
20               210 - 21F       410 - 41F       610 - 61F
21               220 - 22F       420 - 42F       620 - 62F
22               230 - 23F       430 - 43F       630 - 63F
24               240 - 24F       440 - 44F       640 - 64F
25               250 - 25F       450 - 45F       650 - 65F
26               260 - 26F       460 - 46F       660 - 66F
27               270 - 27F       470 - 47F       670 - 67F
29               280 - 28F       480 - 48F       680 - 68F
30               290 - 29F       490 - 49F       690 - 69F
31               2A0 - 2AF       4A0 - 4AF       6A0 - 6AF
32               2B0 - 2BF       4B0 - 4BF       6B0 - 6BF

PCHID numbers used for direct book connections: 000-0FF
```

Note that slot numbers 5, 14, 23, and 28 are missing. These slots are used for STI connectors.

Consider an ESCON adapter in slot 15 of I/O cage 1. PCHID numbers 1C0 - 1CF are reserved for this slot. The first ESCON connector on the adapter (whether or not that particular port is enabled or used) is PCHID 1C0, the second connector is 1C1, and so forth. The PCHID numbers are fixed. The CHPID numbers are not fixed and can be arbitrarily assigned when an IOCDS is constructed. For example, an installation might decide to make PCHID 1C0 (the first ESCON channel in our example) CHPID 52 in LCSS 1.

I/O connections to books

Direct book connections (to STI ports) are used for ICB-4 channels. (These channels are used for connections to Coupling Facilities or other z990s.) There are 12 STI ports on a book, and a PCHID number is assigned to each one, as follows:

```
book        PCHIDs
 0        010 - 01B
 1        020 - 02B
 2        030 - 03B
 3        000 - 00B
```

These PCHIDs and connections do not involve I/O cages and are used for ICB-4 connections.

2.9 System control

Figure 2-7 on page 20 provides a conceptual overview of the system control design. While the details have changed, the general structure for system control is similar to the z900.

Various system elements contain *Flexible Support Processors* (FSPs).[6] A Flexible Support Processor is a card based on the IBM Power PC microprocessor. An FSP connects to an internal Ethernet LAN (to communicate with the Support Elements) and provides a SubSystem Interface (SSI) for controlling components.[7] The SSI includes several UARTs, digital I/O lines, and a number of unique sensing and control lines and protocols.

A typical FSP operation is to control a power supply (shown as DCAs in the figure). A Support Element might send a command to the FSP to bring up the power supply. The FSP (using the SSI connections) would cycle the various components of the power supply, monitor the success of each step, monitor the resulting voltages, and report this status to the Support Element.

A z990 has more FSPs than indicated in the figure and the interconnections are more complex, but Figure 2-7 illustrates the general concepts involved. Most system elements are duplexed (for redundancy) and each element has an FSP. There are two internal Ethernet LANs and two Support Elements, again for redundancy. There is a crossover capability between the LANs, so that both Support Element can operate on both LANs.

The Support Elements, in turn, are connected to another (external) LAN (Ethernet or token ring) and the Hardware Management Consoles (HMCs) are connected to this external LAN. There can be one or more HMCs. In a production environment, the system hardware is normally managed from the HMCs. If necessary, the system can be managed from either Support Element. Several or all HMCs can be disconnected without affecting system operation.

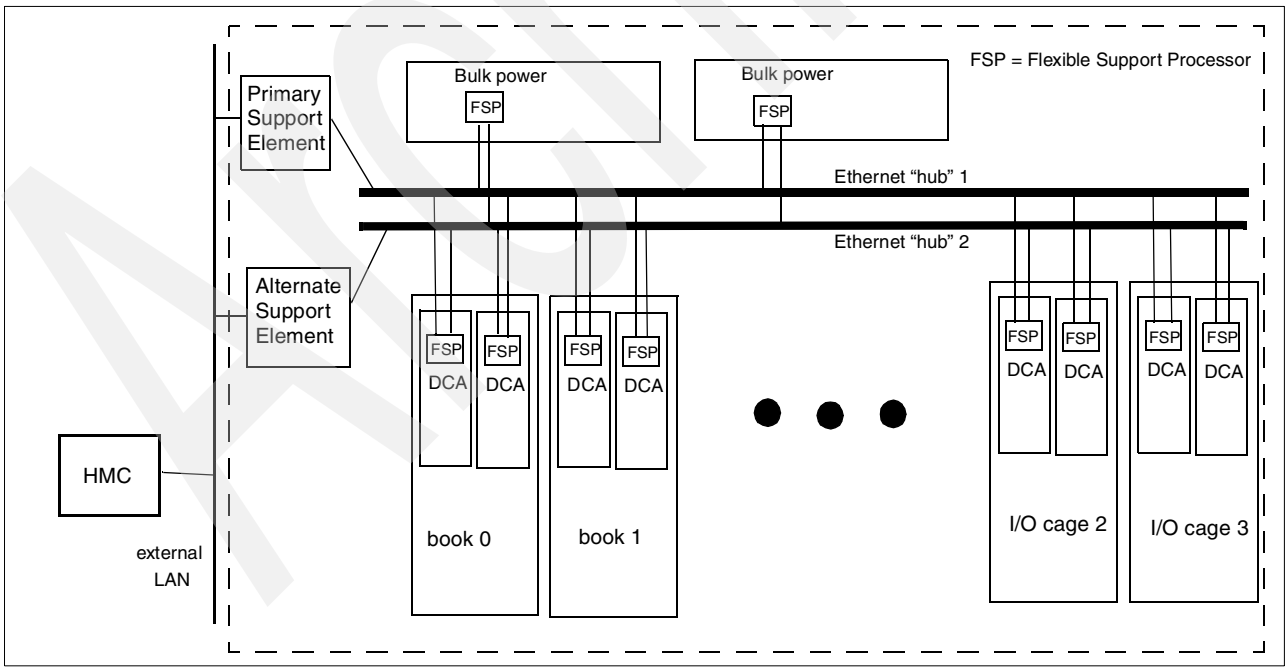


Figure 2-7 Conceptual overview of system control elements

[6] These were informally known as *cage controllers* in earlier systems.
[7] More detailed descriptions may refer to SSI-M and SSI-S notations, corresponding to master and slave functions.

The operation of the internal LANs and the FSPs is checked by a Network Heart Beat (NHB) function that uses complementary programs in the Support Elements and the FSPs to verify correct operation and to trigger alternative reassignments when necessary.

2.10 Power and cooling

A z990 system uses three-phase 200-480 VAC power. A small A08 system uses approximately 6.8 KW (if all I/O adapter slots are filled), while a very large D32 system uses up to 21.4 KW. Two power feed cables are connected to two internal bulk power units that provide power for the system. The system can continue operation after a failure of one of these units. They are designed such that the system will continue to operate after a failure of one phase of the incoming power supply.

Internal functions adjust the power factor to almost 1.0, with little harmonic content. (This complies with increasing demands, sometimes in the form of binding requirements, for *building-friendly* power supplies.)

Incoming power is converted to high voltage DC and this is distributed to the subsidiary power units that convert it to the various voltages need by the controllers, cages, books, and I/O adapters. Very large currents are needed at low voltages and it is not practical to directly distribute the low voltage supplies throughout the system.

As a overview statement, everything in the power system is duplexed. Each book (up to four) exists on its own power boundary, meaning that it is possible to remove power on a book without affecting other books.[8] Two additional power supply units are installed in the processor cage as part of an upgrade to the next higher model.

The processor cage (containing the books) uses 1.2, 1.5, 2.5, and 3.4 volt supplies. (The 1.2 volt requirement is up to 700 amperes!) The I/O cages use 1.8, 2.5, 3.3, 3.4, 5.0, and 24 volt supplies.

Units known as DCAs convert the distributed high voltage to various low voltages. Slightly different DCAs are used in the processor cage and the I/O cages. Up to eight DCAs are used for the processor cage (two for each book), and each I/O cage contains two DCAs. Each DCA has a Flexible Support Processor that is part of the system control function discussed in “System control” on page 20.

2.10.1 Cooling

The z990 uses refrigeration cooling for the MCMs in the books. Everything else is air-cooled.

The refrigeration package is a Modular Cooling Unit (MCU). A single MCU has two cooling loops and cools the MCMs in one or two books. A z990 model B16 or larger will have two MCUs. The MCUs are not redundant. If an MCU fails such that MCM temperatures rise above specified levels, the system cycle time is automatically slowed down. (The cycle time is the same for all books, and a slowdown affects all books.) MCMs produce less heat when run slower, and an integrated backup air-cooling mechanism (with fans) is sufficient at the slower speed.

System slowdown (after an MCM failure) is in steps, depending on the temperatures in the books. The maximum slowdown is 8%.

[8] There are many implications to removing a book. This discussion is only about the power aspects.

The two cooling loops in an MCU are controlled independently. For example, a book connected to one cooling loop can be removed or added without affecting the operation of another book being cooled by the other loop.[9]

The other components of the system (I/O cages, power supplies, memory in the books, and so forth) are air-cooled. A sophisticated arrangement of fans (Air Movement Devices, or AMDs) and sensors provides the operation. Fan speeds are variable and are controlled to maintain the required cooling.

The MCU refrigeration units are air-cooled; that is, their heat sink is into the general airflow of the z990 system. The total heat produced by the system ranges from approximately 23,000 BTUs (small system, 6.8 KW) to 73,000 BTUs (21.4 KW). This heat load must be managed by the raised-floor air system.

2.11 Concurrent changes

A typical z990 may replace several older systems. One consequence of this is that any disruption of z990 service will probably affect more applications than on previous systems. You might view repair and configuration changes as having several levels:

- A *concurrent* repair or configuration change takes place while the system is running and the affected operating systems are running. That is, IPLs or a Power-on Reset (POR) are not required.
- A slightly more complex change, without POR or IPLs, would involve taking a CHPID offline and back online to activate a change to it.
- A more disruptive change might require IPLing one or more operating systems, but without a POR. This implies that some operating systems (in their own LPARs) are not affected.
- A POR affects the complete system for a relatively short time and requires all LPARs to be IPLed again. (These IPL procedures will take varying amounts of time, depending on the complexity of the environments being started.)
- A change or repair that requires *power off* affects the complete system, of course, and implies more time than a POR.

The z990 has many internal, automatic, self-repairing functions. These are usually concurrent functions and may not be visible to any operating system. Many system upgrades and repairs are concurrent activities. Over a number of years, IBM has steadily increased the number of repair actions and configuration changes that can be done concurrently, and this direction is continued with the z990.

2.12 Instruction set

The z990 includes many new and changed instructions. These might be grouped as long-displacement instructions, cryptographic instructions, and other instructions. All of the new and changed instructions are listed in “New and changed instructions” on page 125.

Long-displacement instructions

The concept of a long displacement has been introduced. This is a *signed* 20-bit displacement value, instead of the unsigned 12-bit displacement used in most existing instructions. There are two groups of long-displacement instructions:

[9] Again, this comment refers only to the cooling function. Removing or adding a book involves many other considerations.

- New instructions. There are 44 of these.
- Existing instructions that are changed to include long displacements. There are 69 of these.

The *new* instructions are available only when operating in zArch mode; they are not available in ESA/390 mode. The *modified* instructions operate with a long displacements only in zArch mode. In ESA/390 mode, they operate as they did previously.

All of the new (and modified) instructions are 6 bytes long. The general formats are shown in Figure 2-8. Existing instructions in RXE and RSE format have an 8-bit field that is unused. These instructions are now considered to be in RXY and RSY formats, and the unused field (now the DH field) is concatenated to the left of the 12-bit D field (now the DL field), producing a 20-bit displacement.[10] This is considered to be a signed value and will be added or subtracted from the value in the base register (plus index register, if used). A number of new instructions in the RXY and RSY format have also been added.

While these instructions are available to any assembly language programmer, their design was based primarily on the needs of the C and PL/X compilers, DB2®, and Java functions.

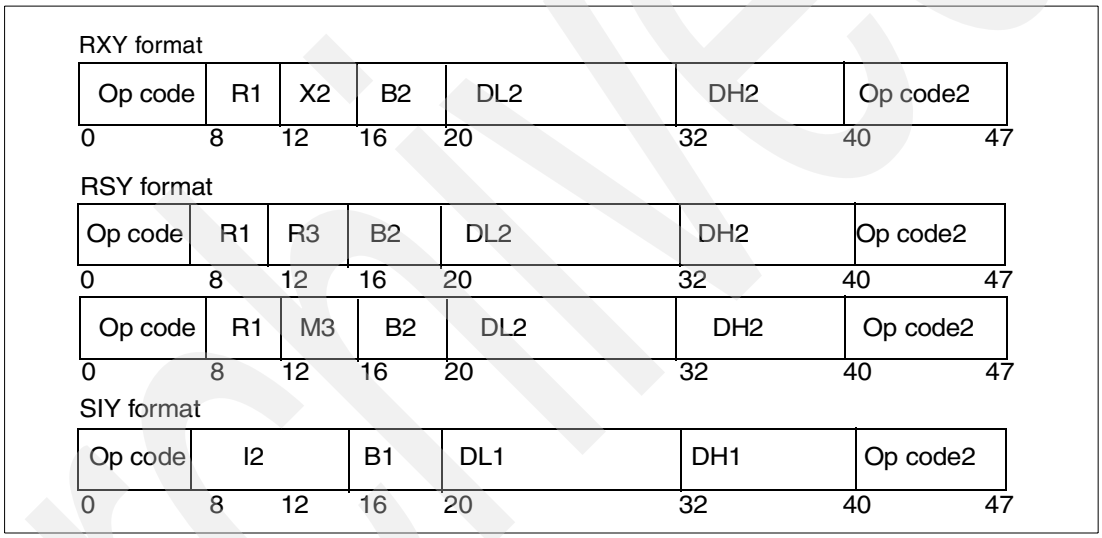


Figure 2-8 Long displacement formats

An attempt to execute any of these instructions on a system that does not have the long-displacement facility installed will produce an operation exception (interruption).

Do not confuse long displacement instructions with relative addressing, as used in the *relative and immediate* instructions added to S/390 architecture a few years ago. An instruction using relative addressing for an operand requires no base register for that operand, while a long displacement instruction needs a base register. However, programs using long displacement instructions may need fewer base registers. Compilers can use these instructions to better optimize the use of registers (and assembly language programmers can do the same thing, of course).

The long displacement instructions will also be made available on z800 and z900 systems.

[10] The naming of these fields is generally DH (for Displacement High order) and DL (for Displacement Low order).

Cryptographic assist instructions

Five new instructions are included with the cryptographic assist function that is standard on every z990 PU. As a group, these instructions are known as the *Message Security Assist* (MSA). Briefly, the instructions are:

```
Cipher message
Cipher message with chaining
Compute intermediate message digest
Compute last message digest
Compute message authentication code
```

These are all problem-mode instructions and are all in RRE format. The cryptographic characteristics of the z990 are discussed in “Sysplex coupling considerations” on page 49.

Other instructions

The Store Facility List (STFL) instruction results have been expanded to include:

```
Bit 17: The Message Security Assist is installed.
Bit 18: The long-displacement facility is installed (in zArch mode)
```

The STFL instruction is a privileged instruction, normally used only by the operating system.

2.13 Additional hardware elements

There are other new or changed hardware elements that characterize the z990.

2.13.1 HSA

HSA is a memory area used by the system, especially by the I/O subsystem. HSA memory is taken from the real memory installed in book 0. The size of HSA memory is variable and has many factors. The largest factor is the number of subchannels used in each LCSS. An IOCDS definition (via IOCP code or HCD parameters) can specify the maximum number of subchannels per LCSS, and this number can range up to approximately 63 K. Changing these numbers (via the IOCDS) requires a Power-on Reset.

In an extreme case, where 63 K subchannels are allowed for each of two LCSS, the HSA size could be approximately 500 MB. Few installations need this number of subchannels, and a number 25% of this size might be considered more typical. You need to work with your marketing representative and your specific configuration requirements to obtain a valid HSA estimate.

Earlier systems allowed a *dynamic expansion factor* to be specified for HSA. This no longer exists and has been replaced by the ability to specify (in an IOCP or HCD) the maximum number of devices that can be used in an LCSS.

2.13.2 Internal battery feature

The Internal Battery Feature is optional on the z990. In the event of an interruption to the input power, the internal battery will provide sustained system operation for a short period of time. The duration of the battery support is highly dependent upon the system model and I/O cages installed. Typically, this would be a minimum of 7.5 minutes for a maximum z990 configuration and up to 13 minutes for a B16 model with a single I/O cage.

Most power interruptions are due to the temporary loss of a single phase on the three-phase input line. The z990 can tolerate loss of a single phase even with no battery backup. This

capability, coupled with the Internal Battery Feature and full operation from either of the dual independent line cords, gives a very high degree of resilience to transient power dropouts. When coupled with a UPS capability, the internal battery ensures that there is no loss of power during startup of the emergency supply.

Batteries are installed in pairs; a single ordered feature code (fc3210) provides two batteries. The pair consists of a front and rear battery in one of the three IBF slots. The Z frame has two IBF slots (at the top of the frame) and the A frame has a single IBF slot (at the top).

2.13.3 Support Elements

Two identical Support Elements (SEs) are standard and operate under OS/2®. The first is the primary SE, and the second is a backup. The backup can be used to preload SE code while the primary is active. The Support Elements are IBM ThinkPads with 256 MB memory. Two options are available, allowing choices of token ring and/or Ethernet operation. This is described in more detail in “SE and HMC connectivity” on page 41.

If a token ring adapter is included with the Support Elements, then a Multistation Access Unit (MAU) is included in the system frame. If the system has only Ethernet adapters, an Ethernet switch is included with the system.

2.13.4 Hardware Management Console

A Hardware Management Console (HMC) must be used with the z990. An existing HMC can be used, provided it is at least at the level of feature code 0073. New HMCs ordered with a z990 are based on IBM 8305-NQU Netfinity systems and have 512 MB memory, a DVD drive, token ring adapter, Ethernet, and use OS/2 as the operating system.

2.14 External time reference

z990 processors have 128-bit TOD clocks. This provides the capability to:

- Host dates beyond 2041 (64-bit TOD has this limit).
- Create sysplex-unique clock values.
- Provide more granular timestamps. (zArchitecture specifies that every clock query will be provided with a unique TOD value. The z990 processor speed exceeds the granularity of the 64-bit clock. If a user issues two Store Clock instructions in quick succession, the CP would have to spin, waiting for the clock to be incremented before it could return the value for the second request.)

Sysplex Timer® attachment

The 9037 Sysplex Timer provides the synchronization for the Time-of-Day (TOD) clocks of multiple CPCs, and thereby allows events started by different CPCs to be properly sequenced in time. When multiple CPCs update the same database and database reconstruction is necessary, all updates are required to be timestamped in proper sequence. The Message Time Ordering function may require connectivity to an ETR.

The z990 External Time Reference (ETR) cards provide the interfaces to the IBM Sysplex Timer. ETR cards are optional on z990 processors; if the ETF feature is ordered, a pair of cards will be installed. The External Time Reference function is part of the processor in the zSeries z990 processors. There is an ESCON MT-RJ port on each ETR card for the cable connection to a Sysplex Timer. Each ETR card should connect to a different 9037 Sysplex Timer in an Expanded Availability configuration.

The two ETR cards are optional on the z990 processors; they need to be ordered. They will automatically be included in an order if the configuration you are preparing contains a CF link.

The ETR network ID is entered on a panel associated with the System Complex Timer task. This task is grouped with the CPC Configuration tasks on the SE/HMC. If ETR is installed and this task invoked, a notebook is displayed with two panels. One contains configuration information and the second contains status information. The ETR network ID is in the configuration panel. (If ETR is not installed, a message box is displayed instead.)

The ETR ID is initialized to X'FF'. You can enter a value in the range 0-31 in the panel. When the configured ETR network ID does not match any of the attached 9037(s) then both ports enter a semi-operational state. This means they are disabled for stepping, but are still receiving data from the 9037.

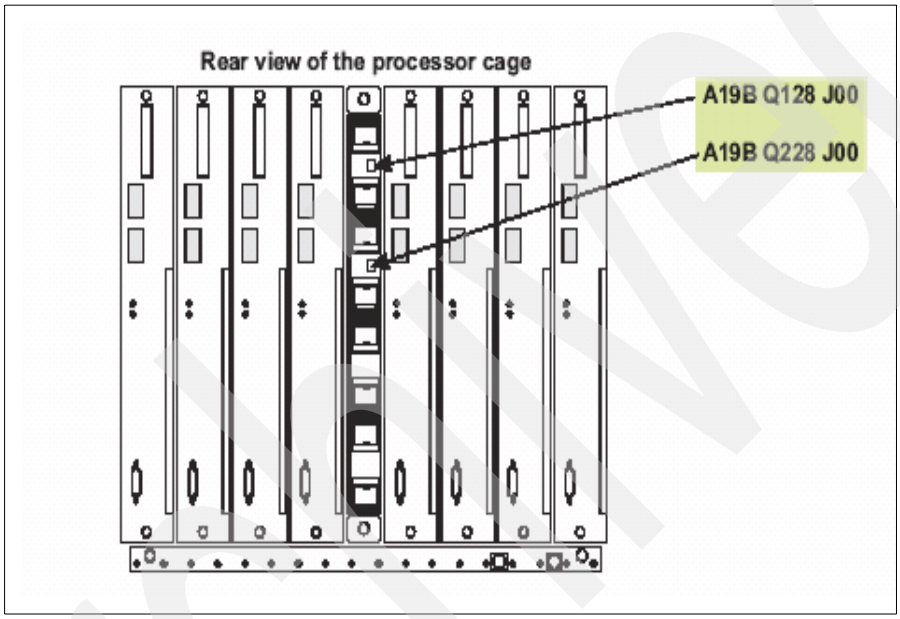


Figure 2-9 Sysplex timer connections - ETR cards

Sysplex Timer synchronization

As processors and Coupling Facility link technologies have improved over the years, the synchronization tolerance between operating systems in a Parallel Sysplex has become more rigorous. In order to ensure that any exchanges of timestamped information between operating systems in a sysplex involving the Coupling Facility observe the correct time ordering, timestamps are now included in the message-transfer protocol between the server operating systems and the Coupling Facility (refer to "Internal Coupling Facility (ICF)" on page 57 for more information on Message Time Ordering function).

Connectivity

The ETR cards are located in the CPC cage of the z990 processors as shown on Figure 2-9. Each ETR card has a single port supporting an MT-RJ fiber optic connector to provide the capability to attach to a 9037 Sysplex Timer Unit. The MT-RJ is an industry standard connector which has a much smaller profile compared with the original ESCON Duplex connector supported on the ETR ports of G5/G6 and earlier servers. The 9037 Sysplex Timer Unit has an optical transceiver that supports an ESCON Duplex connector.

An MT-RJ/ESCON Conversion Kit supplies two 62.5 micron multimode conversion cables. The conversion cable is two meters (6.5 feet) in length and is terminated at one end with an

MT-RJ connector and at the opposite end with an ESCON Duplex receptacle to attach to the under floor cabling. The same conversion kit is used on the z990 processors with the 16-port ESCON feature or ETR when reusing existing 62.5 micron multimode fiber optic cables terminated with ESCON Duplex connectors.

Note: The ETR card does not support a multimode fiber optic cable terminated with an ESCON Duplex connector. However, 62.5 micron multimode ESCON Duplex jumper cables can be reused to connect to the ETR card. This is done by installing an MT-RJ/ESCON Conversion kit between the ETR card MT-RJ port and the ESCON Duplex jumper cable.

Ordering ETR cables

Two ESCON MT-RJ cables for the ETR feature on the zSeries z990 system will automatically be added to any zSeries Fiber Cabling Service contract offered by IBM Global Services. Contact your local IBM Installation Planning Representative, IBM zSeries z990 Product Specialist, or IBM Connectivity Services Specialist for details. If you choose not to use this Service, you may purchase the two ESCON MT-RJ cables separately, or provide them yourself from another source. An ESCON MTRJ-to-ESCON Duplex conversion kit will also have to be provided to connect the MT-RJ cables to the Duplex connectors on the 9037.

2.15 Basic zSeries comparisons

Table 2-2 provides some basic comparisons between various zSeries models. The z/900 models used are the larger (20 PU) models. As always, a table such as this provides only a high-level overview. Many improvements and characteristics are not reflected in the table.

Table 2-2 Comparison of zSeries models

	z800	z900	z900 Turbo	z990-A08	z990-C32
Total Processor Units	5	20	20	12	48
Maximum customer PUs	4	16	16	8	32
Maximum number of CPUs	4	16	16	8	32
Maximum number of IFLs	4			8	32
Standard SAPs	1	3	3	2	8
Standard spare PUs	unused PUs	1	1	2	8
Cycle time	1.6 ns	1.3 ns	1.09 ns	.83 ns	.83 ns
STIs	6 (1.0 GB/s)	24 (1.0 GB/s)	24 (1/0 GB/s)	12 (2.0 GB/s)	48 (2.0 GB/s)
Memory	8 - 32 GB	10 - 64 GB		8 - 64 GB	8 - 256 GB
Maximum MSUs	108	441			
Maximum I/O cages	1	3	3	3	3
Maximum usable CHPIDs	256	256	256	512[a]	1024[b]
Support Elements	2	2	2	2	2
External power	1 phase	3 phase	3 phase	3 phase	3 phase
L1 cache (per PU)[c]	256K/256K	256K/256K	256K/256K	256K/256K	256K/256K

	z800	z900	z900 Turbo	z990-A08	z990-C32
L2 cache	8 MB			32 MB	32 MB
Battery feature	no	optional	optional	optional	optional
Parallel channels	no	yes	yes	no	no
Maximum LPARs	15	15	15	30 (60)[d]	30 (60)
Basic mode (no LPARs)	yes	yes	yes	no	no
Multiple channel subsystems	no	no	no	yes	yes
Crypto coprocessors (CCF)	0 or 2	0 or 2	0 or 2	assist on every PU	assist on every PU
Crypto adapters	PCICC PCICA	PCICC PCICA	PCICC PCICA	PCICA PCIXCC	PCICA PCIXCC
HiperSocket CHPIDs (Max)	4	4	4	16	16
Long displacement instructions	yes	yes	yes	yes	yes

a. The maximum number of usable CHPIDs depends on the number of LCSSs in use. A maximum of 512 usable CHPIDs assumes 2 LCSSs with no spanned channels.
b. This number assumes 4 LCSSs with no spanned channels. (Use of four LCSSs is only a statement of direction for IBM.)
c. These numbers refer to the I cache and the D cache.
d. 60 LPARs is a statement of direction for IBM.

3

Discussion topics

This chapter provides further discussion of a number of interesting or important areas concerning z990 systems, especially as they compare with z900 systems. A short introduction is included for some of the topics for readers not familiar with zSeries developments; the topics are presented in no special order.

3.1 Performance factors

There are several factors that are responsible for the performance characteristics of the z990 processors. The most obvious factor is that the PU has been designed and optimized to meet the demands of new workloads, while at the same time maintaining excellent performance on traditional S/390 workloads.

Each PU utilizes a superscalar design which provides all of the following:

- Decoding of 2 instructions per cycle
- Execution of 3 instructions per cycle (oldest must be a branch)
- In-order execution
- Out-of-order operand fetching

The z990 *book* design, the centrally integrated crosspoint switch, and a large shared L2 cache that is logically and physically spread across the four books enables the movement of data from source to destination as efficiently as possible. Each book has twelve 2.0 GB/second STI links, distributed between three I/O interface adapters (MBAs) that form the base for high bandwidth I/O operations.

Other processor design features that contribute to the enhanced performance of z990 processors are:

- The base clock rate is .83 nanoseconds, compared to 1.3 nanoseconds in the base z900 machines.
- Floating-point performance is optimized for IEEE arithmetic (exploited by Java applications). Floating-point operations have now been fully pipelined at five stages for both IEEE (binary) and IBM floating point (hex) arithmetic.

- A programmable Dynamic Address Translation (DAT) translator is used. This mechanism allows for later updates to the address translation algorithm (mapping virtual addresses to physical memory locations). A high-performance DAT function is complex and this design allows for corrections and design changes to the current algorithm, if necessary, as well as making the translator extendable for future architectural changes.
- A Second Level Translation Lookaside Buffer (TLB) is used. This mechanism provides a secondary cache for the DAT for both the instruction and data caches. The secondary buffers provide an additional 4 K of dynamic address mapping over the current 512 entries available.
- Several on-chip cryptographic assist instructions have been added. This hardware implementation for symmetric encryption/decryption is used to speed up SSL transactions, VPN data transfer, and data repository applications.
- An asynchronous processor-to-SCE interface has been added. Its purpose is to decouple the processor and SCE cycle times, allowing each to run at its optimum speed and not forced into fixed multiples of one another. This interface serves to buffer the exchange of data between the two disparate cycle times.
- A design for asymmetric mirroring for error detection was implemented. This provides a mechanism by which the mirrored execution used for error detection occurs 1 cycle delayed from actual operation. This allows the operation execution logic and circuitry to be optimized for performance.

Collectively, these design changes mean that a z990 processor, with typical workloads, is considerably faster than a z900 processor at the same clock speed. The clock speed in the z990 is faster, of course, but the clock speed ratios in this case are not a valid indicator of performance differences.

Figure 3-1 on page 31 shows the performance comparison between the latest S/390 and zSeries processors.

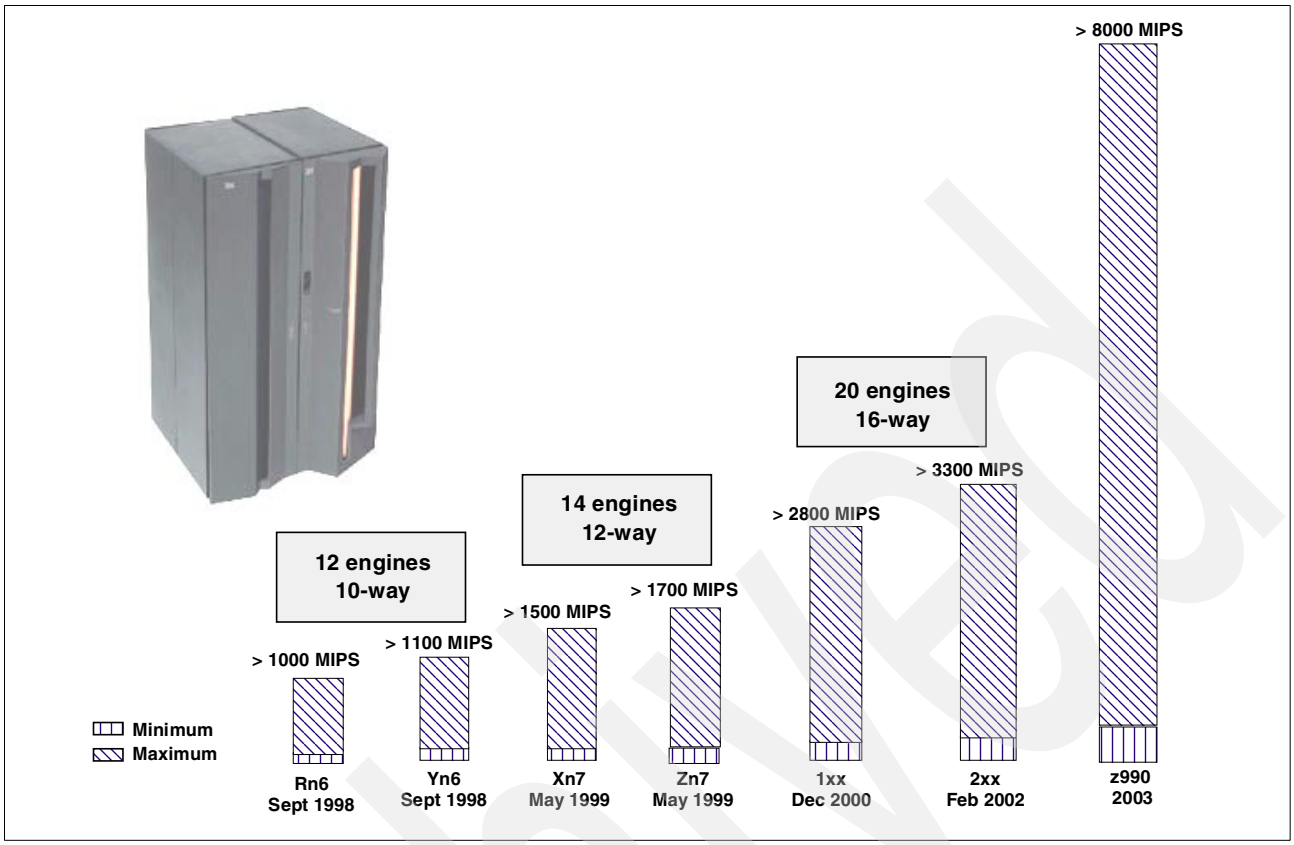


Figure 3-1 Performance characteristics of IBM processors

An initial performance comparison with z900 systems can be summarized as follows:

z990 system	compared to	performance factor
1 processor	z900 2C1	1.52 - 1.58
8-way	z900 2C8	1.48 - 1.55
16-way	z900 216	1.45 - 1.53
32-way	z900 216	2.4 - 2.9

These numbers are based on very early experience with existing application code. At this point the new instructions were not being used and instruction sequences had not been optimized for this machine. You should consult up-to-date IBM sources for more accurate performance information.

PR/SM™ performance factors

The operation of the internal PR/SM firmware has many effects on performance. New factors for the z990 include:

- The system attempts to position all the real memory for an LPAR within one book. This is not always possible. Portions of real memory are not lost due to this attempt.
- The system attempts to dispatch a CP (for an LPAR) that is in the same book as the real memory for that LPAR. If a CP from that book is not free, a CP from another book is used. A potential LPAR CP dispatch is not delayed waiting for a free CP in the target book.
- An attempt is made to redispatch a logical CP to the same physical CP (even if there has been an intermediate user of that CP). This may allow some of the cache or TLB data to be reused.[1] However, if that CP is not available, another CP will be used.

[1] Minor changes to the z/OS dispatcher were needed for this function.

The effect of these functions is minor and is typically in the noise level of performance measurements when practical workloads are used. In practice, LPAR real memory is typically split across books and CPs are dispatched from any book. The L1 and L2 cache mechanism is so effective that the location of real memory (in the local book or in another book) is usually insignificant.

3.2 Channel subsystem

One of the most striking architectural changes associated with the z990 is the set of extensions to the channel subsystem. Perhaps the best way to start a description is to explain *why* these extensions were introduced. The purpose is simple. It is to permit a z990 system to have more than 256 channels. Larger installations have encountered the 256 channel limitation. The maximum number of channels has been a concern for some time, although the move to FICON channels and devices has mitigated the limitation.

The difficulty is to extend this number while maintaining compatibility with existing software. The 256 maximum CHPID number is reflected in various control blocks in operating systems, software performance measurement tools, and even some application code. Simply changing the control blocks is not a viable option since this would break too much existing code. The solution provided by the z990 is in the form of multiple Logical Channel Subsystems (LCSSs). These are implemented in a manner that has little or no impact on existing code.

It is important to note that there is no unique hardware (or feature codes) associated with LCSSs. These are not priced features. These architected functions are implemented in the firmware of the system and are, potentially, exploited by operating systems running in the system.

3.2.1 Logical Channel Subsystems (LCSSs)

Existing code works with single-byte CHPID numbers, producing the limitation of 256 CHPIDs. The new architecture provides multiple sets of channel definitions, each with a maximum of 256 channels. Existing operating systems would be associated with one Logical Channel Subsystem (LCSS) and work with a maximum of 256 CHPIDs. Different LPARs can be associated with different Logical Channel Subsystem definitions. Thus a single operating system instance (using existing code) still has a maximum of 256 CHPIDs, but the system as a whole can have more than 256 CHPIDs. It is necessary to have multiple operating images (in multiple LPARs) to exploit more than 256 channels, but this is a common mode of operation.

In a sense, LCSSs virtualize CHPID numbers. A CHPID no longer directly corresponds to a hardware channel, and CHPID numbers may be arbitrarily assigned. A hardware channel is now identified by a PCHID, or *physical* channel identifier. A PCHID number is defined for each potential channel interface. An I/O adapter card has up to 16 channel interfaces[2] and 16 PCHID numbers are reserved for each I/O adapter slot in each I/O cage. Not all I/O adapters provide 16 channels, of course, but 16 PCHID numbers are allocated to each I/O slot. The PCHID numbers allocated to each I/O adapter and port on that adapter are fixed[3] and cannot be changed by the user.

The z990, as announced, can use two LCSSs. The architecture introduced with the z990 permits more LCSSs. The z/OS versions available with initial z990 shipments can use only the first LCSS. The z/VM version available August 15, 2003 can use either LCSS. More

[2] No existing I/O adapter provides 16 usable interfaces (channels). An ESCON adapter has 16 interfaces but only 15 may be used; the 16th is a spare.
[3] There is a minor exception for ESCON adapters when a spare port replaces a failing port.

complete exploitation of the new architecture is expected to be delivered over time, in evolutionary steps, by the operating systems and various software subsystems.

A given LPAR is associated with a single LCSS, and a single LCSS has a maximum of 256 CHPIDs. Multiple LPARs may be associated with a given LCSS. Using two LCSSs (as announced for the z990) means that up to 512 channels can be used. This number is reduced by *spanned* channels and these are discussed in “Spanned channels” on page 35.

Figure 3-2 illustrates the relationship between LPARs, LCSSs, CHPIDs, and PCHIDs. This is an idealized illustration and ignores such complexities as spanned channels, dynamic I/O changes, and so forth. This illustration also includes the I/O devices and switches used in a later IOCP example.

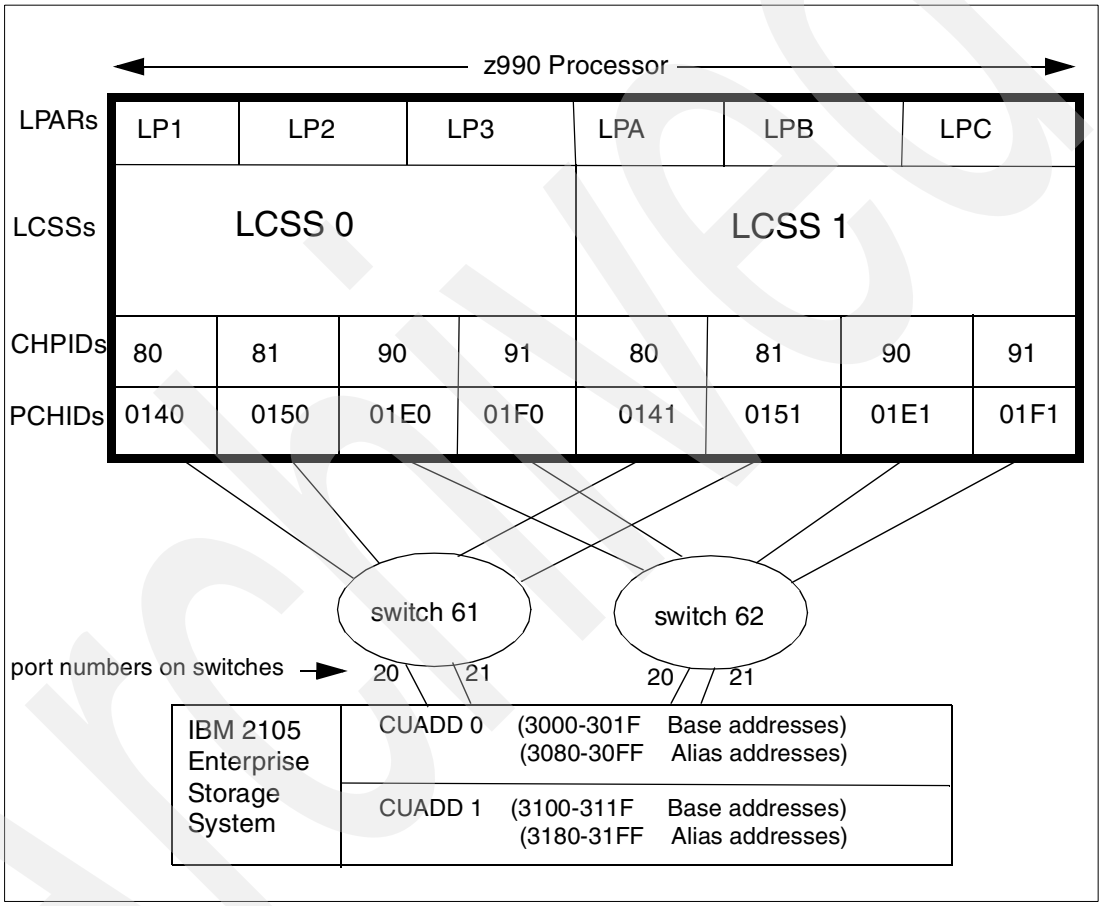


Figure 3-2 Overview of LPARs, LCSSs, CHPIDs, and PCHIDs

This figure illustrates a number of important points, such as:

- Two LCSSs are shown, which is the maximum announced with the z990.
- An LPAR is associated with a specific LCSS. (Also note that LPARs have unique names across the complete system. LPAR naming and numbering has become a little more complex and this is discussed in “LPARs” on page 66.)
- Multiple LPARs (up to 15) may be associated with an LCSS.
- A CHPID is associated with a specific LCSS. CHPID numbers are unique within that LCSS, but may be reused in other LCSSs. (For example, there is a CHPID 80 in both LCSSs.) A CHPID number is arbitrarily selected. For example, we could change CHPID 80 (in either or both LCSSs in the illustration) to C3 simply by changing a value in the IOCDS.
- A CHPID is associated with a PCHID, and PCHID numbers are unique across the system.

- Different channels on a single I/O adapter can be used by different LPARs. In the illustration, PCHID 0140 is the first channel on the adapter in I/O cage 1, slot 6. PCHID 0141 is the second channel on the same adapter. See “I/O interfaces and identification” on page 18 for a discussion of PCHID values and I/O cage slots.

3.2.2 IOCP example

LCSS and PCHID information is included in an IOCDS. An IOCDS is created through the HCD utility program or by use of a standalone IOCP file. In practical use, HCD is normally used. For illustration purposes we will examine an IOCP file and will assume the reader is generally familiar with such files.

The following IOCP definition describes the system shown in Figure 3-2 on page 33.[4] This is not intended to represent a practical system—it has no consoles, for example—but it illustrates the new z990 elements.

```
ID MSG1='BILLIOCP',MSG2='z990 2 LCSS',SYSTEM=(2084,1)
RESOURCE PARTITION=((CSS(0),(LP1,1),(LP2,2),LP3,3)),                X
  (CSS(1),(LPA,1),(LPB,2),LPC,3))),                                  X
  MAXDEV=((CSS(0),64512),(CSS(1),64512))
CHPID PATH=(CSS(0),80),SHARED,PARTITION=((LP1,LP2,LP3),(=)),         X
  SWITCH=61,TYPE=FC,PCHID=0140
CHPID PATH=(CSS(0),81),SHARED,PARTITION=((LP1,LP2,LP3),(=)),         X
  SWITCH=61,TYPE=FC,PCHID=0150
CHPID PATH=(CSS(0),90),SHARED,PARTITION=((LP1,LP2,LP3),(=)),         X
  SWITCH=62,TYPE=FC,PCHID=01E0
CHPID PATH=(CSS(0),91),SHARED,PARTITION=((LP1,LP2,LP3),(=)),         X
  SWITCH=62,TYPE=FC,PCHID=01F0
CHPID PATH=(CSS(1),80),SHARED,PARTITION=((LPA,LPB,LPC),(=)),         X
  SWITCH=61,TYPE=FC,PCHID=0141
CHPID PATH=(CSS(1),81),SHARED,PARTITION=((LPA,LPB,LPC),(=)),         X
  SWITCH=61,TYPE=FC,PCHID=0151
CHPID PATH=(CSS(1),90),SHARED,PARTITION=((LPA,LPB,LPC),(=)),         X
  SWITCH=62,TYPE=FC,PCHID=01E1
CHPID PATH=(CSS(1),91),SHARED,PARTITION=((LPA,LPB,LPC),(=)),         X
  SWITCH=62,TYPE=FC,PCHID=01F1
CNTLUNIT CUNUMBR=3000,                                               X
  PATH=((CSS(0),80,81,90,91),(CSS(1),80,81,90,91)),                  X
  UNITADD=((00,256)),CUADD=0,UNIT=2105,                              X
  LINK=((CSS(0),20,21,20,21),(CSS(1),20,21,20,21))
CNTLUNIT CUNUMBR=3100,                                               X
  PATH=((CSS(0),80,81,90,91),(CSS(1),80,81,90,91)),                  X
  UNITADD=((00,256)),CUADD=1,UNIT=2105,                              X
  LINK=((CSS(0),20,21,20,21),(CSS(1),20,21,20,21))
IODEVICE ADDRESS=(3000,032),CUNUMBR=(3000),STADET=Y,UNIT=3390B
IODEVICE ADDRESS=(3080,128),CUNUMBR=(3000),STADET=Y,UNIT=3390A
IODEVICE ADDRESS=(3100,032),CUNUMBR=(3100),STADET=Y,UNIT=3390B
IODEVICE ADDRESS=(3180,128),CUNUMBR=(3100),STADET=Y,UNIT=3390A
END
```

[4] We cheated a little in this listing. Some lines are too long for the required IOCP format and should be broken into continuation lines. Also, we may have gone to a continuation line at an inappropriate point. We did this in the interest of readability. The “X” continuation indicators should be in column 72 of a proper IOCP file. Also, the PCHID number associated with a CHPID is placed on a separate line by the CHPID Mapping Tool and is typically listed that way; however, it can be entered as shown here.

New elements in this IOCP listing include:

- **LCSS definitions**. The LCSSs used in the IOCDS created from this IOCP are stated in the RESOURCE statement. In the example, we define two LCSSs and indicate which LPARs are associated with each one. This *defines* the LCSSs.
- **Subchannel maximums**. The MAXDEV parameter (in the RESOURCE statement) specifies the maximum number of subchannels (device numbers) that can be defined in an LCSS. (The MAXDEV value includes the devices you have defined plus whatever expansion number you need.) This affects the HSA storage used.
- **LCSS number for CHPID**. Each CHPID statement specifies its LCSS number as part of the PATH parameter.
- **PCHID number**. Each CHPID statement *must* include a PCHID parameter to associate the CHPID with a physical channel. (The PCHID parameters can be added to the IOCP definitions by the CHPID Mapping Tool, through HCD definitions, or defined in a standalone IOCP file, but the PCHID parameters must be present in all CHPID statements in order to create an IOCDS.)
- **LCSS number for PATH and LINK parameters**. PATH and LINK parameters in CNTLUNIT statements must indicate the LCSS number associated each path and link.

As can be seen from this example, the basic concepts and definitions for multiple logical channels subsystems are straightforward and mesh with existing IOCP parameters. Equivalent fields have been added to HCD panels for entering LCSS and PCHID numbers.

3.2.3 Spanned channels

A *spanned* channel is connected to more than one LCSS. With the z990 announcement, these channel types can be spanned:

- HiperSockets channels
- IC (CF coupling links) channels

A spanned channel occupies the same CHPID number in all LCSSs in which it is used. For example, if a HiperSocket channel is CHPID 2C in LCSS 0, it must be the same CHPID number in LCSS 1 (if LCSS 1 is used, of course, and if that HiperSocket is also defined in LCSS1). A HiperSocket that connects LPARs in different LCSSs *must* be spanned.

3.2.4 Channel definitions in the IOCP statement

The following channel types (as defined in an IOCDS) are used with z990 systems:

- FICON channel types
  - FC - Native FICON channel (both for native FICON devices and FICON CTCs)
  - FCV - FICON bridge channel
  - FCP - Fibre Channel Protocol (SCSI)
- ESCON channel types
  - CNC - Native ESCON channel
  - CTC - ESCON CTC channel
  - CVC - ES conversion channel, which connects to a converter in block multiplexer mode
  - CBY - ES conversion channel, which connects to a converter in byte multiplexer mode
- CF link channel types
  - CBP - Integrated Coupling Bus (ICB-3) channel, for both OS and CF partitions, to connect z900 or z800 systems
  - CBP - An ICB-4 connection to another z990

  - CFP - InterSystem Coupling (ISC-3) peer mode channel, for both OS partitions and a CF partition
  - CFS - ISC-3 compatibility mode sender channel, for OS partitions[5]
  - CFR - ISC-3 compatibility mode receiver channel, each must be used for only 1 CF partition
  - ICP - Peer mode Internal Coupling (IC-3) channel, for both OS and CF partitions, to connect among LPARs within a z990 system internally
  - CBS - ICB-2 compatibility link to G5/G6 systems
  - CBR - ICB-2 compatibility link to G5/G6 systems
- OSA-Express channel types
  - OSD - OSA-Express (QDIO)
  - OSE - OSA-Express (LCS)
- HiperSockets channel type
  - IQD - HiperSockets channel, QDIO mode only

Each of these channel types requires that a CHPID be defined, even if it is an internal channel and no physical hardware (channel card) exists. Each channel, whether a “real” channel device or a virtual device (such as a HiperSocket) must be assigned a unique CHPID within the LCSS. There are no default CHPID numbers for the z990 and you can arbitrarily assign whatever number you like (within the X’00’ to X’FF’ range, of course).

Most of these channel types can be shared and used concurrently among multiple LPARs within the same LCSS. This capability is known as the Multiple Image Facility (MIF). Exceptions are for ES conversion channels (CVC and CBY) and CF receiver channels (CFR). These channel types cannot be shared concurrently, but can be defined as reconfigurable channels by specifying the REC parameter on the channel definition. The channel can be reassigned to another LPAR after the former owning LPAR configures the channel offline.

There are no parallel channels provided for z990 systems or OSA-2 channels. CHPID types BY, BL, or OSA are not allowed.

3.3 Cryptographic changes

The cryptographic functions that IBM provides is defined by the IBM Common Cryptographic Architecture. These functions, external interfaces, and a set of key management rules which pertain both to the Data Encryption Standard (DES)-based symmetric algorithms and the Public Key Algorithm (PKA) asymmetric algorithms are available through the facilities provided by the z/OS Integrated Cryptographic Service Facility/MVS™ (ICSF). ICSF uses the corresponding hardware features like the Crypto Coprocessor Facility (CCF), the PCI Cryptographic Coprocessor (PCICC) and the PCI Cryptographic Accelerator (PCICA). The z990 does not have CCFs and does not use PCICC adapters.

As mentioned, the CCF feature is not provided with the z990 processors. Some of its functions are initially replaced by the clear key DES, Triple DES (TDES), message authentication code (MAC) message authentication and Security Hash Algorithm (SHA-1) Crypto-assist instructions available to all processors on a z990. Most of the remaining functions will be replaced by the PCIXCC card when it becomes available. The IBM CP assist for cryptographic function architecture (CPACF/CP Assist) provides a set of cryptographic functions that enhance the performance of the encrypt/decrypt functions of Secure Socket Layers (SSL), Virtual Private Network (VPN), and data storing applications not requiring Federal Information Processing Standards (FIPS) 140-2 level 4 security.

[5] When you use z/OS system-managed CF duplexing, one CF partition can additionally share the CF sender channel.

The cryptographic function on the z990 will, therefore, be delivered in two phases. In the first phase, ICSF will use the crypto-assist instructions and the PCICA functions. In the second phase, ICSF will also make use of the functions provided by the PCIXCC card that replaces the PCICC card and most functions provided by CCF.

3.3.1 Initial cryptographic support

The support that ICSF for z/OS initially provides for z990 comprises:

- Clear key DES/TDES/SHA instructions that are available to all processor units (PUs) on a z990 processor. In other words, the z990 PUs are a very fast DES/TDES/SHA engine.
  - Crypto-assisted instructions can be used by programs running in problem state; these instructions are described in the Principle of Operations manual.
  - These DES/TDES functions use clear keys only; they do not use enciphered keys under a master key.
  - The ICSF Common Cryptographic Architecture (CCA) services CSNBSYE and CSNBSYD provide access to these instructions; the path length is minimal and there is no SAF or EXIT support. An assembly language programmer can code the instructions directly, but we suggest using ICSF interfaces for wider compatibility.
  - Key management support is limited to Rivest-Shamir-Adleman algorithm (RSA) Public-Key Cryptography Standards (PKCS) 1.2 key distribution via CSNDPKE/CSNDPKD services; there is no support for the Cryptographic Key Data Set (CKDS).
  - There is no CPU affinity issue with these instructions.
- SSL and IP Security (IPSEC) acceleration is supported:
  - The RSA operations are driven by ICSF to the PCICA cards.
  - The DES operations are driven by ICSF to the z990 CP engines and use the crypto-assisted instructions.
- Applications running on a z990 that use tamper-proof crypto hardware will see the corresponding ICSF services as not available.

The following considerations apply to the cryptographic functions available on z990 processors during the first phase:

- Processors are shipped with a minimum level of cryptographic function (consisting only of SHA-1) resident in CP hardware and always enabled.
- To enable CP assist DES/TDES functions, feature code (FC) 3863 must be ordered.
- There are no prerequisites for ordering and using CP assist. However, CP assist is a prerequisite for ordering and using PCI adapters.

3.3.2 Second phase support

The z990 second phase crypto support by ICSF consists of exploiting the features available in the PCIXCC card that will replace both the PCICC card and CCF functions. The replacement of CCF and PCICC functions by the PCIXCC card will provide much better performance and reliability than today's PCICC and much better scalability than today's CCF due to:

- Use of a Power PC (PPC) 405 processor
- Extensive hardware error checking
- Faster RSA/SHA/DES engines
- More memory and flash memory
- A hardware-assisted communications protocol

The following considerations apply for the second phase:

- PCI cards (PCIXCC and PCICA) are plugged into the I/O cage and are included in any counts of OSA-Express and FICON Express cards.
- PCICA cards are supported for new z990 machines and MES, and may be carried forward on conversions from z900s.
- PCICC cards are not supported and will not be carried forward from z900s.
- The PCIXCC cards, when available, are needed for this phase.
- An IBM Statement of Direction (SOD) says that IBM intends to provide z/VM guest support for the PCIXCC adapter.

3.3.3 Functions not supported

The following functions will no longer be supported by ICSF with a PCIXCC adapter:

- Digital Signature Algorithm (DSA) signature and key generation.
- American National Standard Institute (ANSI) x9.17 services (offset and notarization), and associated key types.
- Ciphertext_translate (CSNBCTT).
- German bank Pool-Pin offset.
- CSFUDK; this support is replaced by CSNBDKG.
- Commercial Data Masking Facility algorithm (CDMF), commonly known as 40-bit encryption.

These functions are thought to be unused and should not impact crypto usage on a z990.

3.3.4 Functions changed and coexistence considerations

The support added to ICSF for double-length Message Authentication Code (MAC) keys has the following implications:

- Current DATAM and DATAMV keys have identical Crypto-Values (CVs) on left and right keys for internal token and CCA-compliant CVs (different) for external tokens. Such internal tokens will continue to work on PCIXCC cards so that existing CKDS entries can be shared with z990 processors.
- Generation/import/export of DATAM and DATAMV on a z990 creates a key with true CCA CVs; that is, different CVs for left and right keys in internal and external tokens. These tokens will work, with restrictions, in PCICC cards with OS/390® V2R10 if APAR OW46382 is applied; they will not work on CCF-only machines.
- z990 handling of keys with Prohibit Export CV bit set *off* has the following migration implications:
  - An external token having a CV with the *prohibit export* bit set off will be imported to an internal token with identical CV; the prohibit export CV bit will not be set.
- If the CKDS is to be shared, then all CKDS management should be performed from a non-z990 processor to avoid possible sharing problems.
- Almost all ICSF application programming interface (API) calls that would execute on CCF, PCICC, or PCICA will be moved over, without change, to PCIXCC, PCICA, or the Crypto-assist instructions available on the z990. There will normally be no need for application reworking, including recompile or re-linked, to move it to a z990.

3.4 Concurrent system upgrades

The z990 processors have the capability of concurrent upgrades, providing additional capacity with no server outage. In most cases, with prior planning and operating system

support, a concurrent upgrade can also be nondisruptive, meaning with no system outage. Model upgrades are assumed to be in sequence; that is, model A08 to B16 to C24 to D32.

The LIC-Configuration Control (LIC-CC) provides for processor upgrade with no hardware changes by enabling the activation of additional installed capacity. Concurrent upgrades via LIC-CC can be done for:

- CPs, IFLs, and ICFs - requires available unused PUs in installed books.
- Memory - requires available capacity on already installed memory cards.
- Channel cards ports (ESCON channels and ISC-3 links) - Requires available ports on channel cards.

I/O configuration upgrades can also be concurrent by installing—nondisruptively—additional channel cards.

Planned upgrades

Planned upgrades can be done by the Capacity Upgrade on Demand (CUoD) or the Customer Initiated Upgrade (CIU) functions.

CUoD and CIU are functions available on z990 processors that enable concurrent and permanent capacity growth of a z990 processor.

CUoD can concurrently add PUs, memory, and channels, up to the limit allowed by the existing configuration. CUoD requires IBM service personnel for the upgrade.

CIU can concurrently add PUs and memory up to the limit of the installed MCM and memory cards. CIU is initiated by the customer via the Web, using IBM Resource Link™, and makes use of CUoD techniques. CIU requires a special contract.

Unplanned upgrades

Unplanned upgrades can be done by the Capacity BackUp (CBU) for emergency or disaster/recovery situations.

CBU is a concurrent and temporary activation of Central Processors (CPs) in the face of a loss of customer processing capacity due to an emergency situation. CBU cannot be used for peak load management of customer workload.

CBU features, one for each “stand-by” PU, are optional on zSeries and require available PUs. A CBU contract must be in place before the special code that enables this capability can be loaded on the customer machine.

3.5 New HMC/SE functions

Users familiar with the existing S/390 and zSeries Hardware Management Consoles (HMCs) should have no problem in navigating the panels on the new z990 HMC. In this section we highlight the most significant changes and enhancements for the z990.

3.5.1 Integrated 3270 Console

A new HMC icon provides an emulated 3270. One emulated 3270 is available for each LPAR. The emulated 3270s do not appear as standard I/O devices. That is, there is no address (device number) associated with them and special programming is needed to address the 3270 sessions.

The Integrated 3270 Console is exploited by z/VM V4R4 and removes the need for a dedicated z/VM system console and associated 2074 control unit. This feature is exploited by z/VM, the z/VM Standalone Program Loader, and the standalone DASD Dump-Restore program. It can only be used for one console session per z/VM LPAR at a time. Multiple HMCs cannot each simultaneously establish a console session with the same z/VM LPAR.

z/OS currently does not support the Integrated 3270 Console. It cannot be used system console functions or by VTAM®.

The Integrated 3270 console icon can be found on the “CPC Recovery” page on the right-hand side of the HMC display. It is invoked in exactly the same way as the HMC Operating System Messages display. For example, a CPC Image icon that represents the target z/VM LPAR can be dragged and dropped onto the “Integrated 3270 Console” icon.

The hardware support for this feature is also integrated into G5/G6 processors with driver 26 and z800 and z900 processors with driver 3G.

3.5.2 Integrated ASCII console

The Integrated ASCII Console can be exploited by Linux logical partitions (not Linux guests under z/VM) with a planned code drop to the Open Source community. This feature allows a Linux LPAR to establish an ASCII console session on the HMC and should simplify Linux installation and initial operation. It can only be used for one console session per Linux LPAR at a time. Multiple HMCs cannot each establish a console session with the same Linux guest.

The Integrated ASCII console icon can be found on the “CPC Recovery” page on the right-hand side of the HMC display. It is invoked in exactly the same way as for the HMC Operating System Messages display.

The hardware support for this feature is also integrated into G5/G6 processors with driver 26 and z800 and z900 processors with driver 3G.

3.5.3 Optional *Strict* password rules

This new feature enforces strict password rules defined by the ACSADMIN userid and prompt the HMC user for a new password if the existing one has expired. Password expiration intervals can be set on the ACSADMIN “User Administration” screen.

3.5.4 Customizable HMC Data Mirroring

This feature allows two or more Hardware Management Consoles to be *associated* by sharing the same user data file. This eliminates the need to copy data between HMCs via diskette. The mirroring support can be customized to specify the data that is to be mirrored.

3.5.5 Extended console logging

The “View Console Tasks Performed” log size has now been increased to show the last 500 actions. Previously this log only showed the last 100 events.

3.5.6 Operating System Messages display

The “Operating System Messages” display has been enhanced to display the command line on the first page. A check box has been added to indicate a “Reply to Priority Message”.

3.6 SE and HMC connectivity

Configuring and ordering SEs and HMCs requires an understanding of the LAN interfaces involved. Both units normally have two LAN interfaces. The general rules are these:

- An SE is connected to a given HMC by only one LAN.
- In simple situations, the second LAN interfaces included with SEs and HMCs are not used.
- The two LAN interfaces on an SE may be used to connect to different sets of HMCs, using two independent LANs.
- The two SE LAN interfaces may be both Ethernet, or one token ring and one Ethernet. An option of two token ring interfaces is not available.[6] Both SEs in a system will have the same LAN configuration.
- A default HMC configuration has one Ethernet and one token ring interface.

SEs and HMCs may be connected through public LANs, but this is typically not done for several reasons:

- It is an obvious security exposure.
- It could result in connection losses, which would probably impact customer operation.
- MCL distribution from HMC to SE cannot tolerate connection losses.
- An SE upgrade or restore is quite sensitive, and could be impacted if there is much traffic on the LAN.

A very common arrangement is to use at least two HMCs, one near the system and the other near the general operations control area.

The following discussion assumes that you purchase a new HMC with your z990. You can purchase any reasonable number of HMCs to use with the system.[7] You have the following choices when you order your system:

- SEs with a token ring and an Ethernet adapter. These are supplied with two 50-foot Ethernet cables.
- SEs with two Ethernet adapters. These are supplied with four 50-foot Ethernet cables.
- HMC with one token ring and one Ethernet adapter. The HMC comes with a 75-foot token ring cable, a 7-foot token ring cable, and a 50-foot Ethernet cable.
- A display for the HMC.
- An MAU for token ring is included in the z990 frame if your Support Elements include token ring ports. You do not need to order the MAU in this case.
- An Ethernet switch. This is ordered automatically if your order does not include token ring interfaces.

The SE and HMC Ethernet ports can run at 10 Mbps or 100 Mbps and autosense the LAN speed.

[6] There is no obscure theoretical reason for the mixture of LAN interface types offered. It is based on the practical observation that an Ethernet port is now included on the planar board of most PCs, whether you use it not.
[7] Up to 32 HMCs can be used to control your z990 system.

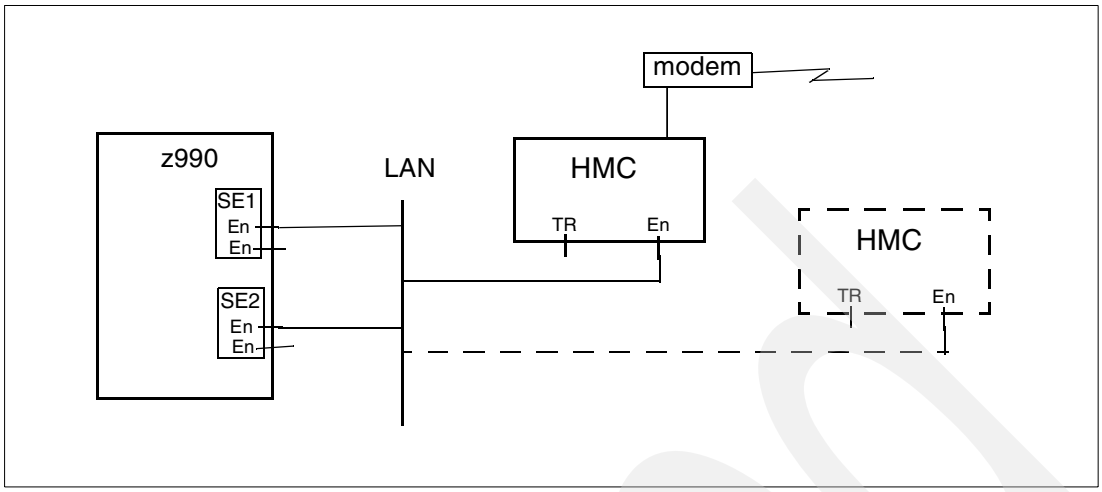


Figure 3-3 Basic SE - HMC connections

Figure 3-3 illustrates a basic SE/HMC configuration. This example uses Ethernet. The mixed token ring/Ethernet SEs would work just as well. A second HMC might be connected to permit operator actions from a different location.

What is *not* shown in this illustration is important—the second Ethernet interfaces are not connected to the LAN. OS/2 (the operating system for SEs and HMCs), in the implementation used for these functions, will not automatically use a second interface to the same LAN as an alternate path if the first interface fails.

The modem shown in the sketch is required if a LAN connection from the HMC to the IBM Network is not available.[8] It is used for Remote Support Facility (RSF) connections to transfer microcode updates (MCLs) and system status information via automated scheduled transmissions. Also, if a system fault occurs, the HMC places a call to the IBM Support System, alerting IBM and transferring error information and logs to reduce the impact of unplanned outages.

Figure 3-4 illustrates use of both LAN adapters in the SEs. This illustration uses Ethernet, but the principles are the same if token ring is used for one of the LANs.

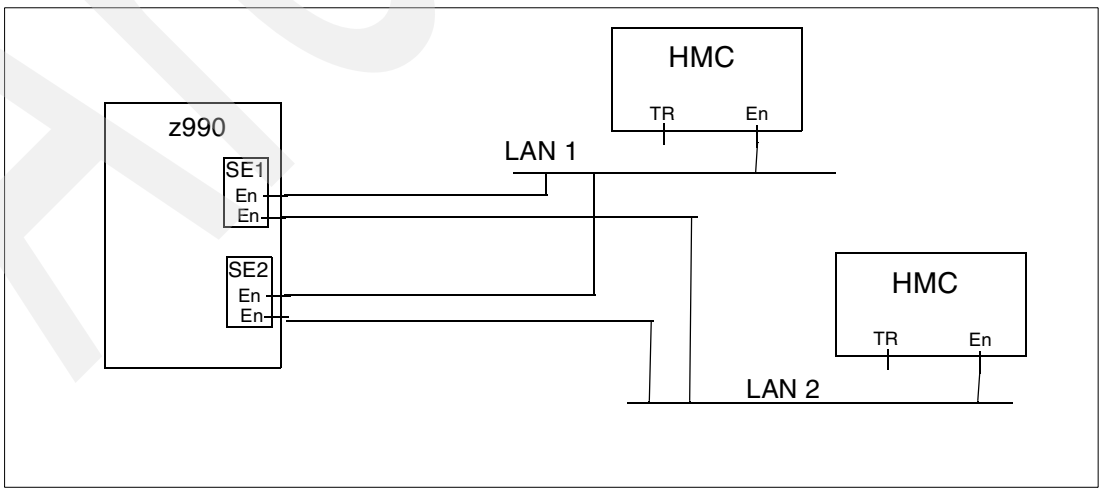


Figure 3-4 Alternate LAN use

[8] This option is not available in *secure accounts*, where a connection to the IBM Support System via modem is not allowed. A LAN connection to a public Internet Service Provider is sufficient for connection to IBM facilities.

In the examples in both figures, other z990 or z900 systems might be connected to the same LANs to share the HMCs.

3.7 ESCON channels

On the z990, ESCON channels are always delivered with the newer 16 channel I/O cards. The older 4-port cards cannot be used. The channels are packaged with 16 ports on a single I/O card. Up to 15 channels on each card are available for use; the last channel is reserved as a spare. In practice, any unallocated ports on the card can act as a spare.

A minimum of two ESCON channel cards are always installed if any ESCON channels are configured for the processor. ESCON channels are ordered in groups of four. For configurations greater than 28 ESCON channels, individual (not pairs) of ESCON channel cards are added as necessary. The active channels are distributed across the physical cards to provide additional redundancy.

If one of the activated ports fails, the system performs a *call home* to inform IBM. An IBM Service Representative will initiate the repair by selecting the “Perform a Repair Action” icon at the Service task panel on the SE. This will start the Repair&Verify procedure.

- If sparing can take place, then the IBM Service Representative moves the external fiber optic cable from the failing port to a spare or unconfigured port on the same card.
- If sparing cannot be performed, the card will be marked for replacement by the procedure. Upon replacement of the ESCON card, the cables that were changed are installed at the *original* port locations. Repair&Verify will recognize the unused ports on the new card as candidates for future sparing.

These ESCON cards, which are also used with z900 and z800 machines, use the small MT-RJ connectors. These are different from the traditional ESCON connections that are familiar to most S/390 owners. You can use an ESCON cable with an MT-RJ connector on one end (for the channel connection) and a traditional ESCON connector on the other end (for the control unit). Or you can use conversion cables.

The conversion cables are short cables with an MT-RJ connector on one end and a duplex connector for traditional ESCON cables on the other end. This permits you to use your existing ESCON cables.

The converter cables are relatively fragile and should only be used for converting large numbers of channels if a suitable housing/cabinet is placed in the floor to anchor the connections. An optional wiring harness provides a block of quick-disconnect junctions for connecting groups of conversion cables to existing ESCON cables.

3.7.1 Consideration for ES conversion channels

There are considerations for ES conversion channels, each of which should be connected to an ESCON convertor. When one of the ES conversion channel types (CVC, CBY) is defined, the channel hardware expects that an ESCON convertor is connected to the channel. If the convertor is not connected, a permanent hardware error may be reported at POR. We recommend you do not define an ES conversion channel type until the convertor is actually connected.

3.8 ESCON directors and multiple LCSSs

An ESCON channel may belong only to a single LCSS. What happens when ESCON Directors are involved, as illustrated in Figure 3-5?

In this example we have two channel subsystems. Each LCSS might have several LPARs associated with it, but no LPAR is associated with more than one LCSS. So, how does this situation appear to the ESCON director and the two disk sets?

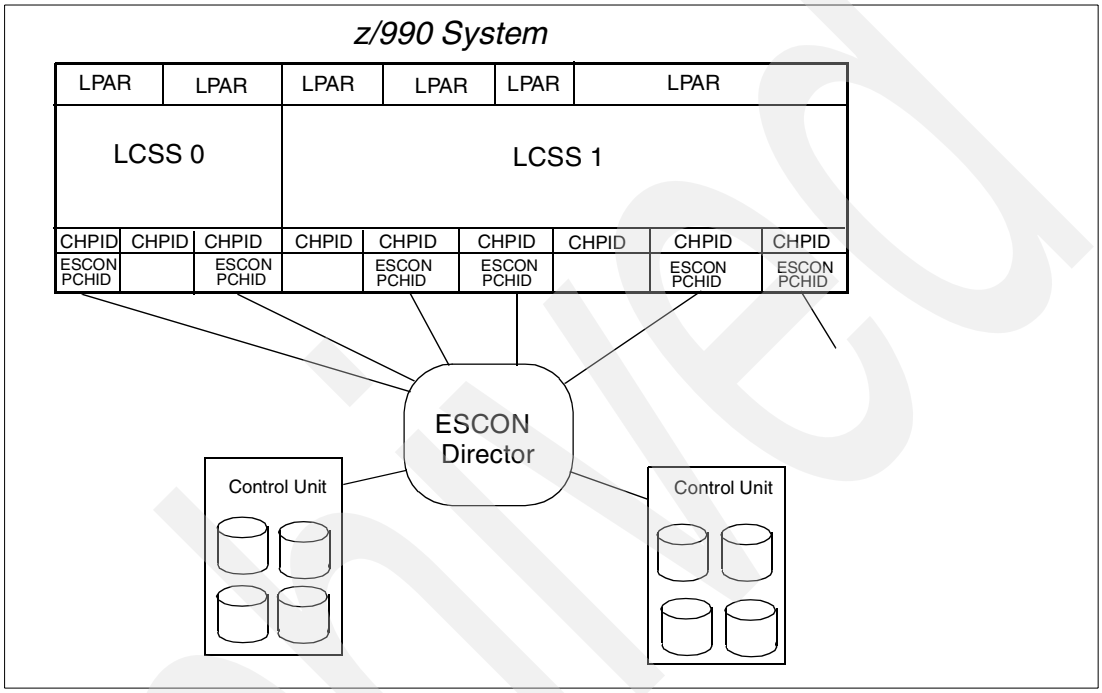


Figure 3-5 DASD connected to multiple LCSSs via ESCON Directors

For all practical purposes, this configuration is treated as if it were two separate zSeries machines.

3.9 HiperSockets

HiperSockets provide high-performance “networks in a box.” A z990 has up to 16 internal HiperSocket LANs compared to the four supported on the z900 processor. Each of these uses a special shared internal CHPID that can be accessed by candidate partitions specified in the IOCDS. Full control can be exercised over which partitions can or cannot access each HiperSocket LAN. For example, you may which to designate certain HiperSocket LANs as production and restrict access to them to partitions running production application servers, database servers, firewalls, and so forth. Similarly, you could designate other HiperSocket LANs for test or development purposes only. Access to a HiperSocket LAN is via an IQD CHPID; all partitions connecting to one of these CHPIDs are in effect, sharing the same internal LAN.

The characteristics of HiperSocket LANs include:

- Excellent performance and response times because all operations are through the system memory bus.
- High availability because there are no external parts or connections involved.

- Cost savings, again because no external parts, OSA-Express adapters or physical network cabling is involved.
- General connectivity for z/OS, Linux, and z/VM, with standard CPs or IFLs.
- Easy to install and implement since they operate like a traditional LAN.
- High levels of security:
  - There is no physical network that might be used for unauthorized connections.
  - Only the partitions authorized to access the IQD CHPID associated with that HiperSocket LAN may connect to it.

  Because of this inherent security, it is probably not necessary to encrypt the traffic travelling over the HiperSocket LAN.
- Up to 4,096 communications queues across all 16 HiperSocket LANs (the z900 supported 4 HiperSocket LANs and 1,024 queues). Each communication queue may be shared by multiple LPARs. Each TCP/IP stack within an LPAR must use a different communications queue, although each queue can be shared across multiple LPARs with MIF and multiple Logical Channel Subsystems with spanning.
- No interference with normal system performance because HiperSocket data flow does not go through the processor L1 or L2 caches.
- Channel access is through QDIO programming.
- Multicasting is supported, but broadcasting is not supported.
- Not used by current TPF systems.

HiperSockets can be configured many ways. Figure 3-6 on page 46 shows a simplistic arrangement (all in one LCSS, which is not shown), but illustrates key concepts. Up to 16 HiperSocket CHPIDs can be specified, and each one is, in effect, a LAN. In the illustration, two HiperSocket LANs are used. One LPAR is connected to both HiperSocket LANs and to an external LAN. This LPAR could be used as a router.

TCP/IP connections to each LAN are handled as normal TCP/IP connections. Notice that HiperSockets exist only within the system. There are no external connections, not even to another zSeries machine. Any external connection must be through other means, such as OSA-Express adapters, channel-to-channel connections, XCF connections, and so forth. However, a z/OS partition connected to an OSA-Express adapter can act as a HiperSocket Accelerator and provide high-speed routing of traffic between an external LAN and the systems connected to a HiperSocket LAN

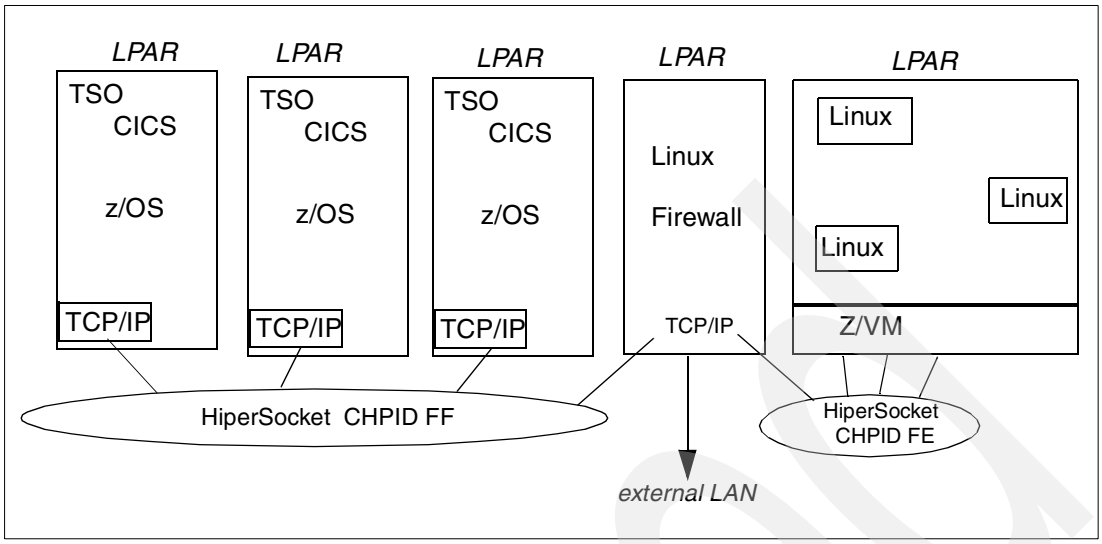


Figure 3-6 Two HiperSocket LANs in a z990 system

HiperSockets are a major addition to IBM's architectures and are expected to become a significant element in future middleware and application designs.

3.9.1 Defining HiperSockets in IOCP statements

To use HiperSockets in a z990, you define CHPID type IQD and associated control units and devices in the IOCDS. An IQD CHPID needs to be defined for each HiperSocket LAN that is in use.

As an example, we configured a HiperSockets network for a z990 environment as shown in Figure 3-7. This HiperSocket LAN *spans* two Logical Channel Subsystems permitting the Linux firewall LPARs in LCSS 1 to talk to the production z/OS LPARs in LCSS 0.

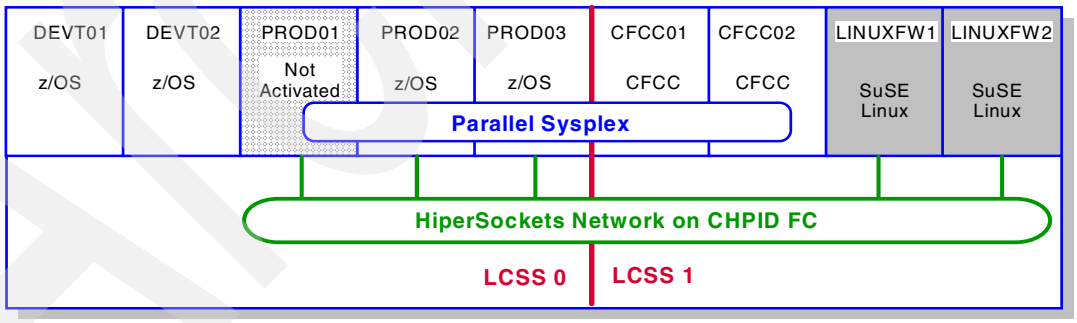


Figure 3-7 A typical HiperSocket LAN configuration on a z990 processor

In this case, we have 9 LPARs within a z990 system, including 2 CFCC LPARs for a Parallel Sysplex environment. A single HiperSocket LAN is used to connect two Linux and three z/OS systems.

The following is an extract of the IOCDS definitions needed for this HiperSockets network:

```
ID    MSG2='FOWLERI.IODF01 - 2003-02-26 11:07',              *
      SYSTEM=(2084,1),                                       *
```

```
                TOK=('SYSA',000000090ECB2064110740890103057F00000000,000*
                00000,'03-02-26','11:07:40','FOWLERI','IODF01')
*
          RESOURCE PARTITION=((CSS(0),(DEVT01,4),(DEVT02,5),(PROD01,1),(*
                PROD02,2),(PROD03,3)),(CSS(1),(CFCC01,3),(CFCC02,4),(LIN*
                UXFW1,1),(LINUXFW2,2))),                                *
                MAXDEV=((CSS(0),64512),(CSS(1),64512))
*
          CHPID  PATH=(CSS(0,1),FC),SHARED,                             *
                PARTITION=((CSS(0),(PROD01,PROD02,PROD03),(=)),(CSS(1),(*
                LINUXFW1,LINUXFW2),(=))),CHPARM=C0,TYPE=IQD
*
          CNTLUNIT CUNUMBR=E000,PATH=((CSS(0),FC),(CSS(1),FC)),UNIT=IQD
*
          IODEVICE ADDRESS=(E000,003),CUNUMBR=(E000),UNIT=IQD
```

Because HiperSockets are not associated with a *physical* CHPID, there is no associated PCHID parameter on the CHPID statement. In this example, device numbers E000-E002 will be used in all partitions that use the HiperSocket.

Access to the HiperSocket LAN is controlled by the IQD CHPID access and candidate lists; these are specified on the PARTITION parameters on the CHPID. The definitions above prevent the development LPARs from connecting to the HiperSocket LAN.

The new CHPARM parameter on the CHPID statement has replaced the OS parameter used on the z800 and z900. This parameter is used to specify the maximum frame size and the Maximum Transmission Unit (MTU) for each HiperSockets network; see Table 3-1. We defined CHPARM=C0, which indicates that the maximum frame size is set to 64 KB and the MTU to 56 KB. The CNTLUNIT statement for HiperSockets cannot specify a UNITADD parameter; it defaults to UNITADD=((00,256)).

Table 3-1 IQD CHPID maximum frame size and MTU size

CHPARM=value	Maximum frame size	Maximum Transmission Unit size
00 (default)	16 KB	8 KB
40	24 KB	16 KB
80	40 KB	32 KB
C0	64 KB	56 KB

However, if you are using HCD to create your definitions, you must specify the address range explicitly. If you have only one TCP/IP stack in a z/OS image, only the first three device addresses are used, and they are used as *read control*, *write control*, and *data exchange* devices, respectively. (Note that we omitted many IOCP statements that are not related to the HiperSocket definition in this example).

3.9.2 Spanning HiperSockets over multiple LCSSs

The previous example shows a HiperSocket LAN *spanned* over two Logical Channel Subsystems. An IQD CHPID can be spanned over multiple Logical Channel Subsystems.[9] This creates a common definition for the CHPID across the Logical Channel Subsystems involved and allows the LPARs within those LCSSs to share the LAN channel (in practice,

[9] The *architecture* of spanning is not limited to spanning two Logical Channel Subsystems. If more LCSSs existed, a single channel could be spanned over all of them or any subset.

sharing of the spanned CHPID is further controlled by the use of access and candidate LPAR lists on the CHPID definition).

An IQD CHPID is designated as being spanned if it has multiple CSS values assigned on the CHPID IOCP statement and also has the SHARED keyword specified.[10] Within HCD on z/OS, a CHPID is designated as spanned if the Operation Mode is specified as SPAN when it is defined.

The TCP/IP Dynamic XCF (DXCF) support in the z/OS Communications Server has been updated to support spanned IQD CHPIDs. During startup, DXCF will dynamically learn which logical partitions in different LCSSs are accessible via the spanned IQD CHPID. Whenever possible, DXCF will then use the HiperSocket LAN to communicate with these logical partitions in preference to using XCF links.

An IQD CHPID does not, of course, need to be spanned. If it is not spanned, the HiperSocket LAN is only accessible to the LPARs defined within one LCSS.

3.9.3 Defining HiperSockets in the z/OS TCP/IP profile

Devices and link statements are needed in the z/OS TCP/IP profiles to define the HiperSocket. The key portions of the profile for one of the z/OS definitions are as follows:

```
;
; HiperSockets
;

 DEVICE IUTIQDFC MPCIPA
 LINK HIPERLFC IPAQIDIO IUTIQDFC

HOME
   10.1.0.101     HIPERLFC
;  10.1.0.102     HIPERLFC
;  10.1.0.103     HIPERLFC

  BEGINRoutes
    Route 10.1.0.0/24 = HIPERLFC mtu 16384
  ENDroutes

START IUTIQDFC
```

You do not specify HiperSockets device numbers (device addresses) in the TCP/IP profile. Instead, specific device names are used to identify the HiperSocket CHPID you are using. The base name is IUTIQDxx, where xx denotes the IQD CHPID. In our case, we used CHPID FC for IQD, and the device name in our TCP/IP profile is IUTIQDFC. (This scheme, where device addresses are not used, is unusual for z/OS.)

3.10 Channel measurement changes

The Channel Measurement Block (CMB) introduced with the XA I/O architecture has been enhanced and extended. This includes the basic changes needed to support the physical and Logical Channel Subsystems on the z990 processor. These Extended Channel Measurement Blocks (ECMBs) can be exploited by operating systems with the appropriate

[10] The SHARED keywork is optional for spanned channels, since it is implied by spanning.

level of support, for example z/OS V1R4 with the z990 exploitation support. For compatibility reasons, the z990 processor also supports the original XA I/O architecture Channel Measurement Blocks.

The storage required for Channel Measurement Blocks was previously determined by the number of DASD and tape devices installed and the CMB parameter from the IEASYSxx member in the z/OS PARMLIB. This was used to allocate a block of contiguous real storage, at IPL time, and this block could not be extended. The new design, using Extended Channel Measurement Blocks, no longer requires contiguous real storage and can be extended after the initial allocation at IPL. The ECMBs now reside in a system common area dataspace when using z/OS V1R4 with z990 exploitation support.

The ECMBs have also increased in size to 64 bytes each. This is to allow the storage of model-dependent device measurement data and future expansion. The SSCH/RSCH counts and sampling counts have now been expanded to a full word each to reduce the possibility of wraparound.

3.11 Sysplex coupling considerations

Figure 3-8 on page 50 illustrates the components of a Parallel Sysplex as implemented within the z990 architecture. Shown are two ICFs, one in each z990, connected to three z990 servers running in sysplex. Each of the two z990 containing the integrated Coupling Facility partition also have sysplex LPAR images.

Also illustrated is the connection required between the Coupling Facilities (defined in the z990s) and the Sysplex Timer. This supports Message Time Ordering (and this is independent of the fact that the other partitions in these z990 are also members of the same sysplex). Note that Message Time Ordering requires a CF connection to the Sysplex Timer whenever:

- The Coupling Facility is an LPAR or ICF on a z990.
- The server does not have Sysplex Timer connectivity to the Parallel Sysplex supported by the CF partition.

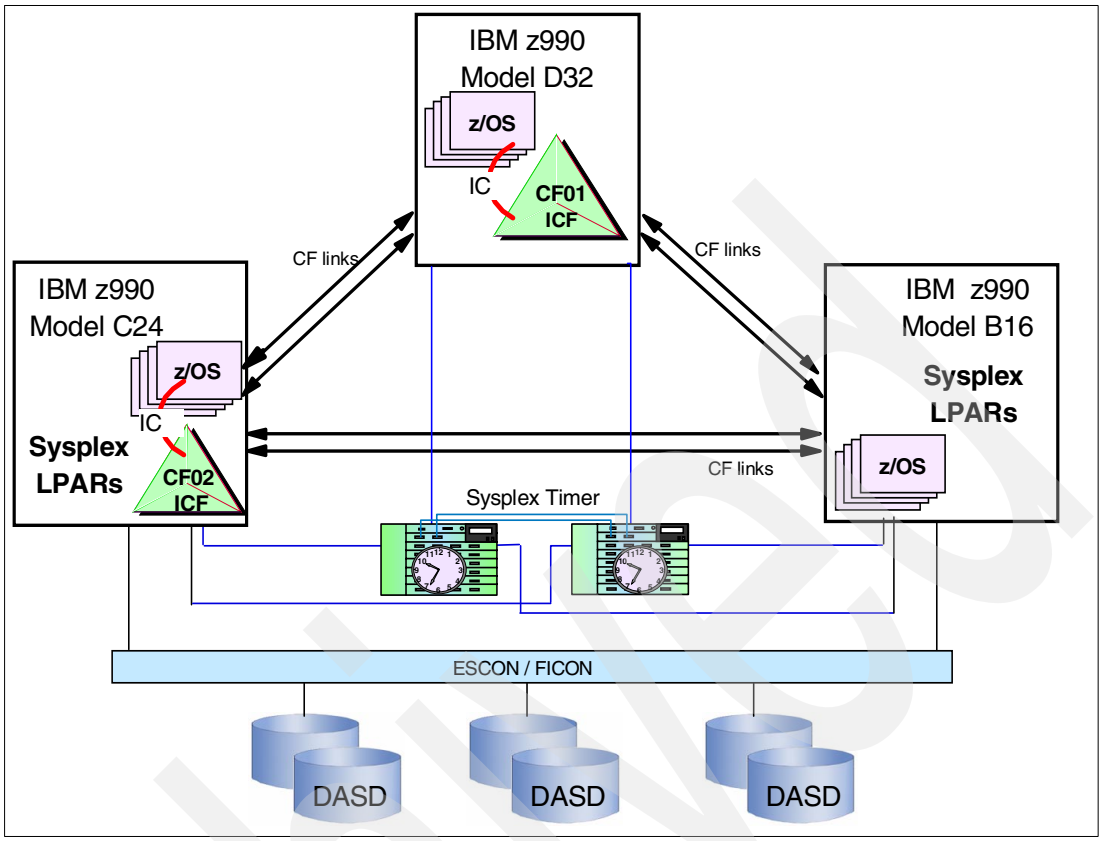


Figure 3-8 z990 Parallel Sysplex connectivity

3.11.1 CF Links on z990

The type of CF links you can use to connect a CF to an operating system LPAR is important because of the impact of the link performance on response times and coupling overheads. The types of links that are available to connect an operating system LPAR to a CF on a z990 are:

- **IC**s: Microcode-defined links to connect a CF to a z/OS LPAR in the same z990 processor. IC links require two CHPIDs to be defined and can only be defined in peer mode. The link speed is greater than 2.0 GB/sec.
- **ICB-4**s: Copper links available to connect two z990s processors; the maximum distance between the two processors is 7 meters (maximum cable length is 10 meters). The link speed is 2.0 GB/sec. ICB4 links can only be defined in Peer mode. The maximum number of ICB4 links is 16.
- **ICB-3**s: Copper links available to connect a z990 to z900 or z800 processors; the maximum distance between the two processors is 7 meters (maximum cable length is 10 meters). The link speed is 1 GB/sec. ICB3 links can only be defined in Peer mode. Maximum number of ICB3 links is 16.
- **ICB**s: (These are also referenced as ICB-2s.) Copper links available to connect a z990 to 9672 G5/G6 processors; the maximum distance between the two processors is 7 meters (maximum cable length is 10 meters). The maximum link speed is 333 MB/sec, with 250 MB/sec being typical. The maximum number of ICB2 links is 8.
- **ISC-3**s: Fiber links defined in peer mode available to connect z990 to z990, z900 or z800 processors. The maximum distance is 10 km, 20 km with an RPQ, or 40 km with Dense Wave Division Multiplexing (DWDM) connections. Link speed is 200MB/sec (for distances

up to 10 km) and 100MB/sec for greater distances (RPQ). The maximum number of ISC links is 32.

- **ISC-3**s: Fiber links defined in compatibility mode are used to connect to G5/G6 systems. The maximum distance is 10 km, 20 km (RPQ), or 40 km with DWDM connections, These operate on single mode fiber at 100 MB/sec or multimode fiber at 50 MB/sec. The maximum number of ISC links is 32.

Note: The number of combined ISC-3s, ICB-2s, ICB-3s, and ICB-4s CHPIDs cannot exceed 32.

There are several advantages in using peer mode links. All peer mode links operate on a higher speed than the equivalent non-peer link. A single CHPID (one side of the link) can be both sender and receiver; this means that a single CHPID can be shared between multiple OS LPARs and one CF LPAR. The number of subchannels defined when peer mode links are used is seven (7) per LPAR per link compared to two (2) per LPAR per link in compatibility mode. This is particularly important with System-Managed CF Structure Duplexing. Peer links have 224 KB data buffer space compared to 8 KB on non-peer links; this is especially important for long distances as it reduces the handshaking for large data transfers.

3.11.2 CF considerations

The z990 family of processors will not provide a special model for a CF-only processor (like the model 100 for z900 processors). You can, however have a z990 processor with all enabled PUs defined as ICFs.

3.12 FICON Express

FICON Express links offer performance improvements over earlier FICON technology. I/O operations per second can be as high as 7200 (native FICON) or 6000 (bridge,) assuming 4K byte data transfer sizes. Effective bandwidth of up to 170 MB/second can be achieved with highly sequential I/O operations, such as moving tracks of data, with native FICON channels and 2 Gbps links. Additional information about FICON Express channels can be found at:

http://www-1.ibm.com/servers/eserver/zseries/connectivity/

FICON links may be used in several ways, as illustrated in Figure 3-9 on page 52:

- Connection to a FICON Bridge (contained in an IBM 9032-5 ESCON director), which converts the FICON link to a maximum of 8 ESCON links.
- Direct connection to control units that have native FICON connectivity.
- Connection to a FICON Director which, in turn, has FICON links to control units supporting FICON connections.
- Connection to another FICON port, creating a channel-to-channel (CTC) connection. (The CTC functions are described further in “FICON CTC” on page 53.)
- Connection to a FICON Director, used to create CTC connection or loops.

IBM has reseller agreements with McDATA and INRANGE to supply Fibre Channel Directors.

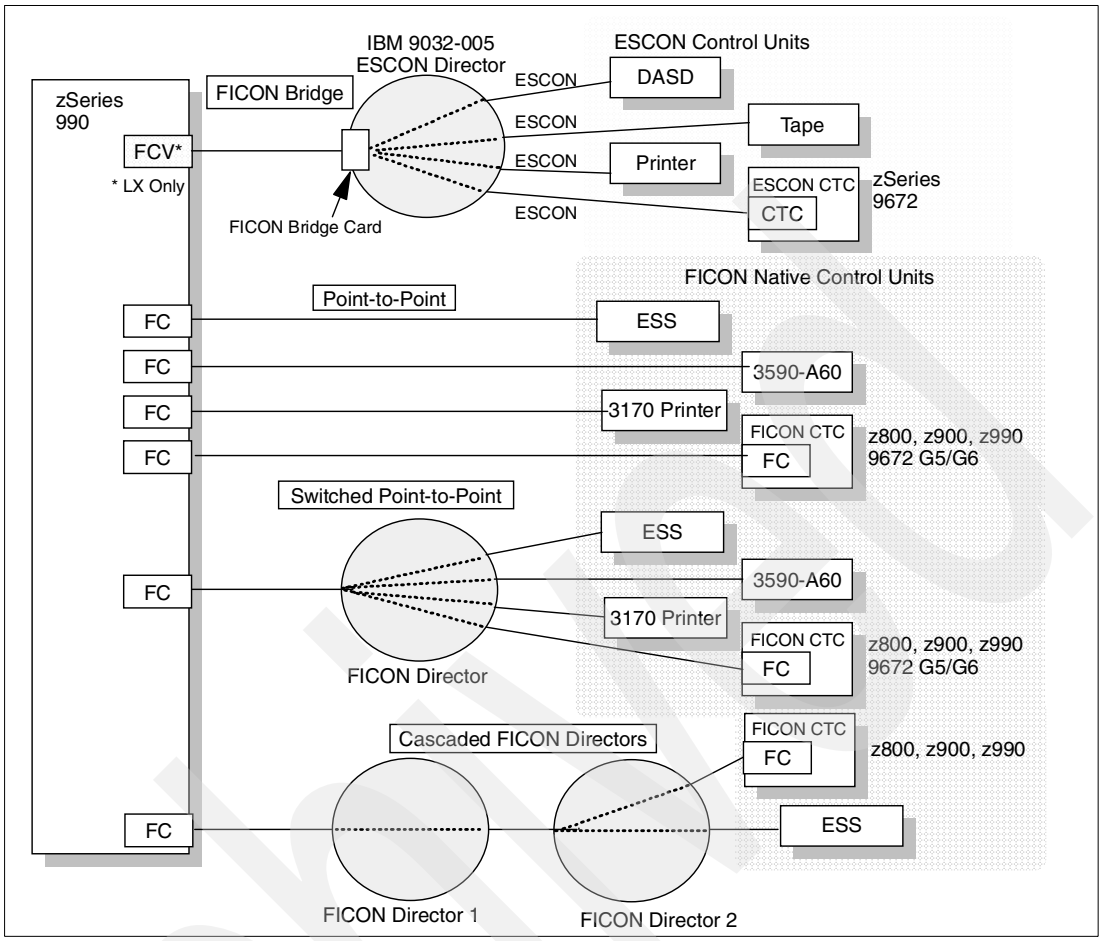


Figure 3-9 FICON connectivity options

There are two versions of the FICON Express card:

- The LX model is for long wavelength operation. It normally uses 9 micron single mode (SM) fiber with an LC Duplex Single Mode connector. Without using repeaters, the maximum supported distance is 10 km at 1 or 2 Gbps.[11] The LX version is compatible with the FICON bridge. The LX version can also be used with 50 or 62.5 micron multimode (MM) fiber if IBM's mode conditioning patch (MCP) cables are used; this reduces the operational distances possible, and cannot be used for higher data transfer rate.
- The SX model is for short wavelength operation and uses 50 or 62.5 micron MM fiber. It connects via an LC Duplex Multimode connector. This is the fiber typically used by SAN infrastructures. Maximum distances possible are 250 m (using 62.5 micron fiber) or 500 m (using 50 micron fiber) at 1 Gbps data transfer rate. At 2 Gbps, the maximum distances are 120 m (with 62.5 micro fiber) and 300 m (with 50 micron fiber). The SX version cannot be used with a FICON bridge.

The former FICON adapters (that is, not the FICON Express cards) for z900[12] are no longer orderable and cannot be used with a z990. If you are upgrading from a z900 processor to a z990, you can use the same FICON Express cards; former FICON cards (non-Express) cannot be used.

[11] An RPQ exists to extend this to 20 km at 1 Gbps and 12 km at 2 Gbps.
[12] z900 can intermix and operate both FICON and FICON Express cards at driver 3C or later, but z990 can use FICON Express cards only.

Each port on a FICON Express card can be initialized with one of three microcode loads:

- Operation in bridge mode. This assumes the FICON is connected to an appropriate bridge unit in a 9032-005 ESCON director. The bridge unit converts the FICON connection into a maximum of eight ESCON connections. This requires an FCV channel type.
- Operation in native FICON mode. This works with native FICON control units. An FCTC CUs can be defined on an FC channel on a z990, z800, or a z900 processor at driver level 3C or later.
- Operation in FCP mode. This is channel type FCP and provides an SCSI interface.

The microcode loading is done at Power-on Reset and can be reloaded by dynamic I/O reconfiguration changes initiated from a z/VM, z/OS, or OS/390 image running on the z990.

3.12.1 FICON CTC

z990 systems support FICON channel-to-channel (CTC) connections. This has several important characteristics compared with ESCON CTC:

- A given FICON channel can be used for both CTC and normal I/O connections, whereas an ESCON CTC channel must be used one way or the other. A FICON channel can multiplex CTC traffic along with other traffic. An ESCON CTC channel cannot do this.
- A FICON CTC connection involving zSeries systems will automatically determine which “side” of the connection will perform the CTC control unit function.
- Any two LPARs, in the same system, can establish a FICON CTC connection using only a single FICON channel. This single channel *must* be connected to a FICON Director. This is illustrated in the bottom of Figure 3-10 on page 54.
- More FICON CTC devices can be defined for a FICON channel. Up to 16K CTC devices can be defined for an FC channel, whereas the number of CTC devices defined for an ESCON CTC channel is limited to 120.
- One end of a FICON CTC link must be a zSeries system, with a FICON or FICON Express card installed, because only these have the ability to perform FICON CTC control unit functions. The other end can be either a zSeries machine or a G5/G6 9672 system.

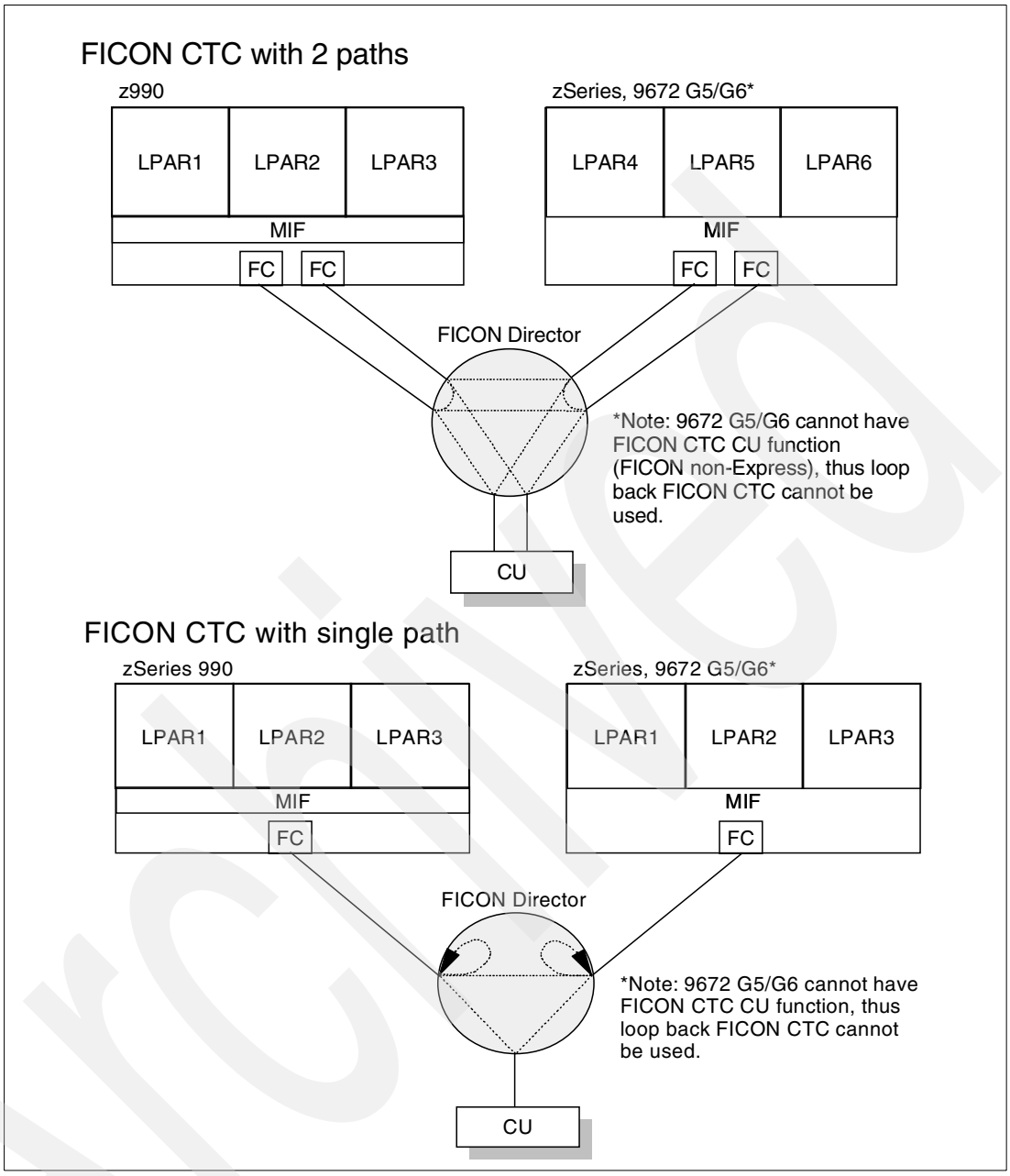


Figure 3-10 FICON CTC links in director

In the upper part of Figure 3-10, two FICON native (FC) channels in z990 are shared among LPAR1, LPAR2, and LPAR3, and communicate with LPAR4, LPAR5, and LPAR6 on another processor. The same FICON channels can also communicate with I/O devices (CUs) attached to the Director.

The bottom half of Figure 3-10 illustrates that a single FICON channel can be looped in a director, to form a CTC link between two LPARs in the same system. For communication between any two of the LPARs in the z990, both FICON channels are used to form a CTC loopback in the FICON Director. The same channel can also communicate with other systems or control units.

When a FICON channel connects to a FICON channel to form a CTC function, the channels will automatically determine which “end” will perform the CTC operation. At least one of the “ends” of such a connection must be a zSeries machine.

In these two examples, the two systems could also be different LCSSs in a single z990 system.

3.13 Characterization

The type of Processing Units (PUs) that can be ordered (enabled or assigned) on a z990 processor are:

- Integrated Facility for Linux (IFL)
- Internal Coupling Facility (ICF)
- Central Processors (CPs)
- System Assisted Processors (SAPs)

There are two standard spare PUs in each book. If these have been used and another failure occurs in a customer-available PU, then an unassigned PU will temporarily be used to replace the failing PU.

3.13.1 Integrated Facility for Linux (IFLs)

An IFL, or Integrated Facility for Linux, provides additional processing capacity exclusively for Linux workloads. Traditional S/390 and zSeries software charges are typically not affected by the additional IFL processing capacity.[13] A processing unit (PU) enabled for IFL work is often referred to as an IFL “engine”. There are several characteristics of an IFL processing unit or engine:

- It may be used in one or more LPARs.
- It is not meant to run anything except Linux or Linux under z/VM Version 4. CMS may also be used under z/VM. The principle is that CMS is needed to manage z/VM, and the primary purpose of z/VM in this case is to host Linux guests. However, most licensed software products for z/VM cannot be licensed to run in an IFL.
- Several PUs may be enabled as IFL processors.
- The IFL PUs may be spread among LPARs used for Linux in any desired manner. For example, you might have one IFL PU running three Linux LPARs; or you might dedicate one IFL PU to a particular Linux LPAR and share another IFL PU among four more Linux LPARs.
- You cannot use standard S/390 PUs and IFL PUs in the same LPAR.
- Linux LPARs are standard LPARs, created by a RESOURCE keyword in the IOCDS and marked Linux Only in the LPAR activation profile.

Figure 3-2 on page 56 illustrates a z990 with three enabled PUs: one standard and two IFLs. The system administrator has created four LPARs. One PU is used by two LPARs for two copies of z/OS. Both IFL PUs are shared by two Linux LPARs. One of these is running Linux directly, and the other is running multiple Linux images under z/VM.

[13] This is a simple statement for a very complex topic and may not always be true. In general, most IBM software costs are not affected by the addition of IFL processors. Other software vendors may have different policies. For software prices that are tied to system model numbers, the statement is usually true because the addition of IFLs does not change the model number of the system. Since model numbers of a z990 system do not indicate how many processors are available, newer software pricing methods may negate this advantage of IFLs.

While an IFL PU cannot be used to run normal zSeries operating systems, such as z/OS, a standard PU may be used to run Linux (and Linux under z/VM). However, you cannot mix standard PUs and IFL PUs in the same LPAR. There are no restrictions or technical disadvantages to running Linux (or Linux under z/VM) with standard PUs. The sole reason for having IFLs is to reduce the cost of software used by the standard PUs.

For example, say you purchase a system with two PUs active and have reconciled all your normal z/OS software costs to this model. You now want to use Linux on the system. If you have unused processing capacity (with your two PUs), you can simply create another LPAR and install Linux in it. If you eventually need more processing capacity, then you must do some analysis. You could upgrade to three PUs and share the additional processing power among all LPARs (including Linux). However, this upgrade will probably cause your software license costs to increase to correspond to the new system capacity.

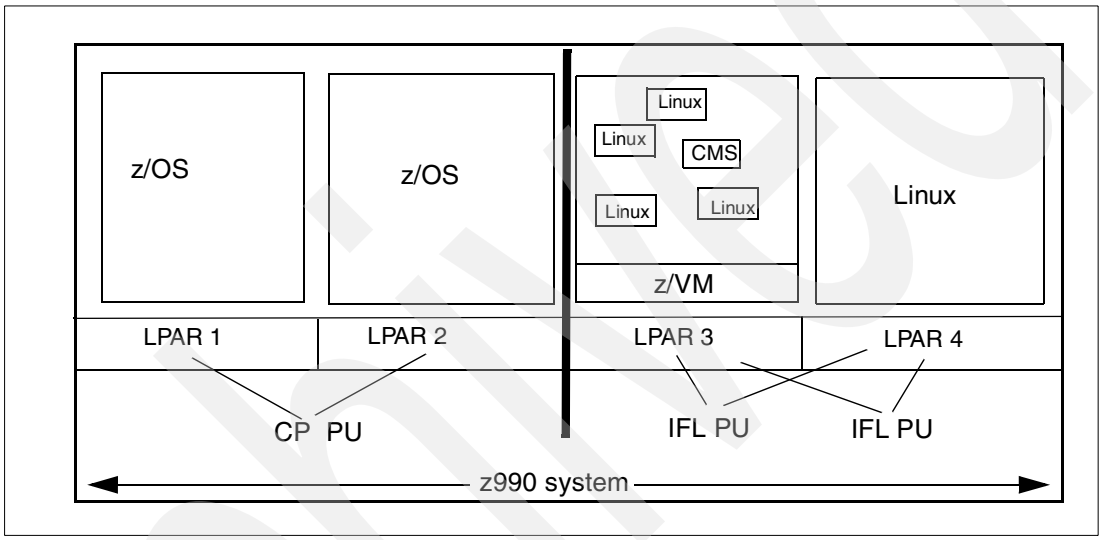


Table 3-2 Separation of standard and IFL PUs

The alternative is to keep your two PUs and add an IFL PU. This will increase your processor capacity, but only for Linux work. You would probably offload Linux work from your standard PUs by moving it to the IFL PU, but this is not required. You could, if your situation made it reasonable, run a Linux LPAR[14] with your standard PUs, and other Linux LPARs with your IFL PU. The only restriction is that you cannot have standard PUs and IFL PUs in the same LPAR.

Adding an IFL

The addition of an IFL engine is not disruptive, assuming unused PUs exist in your system. However, bringing the IFL into a Linux partition can be. Planning will need to be done ahead of time in order to activate the Linux LPAR nondisruptively. In order to use the IFL without a POR, certain conditions must exist:

- The Linux partition was defined in the IOCDS used at the last POR.
- Resources can be made available to activate the Linux partition without having to POR the machine or having to deactivate another partition.

If the Linux LPAR is already running, but is using a standard PU (CP) instead of an IFL, the partition must be deactivated, redefined in its Image Profile to use the IFL, and reactivated. This can all be done via the HMC, but it is disruptive to the Linux partition. When defining an

[14] This term “Linux LPAR” includes Linux running natively in an LPAR and Linux running under z/VM.

LPAR Image profile to use an IFL, the choices in the Image Profile[15] are CP or ICF; you must select ICF at this point in order to use an IFL. This is because the PR/SM hipervisor treats ICFs and IFLs in the same manner. The ICF choice will not appear unless an IFL is present on the machine and the partition type is selected to be Linux Only (for Linux or Linux under z/VM 4.2).

3.13.2 Internal Coupling Facility (ICF)

An ICF, or Internal Coupling Facility, provides additional processing capacity exclusively for the execution of the Coupling Facility Control Code (CFCC) in a CF LPAR. As with IFLs, traditional z/OS software charges are not affected by the additional ICF processing capacity. A processing unit (PU) enabled for ICF work is often referred to as an ICF engine. Functionally, the ICF is a PU engine which is configured to execute in a CF LPAR. Special PR/SM microcode precludes the defined ICFs from executing non-CFCC code. As a result of this behavior, software license charges are not applicable to ICF PUs.

The CFCC level code initially available on z990s is level 12; the following functions are provided by this level of CFCC:

- 64 bit support

  The 64-bit support within CFCC eliminates the 2 GB “control store” line in the CF. The distinction between “control store” and “non-control store” (also known as data storage) in the CF is eliminated. As a consequence, very large amounts of CF central storage can be used for both CF control and data objects. This support allows CFCC to exploit more than 2 GB of real storage.

- System-Managed CF Structure Duplexing

  This provides a general-purpose, hardware-assisted, easy-to-exploit mechanism for duplexing CF structure data. It provides a robust recovery mechanism for failure such as loss of single structure or CF or loss of connectivity to a single CF, through rapid failover to the other structure instance of the duplex pair.

- Support for 48 CF tasks

- Support for Message Time Ordering

  This provides enhanced scalability for Parallel Sysplex. The improved processor and Coupling Facility link technologies inherent on z990 processors necessitate more rigorous time synchronization tolerance for members of a Parallel Sysplex hosted by those models. The Message Time Ordering (also referred as CF Request Time Ordering) is necessary in order to ensure that any exchanges of timestamp information between systems in a sysplex involving the CF observe the correct time ordering. CF Request Time Ordering ensures data integrity in the event the Sysplex Timer is not able to synchronize the time-of-day (TOD) clocks to an accuracy that is smaller than the messaging time between systems. The fastest communication mechanism between processors is communication through the CF. The processor where the CF LPAR is defined must be connected to the same Sysplex Timer as other sysplex member.

- Support for batched IXL requests

  This support allows for batched group buffer pool (GBP) writes and CF cast out data requests, as well as CF cross-invalidate requests in single CF operations. This may reduce data sharing cost in update or insert intensive DB2 data sharing environments.

[15] The Image Profile is a series of GUI panels that is are used to define activation profiles for LPARs. These panels can be accessed through the SE or HMC.

3.13.3 Central Processor (CP)

A Central Processor is a PU that has the z/Architecture™ and ESA/390 instruction sets. It can run z/Architecture, ESA/390, Linux, TPF operating systems and the Coupling Facility Control Code (CFCC). z990 processors operate only in LPAR mode; consequently, all CPs are dedicated to a partition or shared between partitions. Reserved CPs can also be defined to a logical partition, to allow for nondisruptive image upgrades.

All CPs within a configuration are grouped into a CP pool. Any z/Architecture, ESA/390 and TPF operating systems can run on CPs that were assigned from the CP pool. Within the capacity of the z990 books, CPs can be concurrently added to an existing configuration via Capacity Upgrade on Demand (CUoD), Customer Initiated Upgrade (CIU), or Capacity BackUp (CBU). They can also be ordered as a simple upgrade (MES), of course.

3.13.4 System Assist Processor (SAP)

A System Assist Processor (SAP) is a PU that runs the channel subsystem Licensed Internal Code to control I/O operations. All SAPs in a z990 configuration are implemented as Master SAPs.

All SAPs perform I/O operations for all logical partitions and all attached I/O. z990 model A08 processors have 2 SAPs as standard, z990 model B16 processors have 4 SAPs as standard, z990 model C24 processors have 6 SAPs as standard, and z990 model D32 processors have 8 SAPs as standard.

A standard SAP configuration provides a very well-balanced system for most environments. However, there are application environments with very high I/O rates (typically some TPF environments), and in this case additional SAPs can increase the capability of the channel subsystem to perform I/O operations. Additional SAPs can be added to a configuration by either ordering optional SAPs or assigning some CPs as SAPs. Orderable SAPs may be preferred since they do not incur software charges, as might happen if CPs are assigned as SAPs.

3.14 CHPID Mapping Tool overview

The intent of the CHPID Mapping Tool is to ease installation of new z900, z800, and z990 processors. It is also intended for making changes to an already installed z990 processor (not for z900 or z800), either to make slight changes to the mapping or as part of an MES action to add or remove channel features on the processor. For the z800 and z900, some limited mapping can be accomplished on an installed processor, but that will not involve the use of the Mapping Tool. Rather, there is a task provided on the system Support Element (SE) to accommodate this case. On the z990, however, it requires the use of the mapping tool or HCD/IOCP to accomplish the mapping; there is no task on the z990 SE to do it.

The use of the tool for the z800 and z900 processor did not change from the previews standalone version that was available from the IBM Resource Link. Therefore the rest of this topic will address only the use of the tool for z990 processors.

The z990 processors, contrary to previous generations of processors, do not have default CHPIDs assigned to ports as part of the initial configuration process. It is the customer's responsibility to perform these assignments by using the HCD/IOCP definitions and, optionally, the CHPID Mapping Tool. One result of using the tool is an IOCP deck that will map the defined CHPIDs to the corresponding PCHIDs of the processor. There is no *requirement* to use the Mapping Tool. You can assign CHPIDs to PCHIDS directly in an IOCP decks or through HCD. However, this is a very cumbersome process for larger

configurations (as would be expected to many z990 systems), and it is much easier to use the tool to do the channel mapping. If customers choose to do manual assignment of CHPIDs to PCHIDs (using HCD or IOCP), it will be their responsibility to distribute CHPIDs among the physical channel cards (PCHIDs) for availability. The objective of the tool is to help in performing these tasks.

Figure 3-11 shows a diagram containing the suggested steps that the installation can take in defining a new z990 I/O configuration. This section will concentrate in describing the functions associated with the tool: the input that is needed and the output that is created.

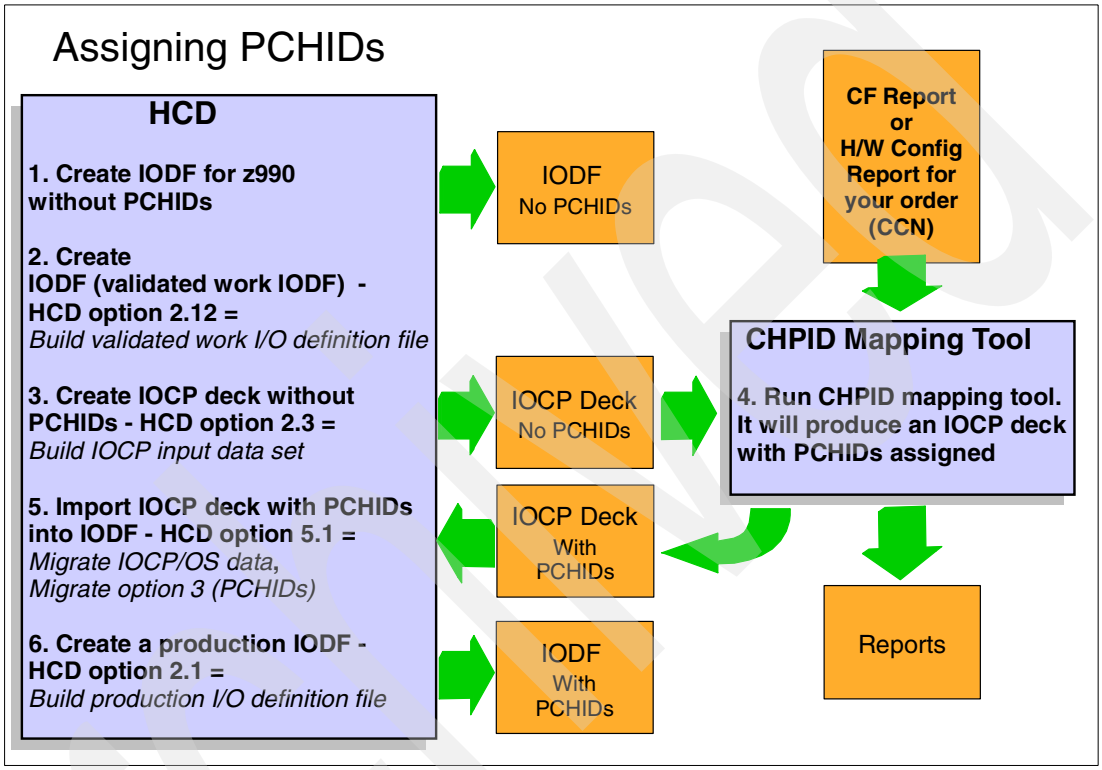


Figure 3-11 I/O configuration support

3.14.1 Mapping function for the z990

The z990 CHPID Mapping Tool provides a method of customizing the CHPID assignments for a z990 system to avoid attaching critical channel paths to single points of failure. It should be used after the machine order is placed and before the system is delivered for installation. The tool can also be used to remap CHPIDs after hardware upgrades that increase the number of channels.

The Mapping Tool can also be used during the system ordering process to ensure that sufficient physical I/O channels have been ordered.

The tool will map the CHPIDs from an IOCP file to Physical Channel Identifiers (PCHIDs) that are assigned to the I/O ports. These PCHID assignments are fixed and cannot be changed. The methods used for determining the PCHID assignments is described below. A list of PCHID assignments for each hardware configuration is provided in the PCHID report available when z990 hardware is ordered.

Unlike previous zSeries systems, there are no default CHPID assignments. CHPIDs can be mapped by importing the IOCP file into the z990 CHPID Mapping Tool. The IOCP file must be built for the new hardware order using HCD or IOCP that supports the z990.

There are two different methods for using the CHPID Mapping tool:

- The *manual method* can be used to define the relationships between each logical CHPID and physical ports on the machine. There is no availability checking, and the RAS characteristics of the resultant mapping is dependent on the user's knowledge of the machine availability characteristics.
- *Availability mapping* is the recommended method for mapping. This method allows you to input the IOCP deck for the proposed machine configuration and then define the order in which channel and control unit mapping should be done. This function takes into account the availability characteristics of the machine and will insure that the highest levels of availability will be achieved by assigning channel paths to avoid single points of failure in the processor.

While using the tool, you will have the ability to switch between manual and availability mapping. You could, for example, map your CHPIDs with the availability mapping option and then make changes manually.

3.14.2 PCHID assignments

The purpose of the CHPID Mapping Tool is to provide a method of assigning CHPIDs to I/O ports in a way that avoids attaching critical paths to a single point of failure. The first step is done by the z990 configurator, which assigns an identifier to each port that can later be associated with a CHPID. PCHIDs are assigned for you when your hardware is manufactured. You can find the description of PCHID assignments in “I/O interfaces and identification” on page 18.

As mentioned, 16 PCHIDs are assigned to each card slot. If a 16-port ESCON card were installed in card slot 1 of I/O cage 1, the first port would be assigned PCHID 100, the second port would be assigned PCHID 101, and so on. A two-port card would use the first two PCHIDs assigned to its slot and the rest of the 16 PCHID numbers for that slot would be unused.

3.14.3 CHPID Mapping Tool description

The CHPID Mapping Tool is a Java-based standalone application available on the IBM Resource Link™, and it must be downloaded to your personal computer for use. Once downloaded, you can make CHPID assignments without further Internet connections. You can make changes, save them (in your machine), and load them at a later time. You can import an IOCP file, assign PCHIDs, and export a new IOCP file. Refer to Figure 3-11 on page 59 for an overview of the external steps needed.

3.14.4 Mapping Tool requirements

A few things are required to access the CHPID Mapping Tool. First of all, you must have access to the WWW with a browser that is at least one of the following:

- Internet Explorer 5.0
- Netscape Navigator 4.7

Your browser must be configured with Java Script and cookies enabled.

You can access the Resource Link Web site at:

http://www.ibm.com/servers/resourcelink

Log in and select **Tools** and the **CHPID Mapping Tool**. There are several security measures in use by Resource Link that protect your hardware configuration data. We recommend that you check for new versions of the Mapping Tool at fairly frequent intervals during periods when you are actively using it.

3.14.5 Using the CHPID Mapping Tool

After downloading and installing the CHPID Mapping Tool, an icon will be available on your PC desktop. The first thing you must do to use the tool is to import your hardware configuration file that you downloaded from Resource Link.[16] To do this, select **Import H/W Config from file.** Use the **Look in:** function to find the location of the file and select **Open.**

The default configuration for your hardware order will now be available to the tool for Manual or Availability Remapping. For either Manual or Availability Mapping, you must also import a *validated* z990 IOCP file (see “Migrating the I/O configuration definitions with HCD” on page 98 for details). A validated IOCP file is one that is generated from a validated work IODF, or has been verified by the ICPIOCP program.

A CHPID Report is available from the “Reports” pull-down of the CHPID Mapping Tool. After you use the tool to assign CHPIDs, the tool will provide a new CHPID Report with the new values.

Using the tool for manual mapping

Manual mapping gives you the ability to assign each CHPID in your IOCP input to a physical slot and port (and its PCHID). When your configuration data file (or CF Report) is initially imported, your hardware configuration will be displayed with blank CHPID fields.

Availability mapping

The preferred method for using the CHPID Mapping Tool is *availability mapping*. You must provide a copy of the system's IOCP deck and define the priorities for channels and control units in the configuration. The CHPID Mapping Tool then decides how to assign CHPIDs to the I/O ports for a system that will provide maximum system I/O availability. Maximum availability is achieved by distributing channel paths to a control unit across different channel cards, STI links, MBAs, and books. In this way, a failure to one channel card or STI link will not affect the availability of a device.

Assigning priority

Priorities are assigned (from 0001 to 9999) for each control unit in the IOCP deck. More than one CU can be assigned the same priority. Assigning the same priority to more than one CU means that these units will be mapped together for availability. There are good reasons to do so:

- If one device serves as a backup for another device, you should assign the same priority to both of their control units.
- If multiple control units are used to provide multiple paths to devices, you should assign the same priority to these control units.

When the tool maps control units, it will assign their CHPIDs to ports, cards, STI links, MBAs, and books with emphasis on availability.

[16] An alternative is to use the CFR file produced by the IBM configurator. This configurator is used by IBM and IBM business partners to create a z990 order.

Once you have decided which control units should be mapped as a group, you should assign the lowest number priority to the control units in the group that you want to be mapped first. The tool will map the CUs with priority 0001 first. It is advantageous to leave gaps in your group numbering (0010, 0020, 0030, etc., for example). This allows room for additional groups to be created in the event of a conflict. Any CUs that do not have a priority will be assigned *afterwards* in control unit number order; that is, a control unit with CUNUMBR of 1000 will take priority over one with a CUNUMBR of 1100. Once priorities have been entered, select **Process CU Priority** from the Tool menu. When the tool completes processing, it will display a list of port assignments that were not included in the IOCP deck.

Intersects

The tool displays *intersects* on the next panel. Intersects are potential availability problems detected by the tool. Reason codes for intersects are provided and an explanation of the codes are displayed at the bottom of the panel. Remember that these codes refer to channels *in the same group*, where this *group* is defined solely by the same priority codes to the Mapping Tool. Briefly, the warnings are:

- Intersect **C:** Two or more assigned channels use the same channel card.
- Intersect **S:** Greater than half the assigned channels use the same STI.
- Intersect **M:** All the assigned channels are supported by the same MBA.
- Intersect **B:** All the assigned channels are supported by the same book.

These should be considered informational messages and not errors. Eventually, the tool will run out of highly available places to assign CHPIDs and will resort to plugging more than one control unit on the same channel card, STI or MBA group. A possible cause for intersects may be that prior mapping to other groups left a small number of unassigned ports.

Intersects might be corrected by assigning the group that displays an intersect to a lower numbered priority, or by dividing the group into smaller groups when possible. If you find an intersect to be unacceptable and regrouping and re-prioritizing does not resolve it, you may need more hardware.

One approach to intersect warnings is to ask your account team to change the machine order, create new CF Reports, and rerun the Mapping Tool. Then continue this process until all warnings are resolved.

3.15 RAS

The design for Reliability, Availability, and Serviceability (RAS) has always been a strong point of the IBM S/390, zSeries, and the associated operating systems. There is so much background information involved, such as the many layers of recovery in z/OS and the interfaces between hardware recovery and software support of this recovery, that we will not attempt to summarize it all here. This section concentrates solely on differences between the z990 and earlier zSeries machines.

The availability element of RAS has become more important with the z990. Typically, a single z990 may replace several older systems. Having several systems is convenient when it is necessary to take a system down; with appropriate planning, the other systems continue operation. With a single system, this becomes more difficult to achieve. However, the z990 has more ways to service and reconfigure the system, without disrupting operation, than in previous systems.

Nondisruptive changes and service

There is considerable discussion about nondisruptive functions for the z990. This is sufficiently important that we need to consider what is meant by *nondisruptive*. It is not a

precisely defined term and the exact meaning may depend on the context. Consider the following examples:

- A PU failure and sparing action is usually nondisruptive. In this case the whole event is transparent to all LPARS, users, and operations. The number of spare PUs available is reduced.
- Power supply modules can be replaced nondisruptively. There is no effect on any LPAR, user, or operation.
- An I/O adapter can be added nondisruptively (provided that a new I/O cage is not needed), without any effect on any LPAR, user, or operation. However, in order to *use* the new I/O adapter, it may be necessary to issue dynamic I/O changes from several z/OS instances (to dynamically update the HSA that corresponds to the active IOCDS) and make the new I/O adapter known to each z/OS. If an operating system instance cannot be dynamically changed for some reason, it may be necessary to IPL that LPAR. This is disruptive to that LPAR, but not to the rest of the system.
- A refrigeration system failure may cause the system to slow down a little. This is nondisruptive, assuming that excess capacity exists or that workload can be shed to match the slowdown. Refrigeration repair and enablement is nondisruptive and the system will typically return to normal operating speed automatically. (Operator action may be needed to enable more workload.)
- Processors (CPs) can be added nondisruptively (if unused PUs exist in the installed books), but LPAR definition changes may be needed to use the additional CPs. Some LPAR definition elements can be changed dynamically, but it will probably be necessary to IPL the operating systems in the changed LPARs. This is nondisruptive to the unaffected LPARs.
- Major memory reassignments (due to substantial memory upgrades, for example) may require a POR. This is completely disruptive for a relatively short time, but can typically be scheduled for a convenient time.
- Adding memory (by changing the memory cards in installed books) is disruptive and the entire system must be down. However, this is an *elective* change and can be scheduled for an appropriate time.
- Removing a book requires that the system be stopped. This is disruptive for a longer period. The timing may or may not be convenient, depending on the reason for removing the book.

These are a few examples of various degrees of nondisruptive and disruptive functions. The z990 is a complex system, especially when all the potential software interactions are considered, and there are many combinations of events and actions that must be considered. IBM has made significant advances in RAS functions with the z990. Some RAS elements are purely hardware (such as power supply service), while others involve interaction with operating systems, systems programmers, and system operators.

RAS discussions should involve the necessary levels of detail without losing sight of the complete picture. In a relatively stable environment, with appropriate planning and training, software selection and maintenance, I/O redundancy, and power redundancy, a z990 system may *never* have unwanted hardware outages throughout the typical lifetime of the system.

Power supplies and flexible controllers

As with earlier S/390 and zSeries systems, all power supplies are redundant. That is, there are two of everything and the system can tolerate a failure of any power supply. There are even two power supplies per book (and eight book power supplies in a system with four books). Power supplies can be removed and replaced nondisruptively.

Each I/O cage has two power supplies for redundancy. One can be removed, repaired, and replaced without affecting system operation.

The power supplies installed with books and I/O controllers each have a *Flexible Support Processor* (FSP) that is a primary element of overall system control. Each FSP has access to both internal LANs used for communication between the Support Elements and the cage controllers.

Power

The internal battery feature (IBF) is useful for riding over short power problems. Of course, its usefulness is questionable if critical I/O devices do not have the same protection.

As with other recent zSeries (and earlier) machines, there are two external power connections and either is sufficient to run the system. This is meaningful only if connections are through different power feeders. This typically requires the cooperation of the local power utility and may require minor construction work to bring the separate feeder into the system area.

The z990 uses three-phase power but can continue to operate if power on one of the phases fails. The z990 can accept a wide range of voltages on the power feeders.

Adding books

A new or repaired book may be added without disrupting normal operation.[17] With proper advance planning (for the definition of inactive LPARs with appropriate memory and processors, for example), the resources added through the new book can be used immediately. This is a very powerful capability. A z990 system can be scaled upward to many times its original capabilities without disrupting operation. No previous IBM system has had such extensive hardware scaling potential without operational disruption.

Adding memory

It is possible to have more usable memory installed in a system than is enabled for customer use. This additional memory may be enabled through the IBM MES process. Again, advance planning is needed to use the additional memory nondisruptively.

Memory may be added by using more of the installed memory (if possible) or by moving to the next z990 model. Upgrading to the next model can be nondisruptive, although advance planning is needed in order to use the additional elements nondisruptively. Different books may have different amounts of memory installed. (The two memory cards in a particular book must have the same effective memory size.)

Adding I/O adapters

All I/O adapter types may be added nondisruptively, provided a new I/O cage is not needed. It is usually necessary to make IOCDS changes to use the new adapters. With sufficient planning, this can be a dynamic IOCDS change that does not involve a POR or IPL.

Changing LPAR definitions

The attributes of an LPAR that can be changed fall into three distinct categories:

- Those that can be changed dynamically
- Those that require an LPAR deactivate/activate to take effect
- Those that require an IOCDS change and a Power-on Reset to take effect

[17] This capability is planned to begin October 31, 2003. Before this time, adding a book is a disruptive function.

The controls for LPAR weights and capping can be changed dynamically. The maximum storage allocation, the maximum number of shared/dedicated CPs, and the number of reserved CPs can only be changed with an LPAR deactivate/activate. Addition of new LPARs can only be done with a Power-on Reset.

These restrictions can be largely avoided with careful planning. For instance, you may wish to define some dummy LPARs as “place-holders” in case new LPARs are needed in the future. (A good LPAR naming convention will help here, since partition names cannot be changed without a Power-on Reset and a non-trivial IOCP change). Partitions should be defined with reserved engines if they are likely to need more capacity than their logical engines can deliver before their next scheduled outage. You should also consider defining them with some (offline) reserved central (and/or expanded storage if 31-bit operation is needed) to accommodate storage growth before their next scheduled outage.

Careful planning may be needed for dummy LPARs to ensure that they will have access to the desired CHPIDs when the LPARs are activated, and will not have access to unwanted CHPIDs.

PU sparing

Every processor book, when installed, contains two spare PUs. A z990 book contains a mixture of single-core PU chips and dual-core PU chips, as needed to populate the MCM designed for the z990. If a PU on a dual-core chip fails, *both* PUs on that chip are taken offline.

A PU failure in a single-core chip will cause a spare PU to be used. A PU failure in a dual-core chip will cause two spare PUs to be used. Spare PUs are usable in a system-wide manner when multiple books are installed. For example, if book 0 needs another spare PU and has none left, then a spare from another book will be automatically used.

A PU failure is usually transparent to application programs. A combined hardware and software recovery action (not new with the z990) can usually transfer the state (register contents and so forth) of the failing PU to another PU and continue application execution.

Memory sparing

The z990 does not contain spare memory DIMMs. Instead it has redundant memory distributed throughout its operational memory and these are used to bypass failing memory. Replacing memory cards requires the removal of a book and this is disruptive. The extensive use of redundant elements in the operational memory greatly minimizes the possibility of a failure that requires memory card replacement.

Internal memory elements

Internal functions, such as cache, dynamic look-aside tables (DLAT), branch history tables (BHT) and so forth involve their own memory elements. These have a variety of recovery designs. The most common is, after appropriate retries, to delete the memory row (element) that has the error. In the case of cache, the failing addresses can be relocated to spare memory. These actions are all transparent to LPARS and normal operations.

In addition, main memory has a *scrubbing* function that is always active.

Storage protection keys are located in separate memory, with three copies of the data. A voting process is used in the case of a mismatch of data among the three copies.

Oscillator and ETR cards

Two oscillator and two External Time Reference (ETR) cards are in the system. An ETR card failure will automatically switch to the other ETR card; the effectiveness of this action depends

on proper planning of the external ETR connections. An oscillator card failure will stop the system. A subsequent POR will cause a switch to the alternate oscillator card and the failed card can be replaced without additional disruption.

Ring failure

The ring that connects L2 cache in the books is critical to system operation. It is a two-way ring and can survive a failure in one direction of the ring. If one direction of the ring fails in a system with fewer than four books, the system performance will degrade but all LPARs continue operation. The rings are simple elements, in hardware terms, and failures should be exceedingly rare.

Support elements and Hardware Management Consoles

A z990 always includes two Support Elements (SEs). These provide redundant coverage and the alternate unit can be used to preload new control code changes. The number of hardware management consoles (HMCs) is up to the customer, but it is only sensible to have at least two. SEs and HMCs can be rebooted during normal system operation. That is, they are needed only for exceptional system events or for operator interaction.

Hardware storage area

The HSA is located in book 0. The HSA cannot be dynamically moved to memory in another book.

3.16 LPARs

As announced, the z990 can have up to 30 LPARs[18], with a statement of direction for 60 LPARs. The key element here is that the previous architectural limit of 15 (which used a 4-bit field in various firmware and software control blocks) has been extended. The new architectural functions use a one-byte field. There a few additional changes that may be important:

- A z990 runs only in LPAR mode. Basic mode (without LPARs) no longer exists.
- A maximum of 15 LPARs may be associated with a single Logical Channel Subsystem.
- Initial z990 shipments support only 15 LPARs. Compatibility support works only with LPARs 1 - 15, and only in LCSS 0.
- Partition identifiers, names and numbers are a little more complex, as explained in this section.

There are several identifiers associated with an LPAR:

- **LPAR identifier**. This is a number in the range 0 - X'3F' and is assigned by the user when defining LPAR Image profiles through the SE or HMC. It is unique across the z990 processor. This identifier is returned by an STIDP or STSI instruction. This identifier is also known as the user logical partition id (UPID).
- **MIF Image id**.[19] This number is defined through HCD or IOCP and is the partition number defined in the RESOURCE statement in an IOCP source statement. It is in the range 1 to X'F' and is unique *within* an LCSS. It does not need to be unique within a z990 system. This id is also known as the IID.

[18] Up to 15 LPARs are available at the time of initial z990 shipments. As announced, this will increase to 30 LPARs on October 31, 2003.

[19] MIF is Multiple Image Facility. It was formerly known as EMIF for ESCON Multiple Image Facility because it was introduced for ESCON channels. The function is more general now, so the "E" portion of the name has been dropped.

- **LPAR name**. This name is defined through HCD or an IOCP and is the partition name in the RESOURCE statement in an IOCP source statement deck. An LPAR name must be unique across the z990 system.

The LPAR identifier is used when defining activation profiles. The HCD field for the MIF Image has the label “Partition Number” and this is equivalent to the MIF Image id used in IOCP RESOURCE statements. We suggest you establish a naming convention for the LPAR identifier. For example, you could use the LCSS number concatenated to the MIF id. LPAR id 1A could mean LCSS 1, MIF id A. This fits within the allowed range of LPAR ids and conveys useful information to the user.

3.17 STIDP instruction changes

The results of the STIDP (Store CPU ID) instruction is different on the z990 than on earlier systems. The logical CPU address is absent from the results, and the LPAR identifier is now a full byte. This is illustrated in Figure 3-12 on page 67.

In the new format, the combination of the LPAR number (8 bits) and the extracted portion of the system serial number (16 bits) provide a 24-bit value that will be unique across all z990 machines and their LPARs.

A program requiring the logical CPU address can use an STSI instruction (instead of the STIDP instruction) to obtain it, although the STAP instruction is the recommended method.

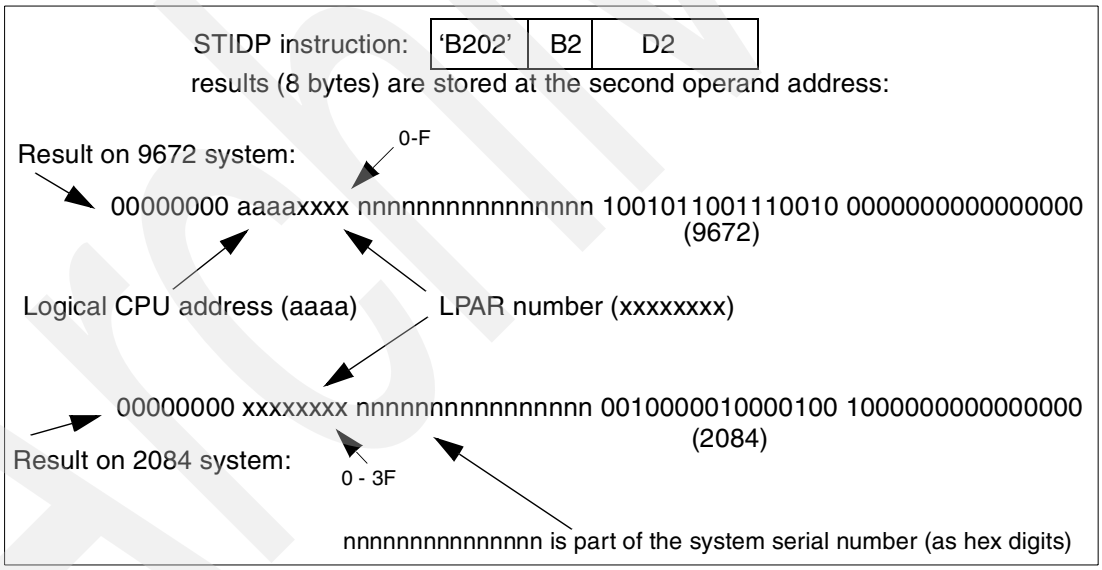


Figure 3-12 Changes to STIDP instruction results

3.18 OSA-Express adapters

The OSA-Express adapters available for the z990 are the same as for the z900, with the OSA-Express Fast Ethernet adapter being replaced by the OSA-Express 1000BaseT adapter. The older OSA-2 adapters available on a z900 are not available on a z990.[20] A maximum of 24 OSA-Express cards per system may be used. Each card provides two ports, each port has a PCHID identifier, and each port requires a CHPID number in order to be used. This differs from the earlier OSA-2 adapters, where one CHPID was used by the adapter and three more were blocked. No CHPIDs are blocked by OSA-Express cards.

The OSA-Express adapters can generally run at *line speeds*. This was not the case for earlier LAN interfaces, and is a major improvement in capability. It has been common practice to use ESCON channels to connect to external LAN routers in order to reach the nominal line speeds for LAN connections. With the OSA Express cards, this is not required, and this provides a number of advantages:

- ESCON-attached routers are expensive. Eliminating them provides a direct savings.
- External routers can be complex and often require unique personnel training (or contract services). Eliminating them can remove the need for an expensive skill, and remove a potential failure point in your system.
- External routers can require another level of IP routing. Eliminating this simplifies your logical routing structure.

Note that the 1000BaseT, Fast Ethernet, and token ring ports connect to copper cables, while the gigabit Ethernet ports connect to fiber cables.

The OSA-Express cards (and the FICON Express card) have indicator lights that may be useful during installation and for problem determination.

Note that the OSA-Express 155 ATM adapter is not supported on the z990.

3.18.1 OSA-Express Fast Ethernet

This adapter can be carried forward from a z900 system, but cannot be ordered as a new adapter for the z990. Instead, new orders will use the 1000BaseT adapter.

The OSA Express Fast Ethernet card has two independent ports (channels) and uses traditional copper wiring with an RJ-45 connector. The ports provide 10 or 100 Mbps Ethernet, with auto-negotiation of the speed used. A QDIO interface[21] may be used for TCP/IP. A non-QDIO interface may be used for SNA (including APPN and HPR) and TCP/IP. By default, the adapter automatically adapts to 10 or 100 Mbps operation, and to half or full duplex operation. You can set these options (to avoid the automatic selection) using Support Element panels or the OSA/SF program. Automatic operation can be a problem when used with a very lightly-loaded network with not enough activity for the adapter to properly sense.

3.18.2 OSA-Express 1000BaseT Ethernet

This adapter has two independent ports (channels) and provides gigabit Ethernet (as well as 100 Mbps and 10 Mbps Ethernet) over copper cables. It replaces the OSA-Express Fast Ethernet adapter, and complements the OSA-Express Gigabit Ethernet adapter (which uses fiber cables).

The OSA Express Fast Ethernet card uses traditional copper wiring with an RJ-45 connector. The ports provide 10, 100, or 1000 Mbps Ethernet, with auto-negotiation of the speed used. A QDIO interface may be used for TCP/IP. A non-QDIO interface may be used for SNA (including APPN and HPR) and TCP/IP. By default, the adapter automatically adapts to 10, 100, or 1000 Mbps operation, and to half or full duplex operation.

[20] These are located in the compatibility I/O cage of a z900. This I/O cage is not available for the z990.

[21] This is an alternative to the standard I/O interface using SSCH commands with traditional CCWs. QDIO functions have been available for some time with the z900 machines, but may not be familiar to installations with earlier systems. QDIO protocols are especially efficient for LAN interfaces. Unique operating system code is required to use QDIO, and this support exists in z/OS and other operating systems.

3.18.3 OSA-Express Gigabit Ethernet

There are two generations of this adapter. The older generation may be carried forward from a z900 system. The newer generation is provided for new orders. The primary difference is the connectors. The older generation uses SC Duplex fiber connectors and the newer generation uses LC Duplex fiber connectors.

These adapters have two independent ports (channels). Each adapter is available in two versions. The LX (long wave length) version uses single mode fiber with an SC or LC Duplex connector. The SX (short wave length) version uses multimode fiber with an SC or LC Duplex connector. The LX version can use multimode fiber for shorter distances if MCP cables are added at each end of the multimode cable.

This adapter uses only QDIO, and is used only for TCP/IP. That is, SNA is not supported through the Gigabit Ethernet card. However, Enterprise Extender support can be used to send SNA traffic over IP.[22] QDIO mode on Gigabit Ethernet on OS/390 or z/OS requires Release 7 or later of Communications Server. Gigabit Ethernet always operates in full duplex mode.

3.18.4 OSA-Express High Speed Token Ring

This adapter has two ports (channels) and each operates independently at 4, 16, or 100 Mbps. The ports automatically adjust to the correct speed using auto-sense and auto-negotiation functions. It uses traditional copper wire connections with an RJ-45 connector. Each adapter port operates in either half or full duplex, depending on the speed. This adapter operates in both QDIO mode (TCP/IP traffic only) and non-QDIO mode (TCP/IP and SNA, APPN, HPR). Implementing QDIO on an OSA-Express token ring card requires Release 10 (or later) of Communications Server for OS/390. QDIO mode for Token Ring on Linux requires Linux kernel V2.4 or later. This card replaces the OSA-2 card used in earlier systems and also provides the higher speed option.

3.18.5 OSA-Express migration

Users migrating from older S/390 machines often find OSA-Express concepts to be quite different from previous LAN interface hardware. For example, OSA-Express adapters combine the traditional channels, control units, and device concepts into a single element—the OSA-Express adapter. The OSA-Express adapter is two channels (one for each port). Furthermore, in some modes of operation, a single OSA Express adapter port can appear as multiple independent ports that can be used by multiple LPARs.

A full description of OSA adapters is beyond the scope of this redbook. We strongly recommend that you consult the IBM Redbook *OSA-Express Implementation Guide,* SG24-5948-01 or later.

A configuration program, known as OSA/SF, is *sometimes* needed to customize an OSA-Express adapter. Earlier versions of this program (used with earlier versions of OSA adapters) required some effort to understand and use. The current OSA-Express adapters, when used for TCP/IP traffic, have greatly reduced the need to use OSA/SF. Also, newer versions of the OSA/SF program can be used at a workstation and provide easier-to-use GUI interfaces. In addition to adapter configuration, OSA/SF can display very useful statistics about LAN usage.

[22] This requires matching Enterprise Extender software at the “other end” of the connection, of course.

Any 155 ATM or FDDI adapters must be replaced with different modes of connections, such as an Ethernet or token ring link to a switch or router that provides the ATM or FDDI interfaces.

Table 3-3 LAN interface summary

	CHPID type	SNA/APPN/HPR	TCP/IP	OSA/SF needed
Gigabit Ethernet	OSD (QDIO)	No[a]	Yes	No
Fast Ethernet	OSD (QDIO) OSE (non-QDIO)	No[a] Yes	Yes Yes	No Yes
1000BaseT	OSD (QDIO) OSE (non-QDIO)	No[a] Yes	Yes Yes	No Yes
Token Ring	OSD (QDIO) OSE (non-QDIO)	No[a] Yes	Yes Yes	No Yes[b]

a. However, this support can be provided by the Enterprise Extender function.
b. OSA-Express Token Ring and Ethernet requires OSA/SF for non-QDIO except for when it uses the default OAT without port sharing.

3.19 z/VM functions

The z990, working with z/VM V4R4, provides adapter interruption functions. This is a low overhead design that applies to FICON Express when used in FCP mode and to all OSA-Express adapters used in QDIO mode. A performance assist for virtualization of adapter interrupts allows z/VM to passthrough interrupts to V=V guests.

3.20 Parallel channel planning

z990 systems do not have parallel channels.[23] If you have parallel channel devices, you need to do some planning. Plans typically involve the following choices:

- Retire the devices.[24] This is not as trivial as it may sound. Parallel channel tape and DASD devices are probably quite old and ready for retirement. Maintenance costs may exceed the cost of replacement devices using ESCON or FICON channels, and newer DASD devices have far greater capacity and performance.
- Purchase converter boxes to connect parallel devices to ESCON channels. IBM made such converters some time ago (IBM 9034, often known as Pacer units), but these are no longer manufactured. IBM recommends the use of the Optica product, described in 3.20.2, “Optica planning” on page 71. The IBM 9034 units may also be used, but these cannot be ordered through normal IBM ordering channels.
- Keep only selected parallel devices and consolidate them onto as few channels as possible.

The most typical parallel devices that installations might want to keep include:

- IBM 34xx tape drives (“round tapes”), kept for archival functions
- IBM 3174 control units, used for 3270 operating system consoles and TSO/CICS® 3270 terminals
- Various line printers

[23] Parallel channels are often known as “bus and tag” or OEMI channels, named for the functions of the two cables used for these channels.
[24] We use the term *devices* here to include the associated control units. It is the control units, of course, that actually use the channels.

There are often valid reasons for keeping these devices. However, we suggest that they can be consolidated onto a reduced number of channels. For example, a single channel could probably handle a mixture of all of the devices mentioned here. You might not normally mix tape drives with other devices on the same channel, but this depends on how often the tape drives are used. Typically, the older drives are so rarely used that their channel sharing characteristics can be ignored.

Most users have migrated away from “real coax-attached” 3270 terminals and the use of 3174s has declined. Also, increasing use of many LPARs makes the use of IBM 2074 units more attractive than using a large number of 3174s. The non-SNA IBM 3174s (including ESCON models) are used for z/OS operator consoles, and one 3174 is needed for each LPAR because 3174s cannot be shared by multiple LPARs. An IBM 2074 unit can be shared by multiple LPARs and connects to TN3270 client sessions on PCs, while appearing as multiple local, non-SNA 3174s to the operating system.

3.20.1 Byte multiplexor

A few parallel devices may *require* byte multiplexor channels.[25] Typically, there are no modern direct replacements for these devices. Few customers still use these devices, and each might be considered as a special case. Possibly EP and PEP lines are the most common of these.[26] The ESCON-to-parallel converter units may support such devices; you should check with the manufacturer for the latest information. The older IBM 9034 units (sometimes known as Pacer units) do support byte multiplexor modes, but may not have been tested with your particular device types. Again, we suggest you consider these as special cases and discuss them with your marketing representatives.

A 9034 unit used with byte multiplexor control units must have serial number 41-53345 or higher and must contain a logic card with part number 42F8047. If use of older 9034 units is required, RPQ 8P1767 should be investigated.

3.20.2 Optica planning

Optica Technologies, Incorporated, is a company based in Ohio that designs and manufactures a range of connectivity products. More information can be found at:

```
www.OpticaTech.com
```

The product we discuss here is the 34600 FXBT ESCON Converter, which we refer to as the *converter*. It converts an ESCON channel to a parallel channel. Optica produced two earlier converters, including one quite similar to IBM’s Pacer unit. The converter described here is new and has been designed for z990 systems. It has been tested with early z990 machines.

IBM recommends these converters for connecting older parallel channel control units to z990 ESCON channels. Optica is not part of IBM. It is a separate company and you must order the converters directly from Optica.[27] IBM is not involved in the ordering process. You should place orders in a timely manner so that the converters are available when your z990 is delivered.

[25] Other devices were traditionally used with byte multiplexor channels, but can also be used with block multiplexor channels.
[26] If you do not recognize these names, you need not worry about them.
[27] IBM may remarket the Optica converter in some countries, and these statements must be modified accordingly.

The converter is a small box, about 7.5 inches wide, 2 inches high, and 12 inches deep (19 cm x 5 cm x 30 cm) and connects to a utility power outlet (100v - 240 v, 50/60 Hz). Each converter handles one ESCON channel and one parallel control unit connection.[28] The converters may be individually positioned (most common) or rack mounted (not so common). Rack mounting consists of a cage that fits a standard 19-inch rack and holds 8 converters. Each unit requires its own power connection in the rack.[29]

The most common location for the converters is *under* a raised floor. They are rugged units that do not need to be monitored. Each converter has an internal fan, but the unit can function indefinitely without the fan if there is a reasonable cool air flow—as there usually is under a raised floor. A typical z990 installation might have several converters placed under the raised floor, in convenient locations to connect to existing bus and tag cables. New ESCON cables would be run from the converters to the z990. The only other requirement would be for utility power outlets under the floor, to run the converters.

Figure 3-13 illustrates the conceptual design of the converter. The FIFO elements are "bit buffers" and the channel interfaces include the appropriate transmitters and receivers. Internal control is by an IBM PowerPC® microprocessor core. This is used to program the gate array. Normal channel data movement is completely handled in the gate array, permitting full- speed channel operation (200 Mbit/sec ESCON and 4.5 MBytes/sec parallel). The microprocessor interprets channel commands and provides the logic for status conversion and so forth.

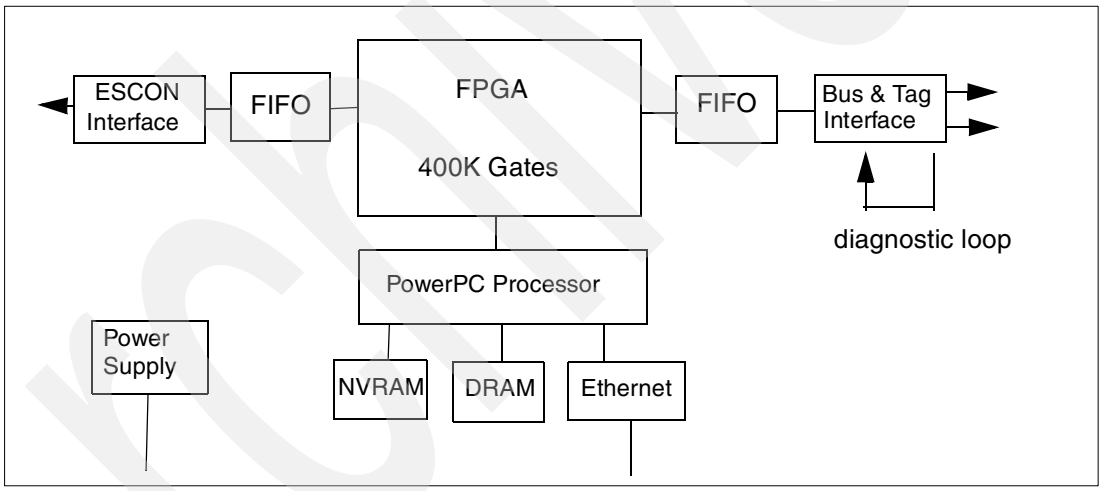


Figure 3-13 Conceptual design of Optica converter

The converter has extensive internal diagnostics. Some are run at power on, others are run when connected to an external ASCII Telnet session via the Ethernet adapter. This Ethernet connection is not intended for connection to a public LAN. (All converters are supplied with the same IP address, although it can be changed.) In the rare case when diagnostic work is needed, a laptop PC is connected to the converter Ethernet port (using a crossover cable) and the PowerPC program (in the converter) provides a menu-driven set of diagnostic functions. The parallel channel interface can be looped internally under control of the diagnostics. An external plug is needed to loop the ESCON interface.

[28] Control units may be daisy-chained, as usual, with the same restrictions that exist when using "real" parallel channels.
[29] The same unit is used for individual positioning or rack mounting. Rack mounting causes two small tabs to be added to the converter case; these secure the converter to the cage in the rack.

If an ESCON loop plug is inserted, the converter automatically runs an extensive set of internal diagnostics. Results are shown in the LEDs and 4-character display on the panel. No Ethernet connection is needed for this operation. (The *CU busy* LED on the panel may be interesting during normal operation, and a disadvantage of installing the unit under the floor is that you cannot see the panel.)

The converter returns logout information to the ESCON channel if internal errors are detected during normal operation. Service is by complete replacement of the unit. Optica provides fast service turnaround and loaner units where appropriate.[30] Larger installations with a considerable number of these converters might want to purchase a spare. Optica indicates that failures of earlier units were very rare and a spare is probably unnecessary in most cases.

The Optica units support byte multiplexor operation. However, testing of this mode is limited by their difficulty in finding appropriate working byte multiplexor devices for testing purposes. If you need byte multiplexor operation, you might want to discuss your configuration with Optica.

Recent information indicates the following control units and devices can be supported. You should contact Optica for the latest information.

Control Unit	Devices	Notes
3880-3,-13,-23	3380	DS mode, 900 m fiber
3990-1,-2,-3	3380, 3380-CJ2	DS mode, 1200 m fiber
3990-2,-3	3390	DS mode, 1200 m fiber
2440	2440	DCI mode, 3000 m fiber
3803-1	3420-3,-5,-7	DCI mode, 3000 m fiber
3803-2	3420-4,-6	DCI mode, 3000 m fiber
3803-2	3420-8	DCI mode, 2800 m fiber
3480	3480, 3490	HST,DS modes, 3000 m fiber
3174-x1L	(many)	DCI, HST, DS modes, 3000 m fiber
3274 (A,B,D)	(many)	DCI mode, 3000 m fiber
5088	(5081, etc)	HST, DS modes, 3000 m fiber
6098	(various)	HST, DS modes, 3000 m fiber
3172	(LAN, TP)	DCI, DS modes, 3000 m fiver
3720, 3725	(TP, LAN)	DCI mode, 3000 m fiber
3745	(TP, LAN)	DCI, DS modes, 3000 m fiber
8283		DCI, DS modes, 3000 m fiber
(3262-5, 3800-1, 3820-1, 4245, 4248,6262)		DCI mode, 3000 m fiber
(3800-3,-6,-8, 3825, 3827, 3835)		HST, DS modes, 3000 m fiber
3088	CTC	HST, DS modes, 3000 m fiber
9032, 0933		see Optica specification sheet
3814		
3848 crypto		DCI, DS modes, 3000 m fiber
3890™-XP		DCI mode, 3000 m fiber
3897/3898		DCI, DS modes, 3000 m fiber
4753		DCI, DS modes, 3000 m fiber

In this list, the mode notation is:

```
DCI = Direct Current Interlock (single tag mode)
HST = High-Speed Transfer (DCI alternate tag mode)
DS = Data streaming
```

The maximum ESCON fiber distances mentioned (usually 3000 m) must be reduced by 200 m for every ESCON Director in the path. Other limitations exist for specific devices; you should verify your intended use with Optica.

[30] IBM may offer service for Optica converters in some countries.

A converter can be placed on the output side of an ESCON Director, between the director and a control unit. Optica documentation shows this configuration only for IBM 9032-005 directors; for any other model we suggest you contact Optica for more information. When an ESCON Director is used in the path, it must be configured as a dedicated connection.

Each converter includes a cable for bus and tag connections. This cable has a single large D shell on one end and splits into two cables with standard parallel channel connectors. No ESCON cables are included.

Optica Technologies Incorporated can be reached at:

```
700 Pleasant Valley Drive
PO Box 848
Springboro, Ohio 45066-0848
Telephone: (937) 704-0100
Fax: (937) 704-0101
```

3.21 IBM 2074 utilization

Traditional S/390 operating systems (meaning everything except Linux) use 3270 terminals for customization, operation, and a basic user interface. For practical purposes you need locally-attached, non-SNA 3270 devices for operator consoles and initial TSO or CMS sessions. Using these TSO or CMS sessions, you can further customize the system for connection to SNA networks and TCP/IP networks. You can connect 3270 sessions two ways: through an IBM 3174 control unit, or through an IBM 2074 control unit. The 3174 is no longer manufactured, although many are in use in existing S/390 installations. The 2074 is currently manufactured and marketed.

The 3174 control unit connects to “real” 3270 terminals, via coax cable. The 2074 connects to TN3270e clients, via LANs. The 2074 converts the TN3270e sessions such that they appear to originate in a locally-attached 3174 connected to “real” 3270 terminals. See the IBM Redbook *Introducing the IBM 2074 Control Unit*, SG24-5966, for more details.

A very minimal 2074 setup, used with two LCSSs, is sketched in Figure 3-14 on page 75. The 2074 is connected to the z990 with ESCON channels. (The 2074 can have one or two ESCON channel connections; we used one for each LCSS.) The 2074 provides two emulated (TN3270e) sessions on its own display console and four LAN connections (two Ethernet and two token ring) to connect to clients having additional TN3270e sessions. The PCs shown typically use IBM’s PCOM product[31] as the TN3270E clients.

If the 2074 is directly connected to one ESCON channel, that channel can be used with only one z990 LCSS. A 2074 with two ESCON channels can be directly connected to two LCSSs. If the 2074 is connected through an ESCON director, the connection possibilities are more complex although a single ESCON channel on the 2074 could be used by both LCSSs.

[31] The full product name is eNetwork Personal Communications.

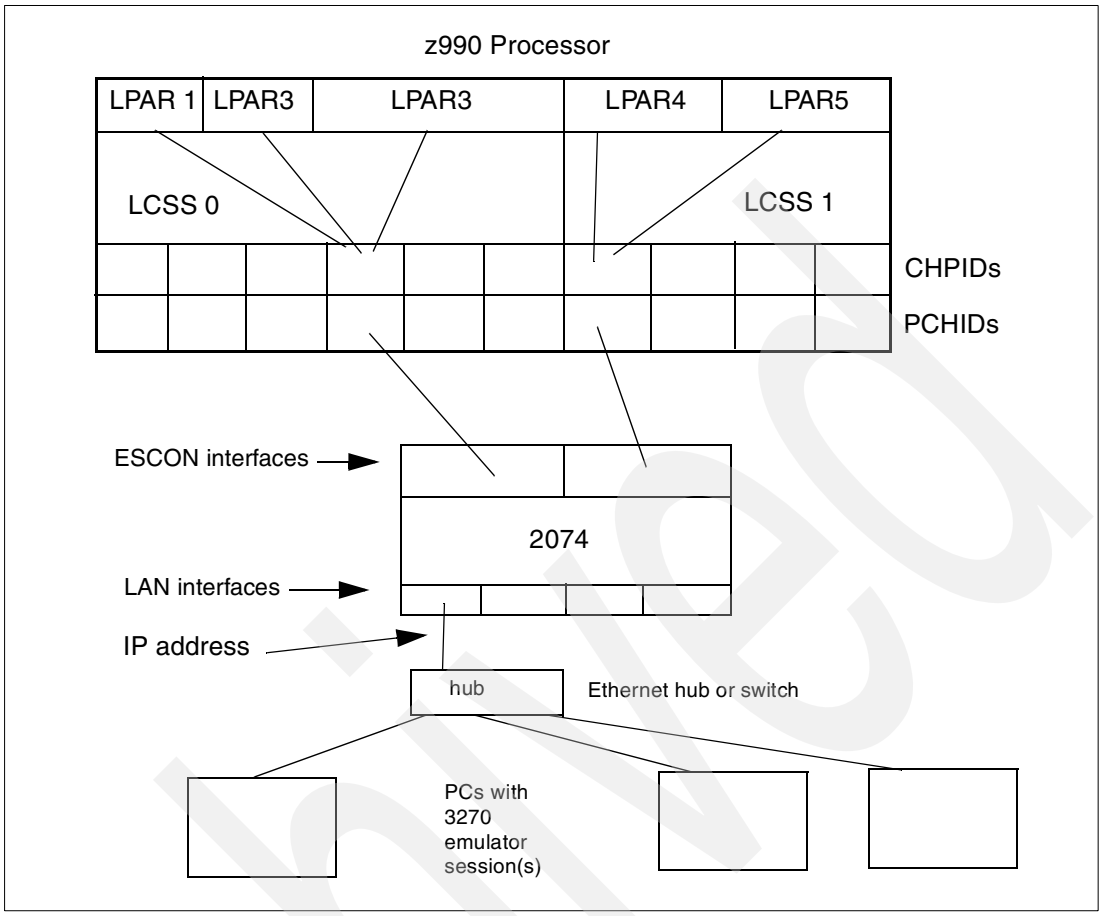


Figure 3-14 ITSO 2074 and LAN connections

Do not confuse the single LAN connection (in the illustration) with a single path to the processor. There are a number of ways to customize a 2074, but a typical setup might be this (assuming the use of a 2074 with two ESCON adapters, as illustrated):

- Each PC might have multiple TN3270e client sessions and these are independent of each other. Three sessions on a single PC is logically equivalent to a single session on each of three PCs.
- Each client session has two key parameters for its initial connection to the 2074:
  - A port number for the 2074 logical interface. The 2074 will have one port number for each ESCON adapter. (The default port numbers are 3270 and 3271.) By specifying the appropriate port number, the client indicates with which 2074 ESCON interface his client session is associated. There are two largely independent sets of definitions in the 2074, one for each ESCON interface.
  - An LU name. The LU name field was originally for SNA protocols, but it is used for a different purpose here. The client LU name should match an LU name preconfigured in the 2074 by the system administrator.
- The LAN interfaces are common to all client connections of the 2074. There are multiple LAN interfaces so as to work with both Ethernet and token ring. There are two of each to permit connections to multiple LANs for increased availability.
- Each LAN interface must have a different IP address, assigned by the user. The two Ethernet LAN interfaces, for example, are not redundant adapters with the same IP address.

For example, a TN3270e client might establish a session with the IP address of the 2074, port 3270 (which selects the first ESCON channel), with LU name AAAA. The 2074 could be

configured such that LU name AAAA is equated to CU 1, UA 3 and this (via IOCDS definitions) is device number 700 for LPAR 2 and this might be an MVS operator console. LU name BBB might, through a different series of definitions, be a TSO session for LPAR1.

The IBM 2074 is a very flexible device and this flexibility can be confusing initially. We recommend the redbook mentioned earlier, *Introducing the IBM 2074 Control Unit,* SG24-5966, for more information.

The only element that is new for the z990 is that the two ESCON channels might go to different Logical Channel Subsystems on one processor.

3.22 Cable ordering

Fiber optic connectivity has changed considerably over the past decade, driven by new application requirements for bandwidth and connectivity. Fiber optic cabling technology has kept pace with this demand. New standards have evolved, new fiber optic products have been developed, and the number of cabling options has expanded. The result of this technology expansion is that several hundred different feature codes would be needed to specify fiber cables for a large system. The product configurators can no longer predict the exact cable needed and substantial planning is needed for the installation or migration of a high-end processor. As a result, fiber optic cables can no longer be ordered by feature codes.

Fiber optic cables can be obtained, along with the necessary planning and installation, by ordering a service. IBM Global Services (IGS) has developed a series of options to assist the customer with their cabling responsibilities. Two types of services are available under this offering: zSeries fiber cabling service and enterprise fiber cabling service. zSeries fiber cabling service contains three options to provide individual jumper cable support. Enterprise fiber cabling service contains two options to provide structured (trunked) cable support. The IGS support, depending on the option selected, can provide fiber cable planning, the procurement of cables and associated hardware, and the installation of the cables.

In summary, the five options are:

- Option 1 provides complete individual jumper cable service for a single zSeries machine.
- Option 2 provides a migration of existing cables service for a single zSeries machine.
- Option 3 provides the purchase and installation of customer-specified IBM-qualified individual jumper cables for a single machine.
- Option 4 provides a complete cable service, with trunking, patch panels, and associated hardware for a single zSeries machine. Individual jumper cables can also be purchased as part of this service.
- Option 5 provides a complete cable service, with trunking, patch panels, and associated hardware for the Enterprise. Individual jumper cables can also be purchased as part of this service.

The following chart may help with understanding these offerings.

Option	Enterprise planning	zSeries Cable planning	Trunking	New cables	Cable installation	Document
1	No	Yes	No	Yes	Yes	Yes
2	No	Yes	No	No	Yes	Yes
3	No	No	No	Yes	Yes	Yes
4	No	Yes	Yes	Yes	Yes	Yes

Option	Enterprise planning	zSeries Cable planning	Trunking	New cables	Cable installation	Document
5	Yes	Yes	Yes	Yes	Yes	Yes

Types of cables

The following cables and connectors are used with various z990 features:

Feature.....	Connector	Cable type
ISC-3 link	LC Duplex	9 micron single mode
ETR	MTRJ	62.5 micron multimode
ESCON channel	MTRJ	62.5 micron multimode
FICON Express LX	LC Duplex	9 micron single mode
FICON Express SX	LC Duplex	50 or 62.5 micron multimode
OSA-Express GbE LX	LC Duplex	9 micron single mode
OSA-Express GbE LX	SC Duplex*	9 micron single mode
OSA-Express GbE SX	LC Duplex	50 or 62.5 multimode
OSA-Express GbE SX	SC Duplex*	50 or 62.5 multimode
OSA-Express 1000BaseT	RJ-45	Category 5 UTP
OSA-Express Fast Ethernet**	RJ-45	Category 5 UTP
OSA-Express token ring	RJ-45	STP or UTP

* These connectors used only for older adapters moved to the z990
** This adapter not available for new orders.

Quick connect

The Fiber Quick Connect (FQC) direct-attach harness can be used to connect to the ESCON ports in the zSeries z990. The harness has one MTP (multifiber terminated push-on) connector at one end and six MT-RJ connectors at the other end. One MTP connector contains 12 optical fibers; plugging one MTP connector is the equivalent of plugging six duplex jumper cables. The MTP connector is plugged into one position of the 10-position MTP Base Bracket, mounted at the lower frame front or rear above the floor cuttings, depending on whether the ESCON cards are mounted in the front or rear slots of the I/O cage. The harnesses are routed from there to the ESCON cards, and each MT-RJ connector is plugged into one of the ESCON card receptacles. It takes five direct-attach harnesses to support two ESCON 16-port cards, since all ports are connected to the MTP brackets, even if not all of them are active in the current configuration. The harnesses and MTP Brackets are installed at the factory and the direct-attach harnesses are plugged to the MTP Couplers.

The harnesses enable a trunk cable with MTP connectors to connect to the machine's fiber optic ports. Once the harnesses are plugged, all connects and disconnects can be done using the trunk MTP connectors, making the installation, relocation or rearrangement of the cable connections faster and more convenient.

The FQC option is not required, and may not be appropriate for smaller z990 systems. It does reduce the time needed to connect ESCON cables to the z990. This may be important in larger installations where systems are sometimes repositioned.

4

Software considerations

There are many significant changes in the z990 architecture and hardware features when compared to the z900 processor. Extensive software support has been made available to accommodate these changes in the OS/390, z/OS, z/VM and Linux for zSeries operating systems. The primary features of the z990 that demand this support are:

- Support for more than 15 LPARs
- Support for multiple Logical Channel Subsystems
- Support for more than 256 CHPIDs
- Support for more than 16 physical processors
- Support for an increased number of devices

This software support has been designed at two levels: *compatibility* support and *exploitation* support.

The compatibility support allows these operating systems to define a z990 environment and execute on the hardware in a way that is compatible with earlier processors. In some cases new features, such as the increased number of HiperSockets, may also be exploited.

The exploitation support does exactly what it implies. It allows exploitation of all the announced new features on the z990 without the restrictions that are necessarily imposed by the compatibility support. For example, the z/OS V1R4 and z/VM V4R4 exploitation support allows you to define and run more than 15 LPARs simultaneously.

4.1 OS/390 and z/OS software considerations

As mentioned, z/OS software support for the z990 will be delivered in two phases: *compatibility* and *exploitation*. The compatibility support is delivered at the same time that the z990 hardware becomes generally available. The exploitation support will be available at a later date.

The compatibility support allows OS/390 V2R10 and z/OS releases (except z/OS V1R1) to run on, and coexist with, a z990 processor. Certain functions of the z990 (particularly the new ones) will not be used until the exploitation support is available and installed.

The compatibility support is available as a Web deliverable for OS/390 V2R10, z/OS V1R2 and z/OS V1R3. For z/OS V1R4, both the compatibility and exploitation support will be delivered as separately orderable features and will *not* be available as Web deliverables or through the normal service stream. The compatibility support for z/OS V1R4 will no longer be orderable once the exploitation support becomes available. The exploitation support will be integrated into the base of z/OS V1R5.

4.1.1 Compatibility support and coexistence

Compatibility support for the z990 will be available for OS/390 V2R10, z/OS V1R2, z/OS V1R3 and z/OS V1R4. The compatibility support allows these releases of the operating system to do the following:

- Define a z990 environment with HCD
- Run on a z990 processor in an LPAR in LCSS 0
- Make dynamic I/O changes for LCSS 0 (only)
- Coexist in a sysplex that contains a z990 processor
- Coexist with z990 processors sharing DASD (even outside of a sysplex)

These are broad statements which, in themselves, cannot cover the variations of all possible environments. Under certain circumstances, parts of the compatibility support may not need to be installed, although it is highly desirable that they are. For example, you only need to install the HCD maintenance on the system that is defining the z990; it is not needed on the systems where HCD is not used. The precise circumstances of when the compatibility support is needed are explored later in this section.

Attention: Compatibility and exploitation support will *not* be available for z/OS 1.1.

The rules for whether compatibility support is required or not are:

- Compatibility support is required for *all* images running on a z990.
- Compatibility support is required on *any* image that is used for defining the I/O configuration for the z990.
- Compatibility support is required on *all* images in a sysplex (whether running on a z990 or not) if a Coupling Facility LPAR for that sysplex is running on a z990 and has an LPAR identifier greater than X'F'. Note that this is not the same as having more than 15 LPARs. LPAR ids can be assigned arbitrarily; it is possible to have an LPAR id greater than X'F' even if it is the only LPAR on the processor.

Compatibility support allows the supported operating systems to run in LCSS 0 on the z990 processor. It is not possible to run z/OS or OS/390 in LCSS 1, even with the compatibility support installed. (Running in LCSS 1 is supported for z/OS 1.4 with the exploitation support.) A Coupling Facility LPAR can reside in any Logical Channel Subsystem.

The compatibility maintenance is supported on any processor that is already supported by that operating system. For example, you could install the compatibility maintenance onto a z/OS 1.2 system that is running on a 9672 G5 processor.

Figure 4-1 on page 81 illustrates a situation where compatibility support is *not* required.

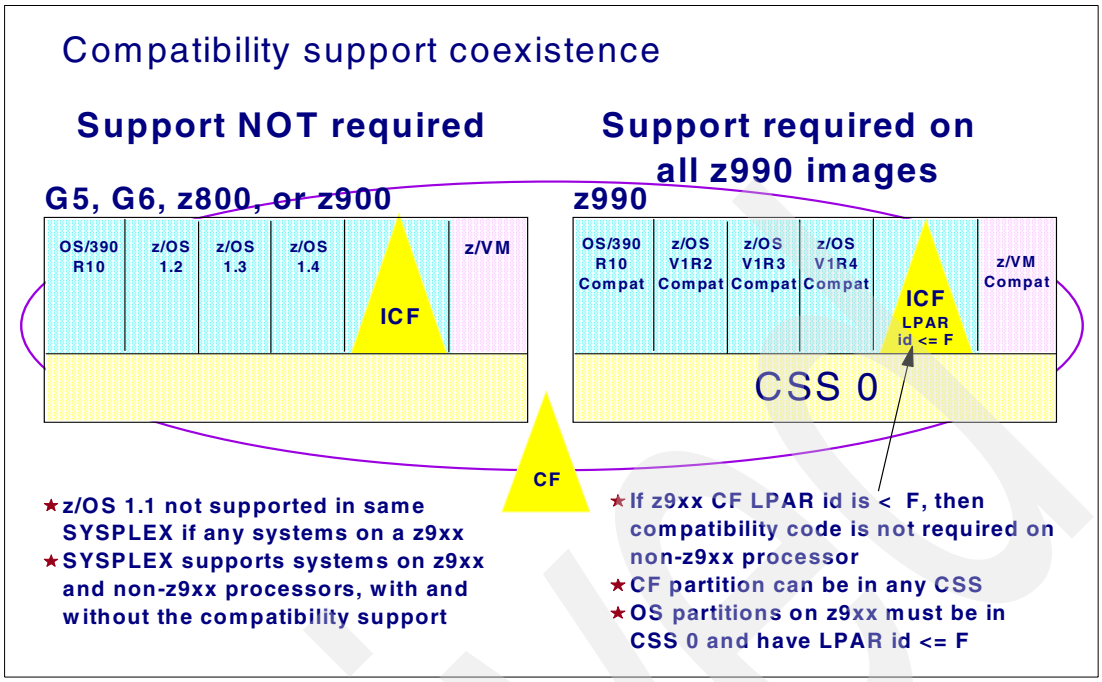


Figure 4-1 Conditions where compatibility maintenance is not required

Figure 4-2 illustrates a situation where compatibility support *is* required.

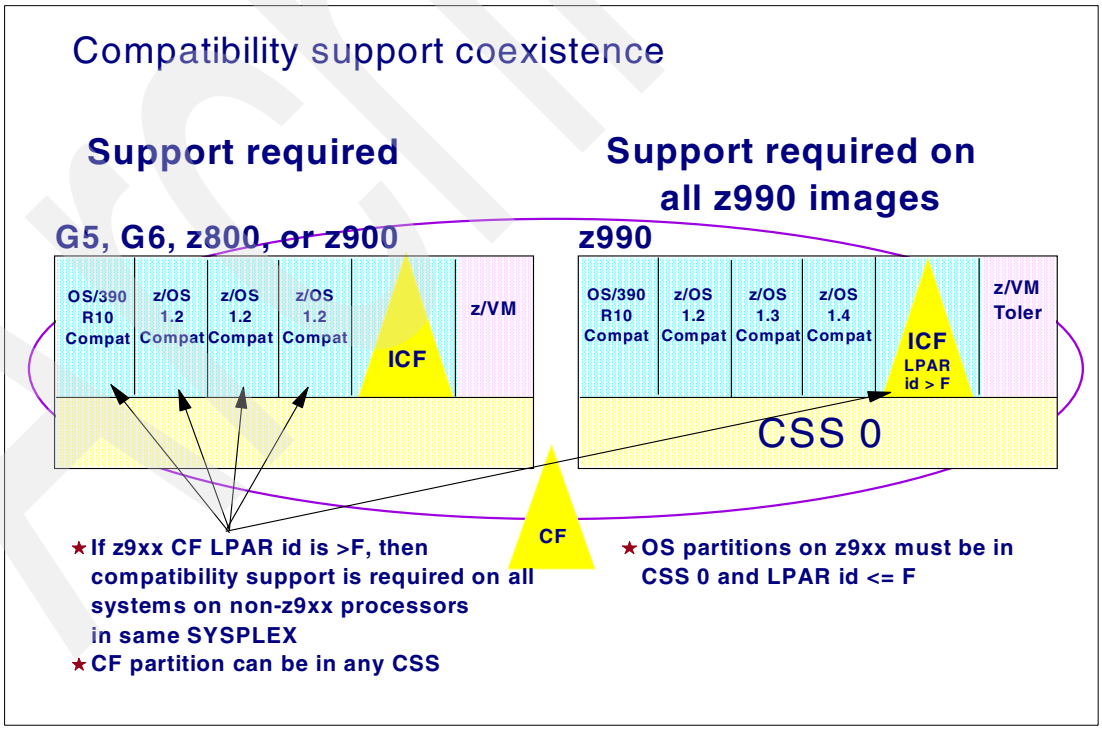


Figure 4-2 Conditions where compatibility maintenance is required

Compatibility support updates several functional areas in order to accommodate the z990 processor, both as a standalone system and in a multisystem environment.

HCD

The support added to HCD[1] for both compatibility and exploitation support allows the definition of the z990 processor and I/O configuration while working in another (possibly older) system. The new HCD elements apply to z/VM and z/OS.

The updated HCD function introduces a new concept: a *validated work IODF*. This is a new status for an IODF data set. It contains a complete set of validated processor, LCSS, channel, partition, control unit and I/O device definitions. A validated work IODF would not normally contain the physical channel identifiers (PCHIDs) for channels. These can be added to the validated work IODF by the output from the CHPID Mapping Tool (CMT).

Support has been added to HCD to work with the CHPID Mapping Tool for PCHID assignments. This support allows an IOCP source statement data set to be created from a validated work IODF; it also allows the data from the CHPID Mapping Tool to be merged with the validated work IODF to complete the PCHID assignment.

The compatibility support for HCD allows an installation to do the following:

- Define a z990 environment with multiple Logical Channel Subsystems.
- Make dynamic hardware changes to CSS 0 only. Devices cannot be added, modified or deleted if they are also defined to another CSS. This would require a Power-on Reset. Full dynamic change support is implemented in the exploitation support for HCD.
- Software ACTIVATEs can be done regardless of how many Logical Channel Subsystems are defined.

Definition of the new processor environment is discussed at length in “Migrating the I/O configuration definitions with HCD” on page 98.

CFRM and CFRM Policy support

The CFRM support for the z990 processor now allows two-digit LPAR identifiers to be assigned in the CFRM policy. This extension is essential for allowing more than 15 LPARs to coexist on the processor.

The LPAR Id is *not* the same as the partition number used in prior processors. The LPAR identifier is assigned in the image profile on the Support Element and is unique to each LPAR. It is a two-digit value in the range x’00’ to x’3F’.

Attention: The LPAR id for a Coupling Facility (specified in its Image profile on the Support Element) must correspond with that specified on the **`PARTITION()`** keyword in the CFRM. This is different than earlier zSeries and S/390 processors. On previous systems, the value specified on the PARTITION() keyword was the partition number from the IOCDS.

This update is identical for both compatibility and exploitation support. The LPAR id is also used in IOS and XCF messages.

Automation

Several messages have been changed to accommodate the new two-digit LPAR identifier. These are shown in Figure 4-3 and Figure 4-4 on page 83.

[1] This section discusses compatibility support for HCD. Equivalent compatibility support is also needed for HCM.

Message	Command or Message Text
IXC357I	D XCF,COUPLE
IXC360I	D XCF,STR,STRNAME=
IXC361I	D XCF,CF
IXC362I	D XCF,CF,CFNAME=
IXC500I	IXC500I CONFIRM REQUEST TO USE COUPLING FACILITY type.mfg.plant.sequence...
IXC505I	IXC505I STRUCTURE strname IN COUPLING FACILITY type.mfg.plant.sequence...
IXC506I	IXC506I CONNECTION conname TO STRUCTURE strname IN COUPLING FACILITY type.mfg.plant.sequence...
IXC507I	IXC507I CLEANUP FOR COUPLING FACILITY type.mfg.plant.sequence...
IXC515I	IXC515I STRUCTURE strname IN COUPLING FACILITY...DEALLOCATE. MORE CURRENT VERSION OF STRUCTURE FOUND IN COUPLING FACILITY
IXC517I	IXC517I SYSTEM sysname ABLE TO USE COUPLING FACILITY...
IXC518I	IXC518I SYSTEM sysname NOT USING COUPLING FACILITY ...
IXC519E	IXC519E COUPLING FACILITY DAMAGE RECOGNIZED FOR...

Figure 4-3 Changed messages to accommodate two-digit LPAR ids

Message	Command or Message Text
IXC551I	IXC551I STRUCTURE strname IN COUPLING FACILITY ...
IXC579I	IXC579I deallocatype DEALLOCATION FOR STRUCTURE strname ...
IXL008I	IXL008I PATH chpid HAS BEEN INVALIDATED TO CUID: cuid COUPLING FACILITY...
IXL010E	IXL010E NOTIFICATION RECEIVED FROM COUPLING FACILITY...
IXL141I	D M=CHP for coupling facility CHPID
IXL150I	D CF or D CF,CFNAME=name
IXL157I	IXL157I PATH chpid IS NOW OPERATIONAL TO CUID: cuid ...
IXL158I	IXL158I PATH chpid IS NOW NOT-OPERATIONAL TO CUID: cuid...

Figure 4-4 More changed messages to accommodate two-digit LPAR ids

SMF

CPU and PR/SM activity data is recorded by RMF™ in the SMF type 70 subtype 1 records. Prior to the z990, these records were always shorter than 32 KB. With the increased number of LPARs and logical CPs, these records could potentially increase in size beyond the 32 KB limit. To accommodate this, each record is now broken into pieces where each piece is shorter than 32 KB. Each piece is self-contained; that is, the record can be processed without re-assembling the broken pieces. If you have any site-specific processing of this data outside of RMF, you may need to review that application to ensure that it is no longer dependent upon all this data being contained within a single record.

RMF Monitor 1 Device Activity reporting is recorded in the SMF 74 subtype 1 records. These have been updated to support an extended device data section. This section now includes the Initial Command Response time for the device. A similar change has also been made to the RMF Monitor II Device Activity recording in the SMF 79 subtype 9 records.

The SMF records for the SRM Decisions data stored in type 99 (subtypes 8 and 9) have also been extended. These records now contain the LCSS id for the WLM LPAR management and I/O subsystem information.

RMF

There are several changes to RMF reports to accommodate the enhanced I/O subsystem and improved collection of channel measurement data.

The I/O Activity report no longer shows the Control Unit Busy (CUB) and Director Port Busy (DPB) times. (The corresponding percentage and pending reason fields for CUB and DPB have also been removed from the Monitor III reports). This information was already available at an LCU level and is much more useful than figures broken out for individual devices. Furthermore, the Director Port Busy field would only show a non-zero value for the events when *all* director ports were busy. If an individual director port was found to be busy but a connection was established through an alternate path, then this figure was not updated. With FICON connections, a Director Port was never reported busy since this type of channel allows multiple data transfers to occur simultaneously.

A better measure of fabric contention has now been provided with the Initial Command Response Time (CMR). This is a measure of the time taken from sending a command to a device, to a response that it has accepted the command. This new metric (AVG CMR DLY) has now replaced the older AVG CUB and AVG DPB columns on the Monitor I Device Activity Report. Corresponding DELAY CMR% and PENDING REASON CMR displays have been introduced into the Monitor III reports. Monitor III exception reporting has also been updated to replace the previous CUBDL and DPBDL conditions with the new condition, CMRDL.

ICKDSF

ICKDSF Release 17 is now needed on all systems that share DASD with a z990 processor. This applies to z/VM and z/OS systems. The need for this ICKDSF release applies even to systems that are not part of the same sysplex, or that are running a non-MVS-based operating system, such as z/VM.

ICSF

If you use zSeries cryptographic hardware functions with ICSF, you must install the compatibility support for this feature.

Important: *The ICSF compatibility support is only available for z/OS V1R3 and V1R4*. If you are running on an earlier level of z/OS or OS/390, then you *must* upgrade the LPARs using ICSF to either z/OS V1R3 or V1R4 before you can move that environment onto a z990 processor.

The PCIXCC card is the only cryptographic hardware for the z990 that supports Secure Keys. Prior to the availability of these cards, cryptographic functions are supported by the PCICA cards and the CP Assist functions on each PU. These provide Clear Key support. The PCICC cryptographic card used on the z900 and z800 processors is *not* supported on the z990.

4.1.2 Compatibility support restrictions

The compatibility support provided does not allow you to make full use of all the capabilities of the z990. Specific restrictions include:

- z/OS must be IPLed in a partition defined in CSS 0. If it is IPLed in a partition in any other CSS, it will terminate with a 07C-01 wait state.
- z/OS must be IPLed in a partition that has an LPAR identifier in the range 0-F. The LPAR identifier is specified in the Image profile on the HMC. If the LPAR identifier is outside of this range, then z/OS will terminate with a 07C-02 wait state.
- Dynamic activates for hardware changes can only be done for CSS 0. A Power-on Reset is required for changes to other Logical Channel Subsystems. Dynamic activates for hardware changes within CSS 0 cannot be done if the resource is also defined in any other CSS. For example, if a DASD control unit has connections to CSS 1, then additional connections cannot be added to CSS 0 dynamically while in compatibility mode.

4.1.3 Exploitation support

The z/OS exploitation support for the z990 will only be delivered for z/OS V1R4 and later releases. It will be shipped as a separately orderable feature.

The z/OS V1R4 exploitation support will allow:

- z/OS V1R4 to run in a partition defined to any Logical Channel Subsystem
- z/OS V1R4 to run in a partition with an LPAR identifier greater than X'F'
- Dynamic activates for hardware changes to any LCSS

4.1.4 Exploitation support considerations

The following areas need to be considered when running z/OS V1R4 in exploitation mode:

SMF

The SMF type 89 record used for recording Product Usage data has been extended for exploitation mode support. A new field, SMF89LP3, allows an 8-bit LPAR id to be stored. This field is marked valid by the new flag bit SMF89LPM. When an LPAR id is less than or equal to X'F', the LPAR id is stored in both the new field and the old 4-bit SMF89LP2 field to maintain compatibility.

Standalone dump

The z/OS V1R4 systems that have the exploitation maintenance applied must generate a new version of the standalone dump program. This standalone dump program cannot be used for dumping systems at earlier releases of z/OS or z/OS V1R4 systems that have only the compatibility support installed.

Automation

The output from the **D M=CPU** command has been enhanced to show the two-digit LPAR id (set in the Image profile), the LCSS id associated with the logical CPUs associated with the LPAR in that LCSS, and the MIF Image Id; see Figure 4-5 on page 86. The logical CPU address no longer appears in the first digit of the serial number as a result of the change to the STIDP instruction.

```
D M=CPU
IEE174I 14.45.55 DISPLAY M 159
PROCESSOR STATUS
ID  CPU              SERIAL
0   +              1293052084
1   +              1293052084
2   +              1293052084

CPC ND = 002084.R01.IBM.02.000000049305
CPC SI = 2084.R01.IBM.02.0000000000049305
CPC ID = 00
CPC NAME = XXXXXXXX
LP NAME = SC66, LP ID = 12
CSS ID   = 1
MIF ID   = D

+ ONLINE    - OFFLINE    . DOES NOT EXIST
CPC ND   CENTRAL PROCESSING COMPLEX NODE DESCRIPTOR
CPC SI   SYSTEM INFORMATION FROM STSI INSTRUCTION
CPC ID   CENTRAL PROCESSING COMPLEX IDENTIFIER
CPC NAME  CENTRAL PROCESSING COMPLEX NAME...
```

Figure 4-5 Changes to the D M=CPU command output

The output from the **D IOS,CONFIG(HSA)** or **D IOS.CONFIG(ALL)** commands has been changed to remove all references to SHARED and UNSHARED control units. Previously, this command would have been used to determine the HSA space available for dynamically adding control units and I/O devices. On the z990 processor, the number of additional devices that can be added dynamically is determined by the MAXDEV value associated with each LCSS. This parameter is specified via HCD and is set in the IOCDS. The new output from the **D IOS** command is shown in Figure 4-6.

```
z/OS with Exploitation Feature on z9xx Processor

IOS506I hh.mm.ss I/O CONFIG DATA
HARDWARE SYSTEM AREA AVAILABLE FOR CONFIGURATION CHANGES
PHYSICAL CONTROL UNITS              50
CSS  0 - LOGICAL CONTROL UNITS     100
         SUBCHANNELS              2000
CSS  1 - LOGICAL CONTROL UNITS     120
         SUBCHANNELS              3240
CSS 2 - LOGICAL CONTROL UNITS      120
         SUBCHANNELS              4000...
```

Figure 4-6 Output from the D IOS command on the z990

The IEE174I, IOS050I and IOS051I messages have also been changed to display not just the CHPID number, but also the associated PCHID. This addition assists diagnosis of hardware problems. The new output is shown in Figure 4-7 on page 87.

.

```
D M=CHP(58)
IEE174I 17.00.35 DISPLAY M 263
CHPID 58:  TYPE=1B, DESC=FICON SWITCHED, ONLINE
DEVICE STATUS FOR CHANNEL PATH 58
    0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
022 +  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
023 +  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
098 .  +  +  .  .  .  .  .  .  .  .  .  .  .  .  .
SWITCH DEVICE NUMBER = B000
DEFINED ENTRY SWITCH - LOGICAL SWITCH ID = 03
ATTACHED ND = 006064.001.MCD.01.000000040012
PHYSICAL CHANNEL ID = 03FD
********************** SYMBOL EXPLANATIONS **********************
+ ONLINE    @ PATH NOT VALIDATED   - OFFLINE    . DOES NOT EXIST
* PHYSICALLY ONLINE   $ PATH NOT OPERATIONAL
```

```
IOS050I CHANNEL DETECTED ERROR ON dddd,yy,op,stat,PCHID=pppp

IOS051I INTERFACE TIMEOUT DETECTED ON dddd,yy,op,stat,PCHID=pppp
```

Figure 4-7 PCHID support for channel messages

EREP

The PCHID value associated with a particular CHPID is now displayed in the EREP Subchannel Logout record, as shown in Figure 4-8.

```
DEVICE NUMBER:     0A02      REPORT:  SLH EDIT                  DAY YEAR  JOB
                             SCP:     VS 2 REL.  3          DATE: 258  02
DEVICE TYPE:       CACA
                             CPU MODEL:    2084                 HH MM SS.TH
CHANNEL PATH ID:   3E        LOGICAL  CPU ID: 232920        TIME: 12 50 29.84
PHYSICAL CHAN ID:  XXXX      PHYSICAL CPU ID: 612920        PHYSICAL CPU ADDRESS: 00

                   CC    CA    FL  CT
FAILING CCW        02 01DA2500 24 5000              VOLUME SERIAL         N/A
                                                    SUBCHANNEL ID NUMBER  000108B2
                   K  FLAGS    CA     US SS  CT     ERROR TYPE            OTHER
SCSW               64 C24017 01DA2318 00 02 5000

---UNIT STATUS----  SUB-CHANNEL STATUS  ----------------SCSW FLAGS-----------------
                                              FLAG 0     FLAG 1     FLAG 2
```

Figure 4-8 EREP support of PCHIDs

Extended Channel Measurement Block (ECMBs)

The z990 processor supports Extended Channel Measurement Blocks for the new I/O architecture. It also supports the original XA I/O architecture CMBs for compatibility reasons.

Under z/OS V1R4 with the exploitation support, the ECMBs will now be placed in a system common area dataspace. Because of this, you might want to review your setting of the MAXCAD value in IEASYSxx. The CMB parameter in IEASYSxx is now redundant and will be ignored.

Dynamic activates for hardware changes

If a z990 processor is running multiple z/OS systems with a mixture of exploitation and compatibility support and a new hardware configuration is being activated, the activate must be done using one of the z/OS systems with exploitation support installed if:

- There is more than one Logical Channel Subsystem defined
- A non-zero LCSS is being changed, or resources affected by the change are defined to non-zero LCSSs

Dynamic CHPID management

On a z990, systems that are part of the same LPAR cluster may be in different LCSSs. This can be a little confusing; Figure 4-9 may help clarify concepts.

Dynamic CHPID management is supported even if the LPAR cluster spans multiple Logical Channel Subsystems. Movement of CHPIDs within the LPAR cluster is confined to movement within that LCSS.

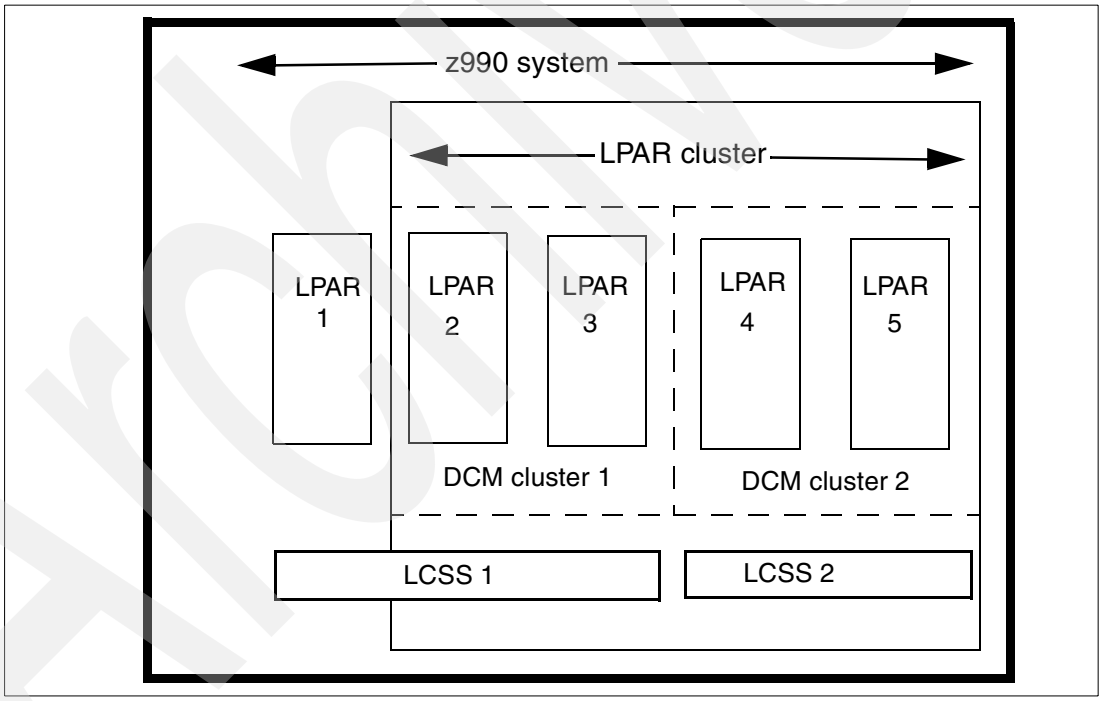


Figure 4-9 DCM clusters

If an LPAR cluster consists of multiple LCSSs, then the DCM command **`VARY SWITCH`** must be issued from one of the z/OS systems with exploitation support installed. The DCM command **`SETIOS DCM=ON/OFF`** can be issued from any system, whether in compatibility mode or exploitation mode.

Greater than 15 LPARs

You must define more than one LCSS if you plan to use more than 15 LPARs. An individual Logical Channel Subsystem can only support up to 15 logical partitions. z/OS (not V1R1)

and OS/390 V2R10 LPARs running in compatibility mode can only reside in LCSS 0. Coupling Facilities, z/VM V4R4, and Linux for zSeries can reside in any Logical Channel Subsystem.

ISV software

You should check with your ISV about any required maintenance for ISV products running on the z990. This is particularly important for any software that uses the results of the STIDP instruction to control where that software can run. This is discussed further in “ISV software” on page 97.

4.2 z/VM software considerations

Compatibility support for z/VM on the z990 is provided for z/VM V3R1, V4R2 and V4R3. This is planned to be available through the normal service stream with APAR VM63124.

z/VM V4R4 will provide support for exploitation of the new facilities on the z990 processor.

The z/VM compatibility support is almost identical to that for z/OS in the functions that it provides, although it does not exploit dynamic CHPID management. On the z990, the compatibility support:

- Allows z/VM to run only in LCSS 0.
- Allows a maximum of 15 z/VM LPARs.
- Allows only LPAR identifiers less than or equal to 15 (X’F’).
- Does not use the extended I/O measurement facilities.
- Does not use adapter interruptions for OSA-Express and FCP channels.
- Does not provide new performance assist functions for V=V guests.

The z/VM exploitation support will allow operation in all Logical Channel Subsystems and with LPAR identifiers greater than X’F’. It will also cover use of extended I/O measurements. list-directed IPL, adapter interruptions, and new performance assist functions. Support for operation with two LCSSs is planned for June 16, 2003 and support for 30 LPARS is planned for October 31, 2003.

You should reference the initial IBM announcement letter for the z990, and subsequent announcements, for more detailed information concerning z/VM support.

4.3 Linux software considerations

Linux support for the z990 processor will be delivered through a code drop in June 2003 for the open source community. This will provide both compatibility and exploitation support, and will support superscalar performance, two LCSSs, and enhanced network performance and functionality.

This z990 exploitation support will also be delivered through developerWorks.

4.4 VSE/ESA™ software considerations

Compatibility support for VSE/ESA on the z990 is provided for VSE/ESA V2R5, V2R6, and V2R7. Note that for VSE/ESA V2R5 and V2R6, you must install APAR DY45944. VSE/ESA support for two LCSSs and 30 LPARs is planned for October 31, 2003.

5

Migration considerations

This chapter is focussed on the changes that existing S/390 or zSeries customers must consider when moving to z990 technology. It is applicable to the following situations:

- Customers upgrading an existing z900 to a z990. This scenario would be applicable to customers who are upgrading primarily to overcome constraints on capacity, channels, the number of logical partitions, or memory.
- Customers replacing a previously installed processor (for example, an IBM 9672 system) with a z990. This scenario would be applicable to customers seeking to run existing workloads while providing the additional capacity needed to support server consolidation from other (non-S/390 or zSeries) platforms.
- Customers consolidating workloads from multiple z900 or 9672 processors onto a single z990. This scenario would be applicable to customers who are using the z990 primarily to overcome constraints on floor space, or seeking to maximize the use of resources such as memory or processing cycles that are fragmented over multiple existing systems.

In the following sections we provide an overview of the considerations at hardware, software, and operational levels.

5.1 Hardware migration considerations

With the exception of HiperSockets and Internal Coupling links, all channels defined in HCD or an IODF source deck must also have an associated Physical Channel Identifier (PCHID). This is discussed in “Channel subsystem” on page 32.

5.1.1 ESCON channels

All ESCON channels on the z990 are implemented with the 16-port ESCON I/O card that was introduced with the z900. These ESCON channels use the MT-RJ connectors. Installations that are upgrading to z990 from z900 may require some changes in this area. z900s that were originally upgraded from G5/G6 processors may have the older 4-port ESCON cards still in use; these are not supported on the z990. The 4-port ESCON channel cards used the large Duplex connectors. When such a z900 processor is upgraded to a z990, these 4-port cards

will be replaced by the new 16-port ESCON cards which have the smaller MT-RJ connector. In these situations, customers have two options for ESCON cables:

- Replace the cabling to the attached control unit, switch, or patch panel with new cabling that has an MT-RJ connector at the processor end.
- Use conversion kits. These cables have an MT-RJ plug at one end and a conventional ESCON duplex socket at the other. They are physically short (approximately 2 metres long), but will allow the existing cabling to be reused. The converter cables are relatively fragile and a secure housing for them in the floor void should be considered if large numbers of them are to be used. If long cable distances and/or multiple patch panels are involved in the connection to the control unit, then the additional link loss from using the conversion kits may need to be considered.

In addition, the physical location of the associated channel within the z990 frames may change from its current position in the z900. It should be possible to minimize extensive cable movement by careful planning and judicious use of the CHPID Mapping Tool.

ESCON CTC

Other than using a MIF id in definitions (instead of a partition number), there is no special change for these channels. Each ESCON CHPID number is within a single LCSS. ESCON channels cannot recognize the effects or existence of other LCSSs.

5.1.2 FICON channels

The z990 can only support FICON Express channels. In contrast, the z900 could support both FICON Express and the older FICON channels. If the z990 is produced as a result of an upgrade from a z900 (as opposed to a newly built machine), then any of the older FICON channels will be replaced with the new FICON Express channels of the same mode. The FICON Express channels use the LC-Duplex connector which is different than the SC-Duplex connector used on the older FICON channels. In this situation, you can reuse your existing cabling by using the appropriate conversion kits as shown in Table 5-1.

Table 5-1 SC-Duplex and LC-Duplex connector conversion cables

Type	Description
50 MM LC-Duplex/SC-Duplex conversion kit	This conversion kit is used with 50/125 micrometer fiber to connect SX ports to cables that already have an SC-Duplex connector fitted.
62.5 MM LC-Duplex/SC-Duplex conversion kit	This conversion kit is used with 62.5/125 micrometer fiber to connect SX ports to cables that already have an SC-Duplex connector fitted.
9 SM LC-Duplex/SC-Duplex conversion kit	This conversion kit is used with 9/125 micrometer fiber to connect LX ports to cables that already have an SC-Duplex connector fitted.

5.1.3 FICON CTC

The addressing of FICON CTCs on a z990 has been enhanced when compared to their implementation on earlier processors. Consequently, this *may* require changes to:

- Existing FICON CTC definitions on a processor being upgraded to a z990
- Existing FICON CTC definitions on processors that connect to a z990

The value specified for the CUADD parameter in a FICON CTC control unit definition has always allowed an 8-bit value to be specified. This is now used to allow identification of the LCSS associated with the target LPAR. The high-order four bits of this value are used to identify the LCSS that contains the target LPAR. The low-order four bits are used to identify the target LPAR within that LCSS. This value must correspond to the MIF Image id; this value is set in the partition number field in the HCD Add Partition screen.

In the case where an earlier processor is being replaced with or upgraded to a z990, the FICON CTC control unit definitions can remain unchanged provided that *all* of the following conditions are met:

- All FICON CTC connected LPARs reside in LCSS 0
- All LPARs retain their existing partition number
- All FICON CTC channels on the z990 retain their original CHPID address

5.1.4 Parallel channels

Parallel channels are not supported on the z990. Installations using parallel channels have two choices for connectivity on the z990:

- Convert these devices to ESCON attachment, or replace them with ESCON/FICON-attached equivalents. For example, the ESCON-attached 2074 has superseded the parallel-attached 3174, which is no longer manufactured.
- Use an ESCON converter to allow the parallel-attached control units to be connected to an ESCON channel with CBY or CVC CHPIDs. This would be appropriate for sites that retain one or two old 3420 open-reel tape drives for compatibility or archive retrieval purposes.

IBM no longer manufactures the 9034-1 ESCON converter. We recommend the Optica 34600 FXBT ESCON Converter as a suitable alternative.

5.1.5 OSA-Express and OSA-2 adapters

A number of changes or upgrades are required for OSA adapters.

OSA-Express Gb Ethernet adapters

During an upgrade from z900 to z990, any OSA-Express Gb Ethernet cards previously installed in the z900 will be retained in the target configuration. These cards use the SC-Duplex connectors. Any additional OSA-Express Gb Ethernet cards provided by the upgrade, or cards on newly built machines, will be a new OSA-Express Gb Ethernet type which uses the LC-Duplex connectors. If you have any cables that are terminated with SC-Duplex connectors, these can be made to connect to the new OSA-Express Gb Ethernet cards by using an appropriate conversion kit. The correct conversion kit type is shown in Table 5-1 on page 92.

OSA-Express 1000BaseT Ethernet

These OSA-Express cards will be provided on new processors and/or any upgrades to provide additional Ethernet connectivity. They are functionally identical to the OSA-Express Fast Ethernet cards that they supersede, except for providing 1000 Mbps operation.

OSA-Express Fast Ethernet

These cards have been superseded by the new OSA-Express 1000BaseT Ethernet cards. Any installed OSA-Express Fast Ethernet cards will be carried across to the z990 during an upgrade from a z900. Any additional ports required will be satisfied with the new OSA-Express 1000BaseT Ethernet cards.

OSA-Express Token Ring

These cards are fully supported on the z990 processor.

OSA-Express 155 ATM

These cards are not supported on the z990. You should use a switch or a router with an appropriate network interface (OSA-Express 1000BaseT Ethernet or OSA-Express Gb Ethernet) to connect to the processor.

OSA-2 FDDI

These cards are not supported on the z990. You should use a switch or a router with an appropriate network interface (OSA-Express 1000BaseT Ethernet or OSA-Express Gb Ethernet) to connect to the processor.

OSA-2 Ethernet/Token Ring

These cards are not supported on the z990. For Ethernet connectivity, you should plan to migrate to the new OSA-Express 1000BaseT Ethernet cards as part of any upgrade from z900. Similarly, for token ring connectivity, you should plan to migrate to the OSA-Express Token Ring cards during any upgrade from z900.

The OSA-2 Ethernet/Token Ring card previously supported two ports sharing a common CHPID. When migrating to OSA-Express 1000BaseT Ethernet or OSA-Express Token Ring, each port must have its own CHPID assigned. These new cards also fully support QDIO to improve performance.

5.1.6 ISC and ICB coupling links

There is an overall limit of 32 internal and 32 external coupling links. Within this upper bound, each z990 can support a maximum of:

- 32 Internal Coupling (IC) links
- 32 ISC-3 links
- 16 ICB-3 links
- 16 ICB-4 links
- 8 ICB (also known as ICB-2) links

ISC links

The ISC links used on the z990 are identical to and fully compatible with the ISC links used on the z800 and z900. They are also compatible with the ISC links on G5 and G6 processors when used in compatibility mode, that is, defined as **`TYPE=CFS`** or **`TYPE=CFR`** CHPIDs.

When connected to a z800, z900, or other z990 processor, the ISC links can operate in *peer* mode (**`CHPID TYPE=CFP`**) and operate at up to 2 Gbps. When connected to 9672 processors, the ISC links must operate in compatibility mode and can transfer data at up to 1 Gbps.

ICB links

The new ICB-4 links can only connect to other z990 processors. ICB and ICB-3 links are needed to connect to G5/G6 and z900 processors, respectively.

5.1.7 HiperSockets

With the exception of spanned IQD CHPIDs, HiperSockets are implemented almost exactly as they were on the z800 and z900. The maximum possible number of HiperSocket LANs and the number of connections to them has been quadrupled compared to the z900

implementation. Up to 16 HiperSocket LANs and 4,096 communication queues can now be defined on the z990.

An IQD CHPID associated with a HiperSocket LAN will need to be designated as *spanned* if the LPARs that connect to it are spread over multiple Logical Channel Subsystems.

The IQD CHPID definition in the IOCDS now has a new parameter, CHPARM, which has replaced the OS parameter previously used on the z800 and z900 CHPID statements. Functionally, this parameter is completely compatible with the one it has replaced. See Table 3-1 on page 47 for details.

5.1.8 Cryptographic hardware

There is no onboard CCF cryptographic coprocessor on the z990 as there was with earlier zSeries and S/390 processors. Many of these functions are now carried out by the Message-Security Assist instructions integrated into each PU.

The z990 also supports the PCICA and the new PCIXCC cryptographic hardware. (Note that the PCICC cryptographic cards available on the z800 and z900 are *not* supported.) Unlike previous processors, the z990 does not require a CHPID to be assigned to each PCICA or PCIXCC cryptographic card.

These cards require the appropriate ICSF support code for the z990. See “ICSF” on page 84 for further details.

5.1.9 Memory

The amount of storage needed for the z990 Hardware System Area (HSA) is greater than that required on the z800 or z900. It is not possible to give a single figure for the new HSA size, since it is dependent upon many factors such as the number of Logical Channel Subsystems defined and the maximum number of devices that each LCSS can support. It is important not to overdefine the number of devices that each Logical Channel Subsystem can potentially address, since this will increase the requirements for HSA storage.

On the other hand, it is important that sufficient room is allowed for growth since the number of devices that an LCSS can support may only be changed by a Power-on Reset. (The reserved space for HSA expansion for additional devices is no longer set through the HMC; it is determined by the maximum number of devices that the LCSS defined as supporting via the MAXDEV parameter). The actual HSA requirements for a configuration defined in an IOCDS can be estimated by using the HSA Estimator Tool on the HMC.

A large configuration (with multiple LCSSs and a large number of subchannels (device addresses) configured for each LCSS) can use considerable storage for HSA.

5.1.10 PR/SM

Unlike previous zSeries and S/390 processors, the z990 only supports operation in LPAR mode; basic mode is *not* supported. If you are migrating from a basic mode environment, you may have to consider the education requirements for your operators and systems programming staff.

5.1.11 Coupling Facilities

A z990 processor can only connect to Coupling Facilities that are implemented on G5, G6, z800, z900, or z990 hardware. Coupling Facilities implemented on earlier technology are not supported.

G5- and G6-based Coupling Facilities will require CFCC level 11 with a toleration patch if there are more than 15 LPARs on the z990. z800- and z900-based Coupling Facilities will require CFCC level 11 or 12 with a compatibility patch if there are more than 15 LPARs on the z990. This CF maintenance requires a POR to be effective.

The size of your structures may increase when moving from a non-64 bit CFCC (level 9 or less) to level 12. More information on any changes and the amount of extra storage required can be found at:

http://www.ibm.com/servers/eserver/zseries/pso/cftable.html

A Coupling Facility LPAR can reside in any Logical Channel Subsystem on the z990.

5.1.12 Power, cooling, and floor space requirements

zSeries 990 Installation Manual for Physical Planning, GC28-6824, is the definitive reference material for z990 power and cooling requirements, and actual installation planning should be done in conjunction with that publication.

The following information is a high-level summary of the key requirements.

Power and cooling requirements

The z990 uses two fully redundant three-phase line cords like the z900. These three-phase supplies can be at any voltage ranging from 200V to 480V. The power consumed is higher than the z900; the actual amount is determined by the machine configuration and is shown in Table 5-2. (This table can also be used for the heat-to-air cooling requirements). Since the z990 can operate from a single power cord in the event that one fails, each power cord and source must be capable of supporting the entire power load of the system.

In addition, separate single-phase service outlets of 100-130V or 200-240V are required to power the Hardware Management Console with its display and modem (if used).

Table 5-2 System power consumption/heat load

	I/O configuration		
Model	with 1 I/O cage used	with 2 I/O cages used	with 3 I/O cages used
A08	6.74 kW	10.46 kW	13.81 kW
B16	9.57 kW	13.27 kW	16.98 kW
C24	11.82 kW	15.53 kW	19.23 kW
D32	13.98 kW	17.68 kW	21.39 kW

Notes:
1. Assumes maximum supported configuration.
2. The power factor is approximately unity.
3. Input power (kVA) equals output power (kW).
4. For heat output expressed in kBTU per hour, multiply table entries by 3.4.

Floor space requirements and loading

Unlike previous zSeries and S/390 processors, z990 processors always consists of both an A-frame and a Z-frame. There is no separate B frame as there was on the z900 for the Internal Battery Feature. The total footprint area for the two frames (including door and covers) is 1577mm (62.1 inches) deep by 1542mm (60.7 inches) wide. This is approximately 5 inches deeper than previous 9672 processors, but a slightly smaller footprint (both width and depth) when compared to a maximum size (A+B+Z frames) z900.

Note: The measurements here do not the include necessary service area clearances which need to be reserved around the machine.

The z990 frame A and Z combination can weigh as much as 2007 kg (4415 pounds). This is more than double the previous maximum weight of a 9672 processor, which was 853 kg (1880 pounds). In contrast, the maximum weight of a z900 with the Internal Battery Feature is 1866 kg (4113 pounds).

5.2 OS/390 and z/OS software migration considerations

z/OS software support for the z990 will be delivered in two phases: *compatibility* and *exploitation*. The compatibility support will be delivered at the same time that the z990 hardware becomes generally available. The exploitation support will be available at a later date.

5.2.1 PR/SM

As previously mentioned, the z990 processor is only supported in LPAR mode; basic mode is not an option. If you currently run in basic mode and are migrating to a z990, this will have to be taken into consideration when you are defining the z990 environment under HCD. Furthermore, for an OS/390 or z/OS system, you should review the setting of the CPENABLE parameter. This parameter is used to determine the thresholds at which processing engines become enabled for interrupts.

5.2.2 ISV software

You should check with your software vendors about any maintenance or new releases required for ISV products running on the z990. This is particularly important for any software that uses the results of the STIDP or STSI instructions to control or monitor where that software is run. The format of the data returned by the STIDP instruction has been changed to accommodate two-digit LPAR identifiers. See “STIDP instruction changes” on page 67 for further details.

Other changes on the z990, such as the extended channel measurement blocks and the new I/O subsystem architecture, may also require compatibility support from your ISVs; this is likely to be especially true for system monitoring products.

5.2.3 OS/390 and z/OS software support

This subject is discussed extensively in “Compatibility support and coexistence” on page 80 and “Exploitation support” on page 85. You should also review the 2084DEVICE PSP bucket from the IBM Software Support Center for any additional maintenance that may be required.

5.3 Migrating the I/O configuration definitions with HCD

The HCD support on z/OS has been substantially enhanced to accommodate the new facilities provided on the z990:

- Multiple Logical Channel Subsystems
- Physical channel identifiers (PCHIDs)
- Greater than 15 LPAR support
- Spanned CHPIDs
- Improved CHPID Mapping Tool

Previously, with z800 and z900 processors, the CHPID Mapping Tool has been used separately from HCD to create a mapping file that was used during installation. (The mapping file was optional since it was quite acceptable for an installation to use the default channel assignments; however, more work would be needed by the systems programmer to allocate the CHPIDs sensibly in order to maximize availability.)

An advantage of using the CHPID Mapping Tool was that it could take the existing IOCP statements produced by HCD, analyze the control unit definitions, and then allocate the CHPIDs across multiple channel cards automatically to achieve the same result. The CHPID Mapping Tool would then produce a report and mapping file for use during installation. This was highly advantageous for customers moving from older hardware, since the IODF would require minimal changes.

A major difference with the z990 is that channel mapping is now done in the I/O configuration data set (IOCDS) rather than through a screen on the Support Element. Within the IOCDS, *every* CHPID statement must now have an associated Physical Channel Identifier (PCHID). The only exceptions are for HiperSockets (IQD CHPIDs) and Internal Coupling Links (ICP CHPIDs).

Another major difference with the z990 is the requirement to include the definition of the multiple Logical Channel Subsystems within the IOCDS. The associated LCSSs must be defined in the following places:

- On the RESOURCE statement, to identify the LPARs associated with each LCSS
- On the CHPID PATH parameter, to identify the LCSS tor that CHPID
- On the CHPID access and candidate lists, to identify the LCSS for each LPAR
- On the CNTLUNIT PATH parameter (and LINK parameter), to identify the LCSS that each CHPID
- On the IODEVICE PARTITION parameter (for explicit device candidate lists) and PATH parameter, to identify the LCSS associated with each LPAR in the list

As you can imagine, this increases the size and complexity of the IOCP deck significantly.

Note: Not all installations use HCD to generate their IOCDS. Some prefer to create a standalone IOCP source file. This can still be done for the z990, but the task has become more complex.

The HCD support for the z990 has introduced a new concept: a *validated work IODF*. A validated work IODF contains all the I/O configuration definitions necessary to move forward to a production IODF, except for the physical channel identifier (PCHID) values. The easiest and recommended way of assigning PCHID values is with the CHPID Mapping Tool. The output from the CHPID Mapping Tool can then be combined with the validated work IODF, using the HCD migration option. This then completes the definition, allowing the IODF to be converted to a production IODF.

Note: In the past, many customers have been content to use the default channel assignments on the z800 and z900 and have not used the CHPID assignment function on these processors. *There is no default CHPID assignment on the z990.*

The remainder of this section discusses the process to create a z990-compatible IODF and IOCDS with HCD. The processes described here are aimed at:

- Defining completely new z990 environments. Migration from G5/G6 servers to z990 fall into this category as there is no upgrade path from G5/G6 to z990.
- Upgrading from an installed z900 to a z990.
- Consolidating an existing zSeries or S/390 environment into a new LCSS on an installed z990.
- Upgrading an installed z990 with additional channels.

Consolidation of multiple environments into a single LCSS is beyond the scope of the overview processes described here.

Multiple LCSS compatibility support

The scenarios described here may all have multiple LCSS defined. The z/OS *compatibility* support will allow z/OS to operate only in LCCS 0. The environments that will be supported in LCSS 1 are as follows:

- z/OS V1R4 with *exploitation* support installed
- z/VM V4R4
- Linux for zSeries at an appropriate level
- Coupling Facility logical partitions

Attention: If a Coupling Facility LPAR is installed in any LCSS other than LCSS 0, or has a partition id greater than X'F', then z/OS compatibility support must be installed on *all* members of the sysplex even if those members are not running on a z990.

5.3.1 Defining a new z990 environment

This process is very similar to that required to define a new z800 or z900 environment. The primary difference is the additional work required to define the Logical Channel Subsystems and the physical channel identifiers (PCHIDs) with the CHPID Mapping Tool. The process is summarized in Figure 5-1 on page 100 and described in the steps that follow.

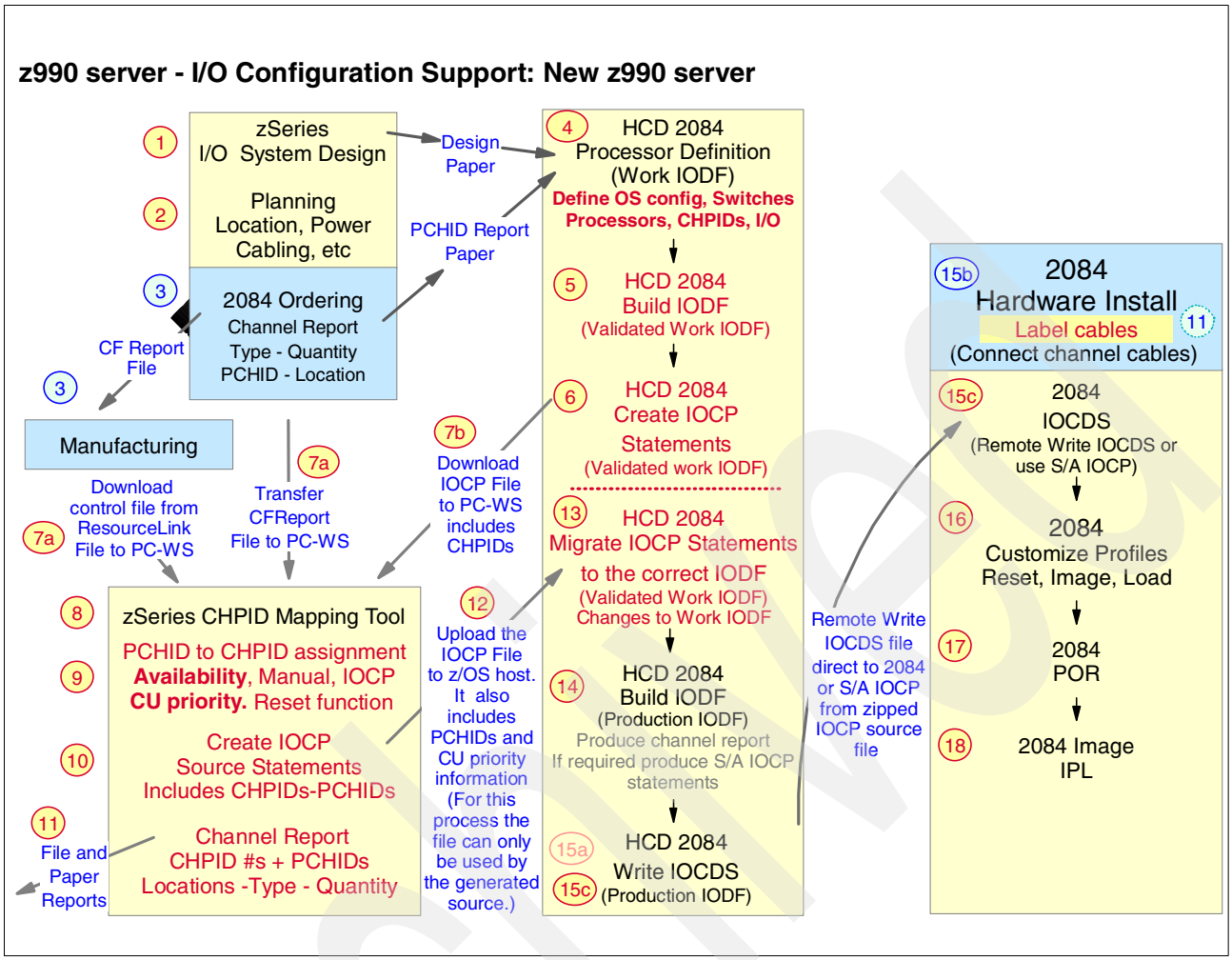


Figure 5-1 I/O configuration support: New z990 server

1. The I/O system design should be planned with your IBM field representative to cater for all your connectivity requirements. Decide on the number of Logical Channel Subsystems, number of partitions, and partition names that the target z990 processor is to support.
2. General planning, location, preliminary cabling planning, and configuration work should be completed with your IBM service representative.
3. Your IBM field representative will configure your z990 server using the IBM configurator (eConfig). This will be sent to manufacturing and the PCHID report will be supplied for reference.
4. Use HCD to create a new work IODF and define the z990 as a new processor with HCD option **1.3**. Alternatively, if you use a common IODF to contain the definitions for multiple processors, you should define a new z990 processor within the shared work IODF.

   At this point you should also define the Network name and CPC name of the z990. The Network name is the first part of the SNA address of the Support Element (SE). It is the 8-character alphanumeric network identifier of the LAN to which the SE is connected. The CPC name is the second part of the SNA address of the SE in the microprocessor cluster. If not specified, it will default to the processor id if a network name has been specified.

   Use the HCD **p** line command to define, change, and/or browse the Logical Channel Subsystem definitions for this processor. Be careful to specify the maximum number of devices carefully for each LCSS as this value can only be changed with a Power-on

Reset. If the value is defined too low, then you will be limited in the number of additional devices that you can define dynamically. Specifying too high a value will cause unnecessary HSA storage to be allocated.

You should then define the logical partitions to each LCSS. The **p** line command is used against each LCSS to display the list of logical partitions currently defined to it. From this display, partitions can be added, modified or deleted. Each partition added needs to have an MIF Image id assigned to it. The MIF Image id must be in the range of X'1' to X'F' and be unique to each partition defined within that LCSS. The MIF Image id is specified in the Partition Number field when the LPAR is defined or modified.

The next step is to define the CHPIDs for the I/O configuration. Again, this is done for each LCSS in turn. It is possible for CHPIDs in two different LCSSs to have the same CHPID number. *If you plan to use the Mapping Tool, do not assign a PCHID value to the CHPID at this stage.* If you do not plan to use the Mapping Tool, you should assign a PCHID to each CHPID. You could use the PCHID report from your order to do this.

At this point you would complete the control unit, I/O device, and OS configuration definitions as for a zSeries or S/390. The only difference to this procedure is with the path definitions for each control unit. During their definition, you must specify the channel paths for the control unit for *each* Logical Channel Subsystem. Fortunately, the LCSSs on each z990 appear simply as separate SMP processors - the name shown in the list is the Processor Id appended with the LCSS number (for example, `BIGTREX1.0` and `BIGTREX1.1`). In this respect, defining the control units to multiple LCSSs is no different than defining them to multiple processors in the same IODF.

5. The work IODF now contains the definitions for the processor, channel subsystems, logical partitions, control units, I/O devices, and operating system configurations. Assuming you did not assign PCHID values, it still requires the physical channel identifiers (PCHIDs) to be assigned to the CHPIDs. This is done with a validated work IODF in conjunction with the CHPID Mapping Tool. To convert the current work IODF into a validated work IODF you should use HCD option **2.12**, which is new.

   It is important that no changes other than the assignment of the PCHIDs are now made to this IODF; if other changes are made, it will lose its validated status. (A validated work IODF has an internal token assigned to it. This token is transferred to the CHPID Mapping Tool and later imported again with the PCHID definitions. The imported token must still match that in the validated IODF. If a validated IODF is changed and then revalidated, a new token will be assigned to it.)

6. Using the validated work IODF, you should select HCD option **2.3** to build an IOCP source statement data set. With previous levels of HCD, it was only possible to generate an IOCP data set from a production IODF. The latest level of HCD allows these data sets to be created from a validated work IODF also. However, these IOCP data sets are only usable at this stage by the CHPID Mapping Tool; it is not possible to use them for a z990 Power-on Reset. Only IOCP source from a *production* IODF can be used to build an IOCDS.

   Important: Once you have built the validated work IODF, do not make any changes to the IODF prior to importing the CMT IOCP statements and building a production IODF.

7. Prior to invoking the CHPID Mapping Tool, you need to have two files available:

   a. Either the Hardware Configuration File (.HWC extension) for this physical processor that was created during manufacturing and downloaded from Resource Link, or the CFReport file (.CFR extension) created when this processor was configured by your IBM technical support team.[1]

b. The IOCP source statement data set created in step 6. This data set needs to be downloaded to a PC workstation as a text file, using your favorite 3270 file transfer software.

8. The CHPID Mapping Tool (CMT) is used to assign physical channel identifiers (PCHIDs) to all physical channels. At this point, the tool may flag that there are channel hardware types for resolution. For example, there may be CHPIDs that support more than one available channel type. A check box is displayed to match the correct channel feature to each CHPID.

9. The availability option of the Mapping Tool places the highest priority on configuring paths to multiple path control units across books, then MBAs, followed by STIs, and finally across I/O cards. The CMT allows priority to be assigned (from 0001 to 9999) for each control unit in the IOCP deck. More than one CU can be assigned the same priority. Assigning the same priority to more than one CU means that these units will be mapped together for availability. When this is complete, the "Process CU priority" button can be used to perform the availability mapping.

Note: Three options are presented:

- Reset CHPIDs by Availability
- Reset CHPIDs assigned by Manual remap
- Reset CHPIDs assigned by IOCP

You can choose to reset any, none, or all three of these categories of assigned CHPIDs. Selecting none of the options will only process the unassigned CHPIDS.

Important: Be aware that, for an established processor, choosing to reset CHPIDs assigned by IOCP may require you to recable your hardware.

10. The CMT should now be used to create the updated IOCP file, which now includes PCHIDs. The option to generate this file is available from the "Tool" pull-down menu. The file should be saved on the workstation.

11. The channel reports should now be generated by the CMT. This will reflect CHPIDs by CSS with their assigned PCHIDs

Important: It is vitally important that the reports created by the CHPID Mapping Tool are saved and given to the installation team for the z990. We recommend that you produce *all* of the available reports, as they are cross-referenced in different ways.

12. The IOCP should be transferred back to the z/OS host where HCD is being run.

Important: With z/OS, this file should be transferred into an existing data set with fixed length, 80-byte records (RECFM=FB,LRECL=80), or it will not be accepted by HCD.

13. The next step is to merge the PCHID values assigned by the CHPID Mapping Tool with the IODF being built. This is accomplished with the HCD migration function in HCD option **5.1**. If your IODF contains more than one processor definition, then you will need to identify the correct one in the "Processor Id" field. The data set that contains the output from the CHPID Mapping Tool should be specified in the "`IOCP only input data set`" field. The processing mode should be set to **2** (Save) and the migration option should be set to **3** (PCHIDs).

[1] An IBM tool known as *eConfig* is used for this process. This tool is used by IBM and IBM Business Partner personnel.

Once the PCHID values assigned by the CHPID Mapping Tool have been merged with the IODF, it needs to be re-validated with HCD option **2.12**.

You should now print a CTC connection report. If you have any FICON CTCs that loop back (that is, both ends of the FICON CTC connection are on the same processor) either via direct connection or via a switch, then you may need to change the CUADD value on the control unit definitions if the target is an LPAR that resides in a LCSS other than LCSS 0. The high-order 4 bits of the CUADD field should reflect the target LCSS. The low-order 4 bits should reflect the MIF Image id of the target LPAR. The CUADD value is displayed in the Logical Address field on the HCD screen and can be changed with HCD option **1.4.c**.

Important: Be aware that you will also have to do a similar change for the control unit CUADD value on any other processors that will communicate with the z990 over FICON CTC links.

14. The work IODF can now be used to create a Production IODF with HCD Option **2.1**.
15. When the hardware is installed, the IOCDS can be created for the target z990 processor. If the new z990 has been connected as part of a microprocessor cluster and has been defined with the network name and CPC name coded in the processor definition, the IOCDS can be created on the target machine using HCD option **2.2**. If the new z990 is not part of a microprocessor cluster, the IOCP source statements need to be created with HCD option **2.3** and transferred to the HMC.

    Traditionally this has been done with diskettes; however, some large I/O configurations may exceed the capacity of a diskette. If this situation should arise, the file should be transformed into a zip file and stored on a diskette. The diskette can then be used as input to the standalone IOCP program on the z990 for creation of the IOCDS. (Alternately, you could ftp the file to the new HMC.)

    Tip: We recommend that you define and connect your z900 and z990 processors as part of a microprocessor cluster to allow an IOCDS to be created directly from any system without the need for manual file transfer.

16. The Reset, Image and Load profiles for the z990 operational environment can be customized from the HMC. When the Image profiles are created, care should be taken to assign the correct *LPAR id* to each logical partition. The LPAR id is not the same as the MIF Image id assigned in the HCD partition definition. The LPAR id is used primarily by the STSI and STIDP instructions (usually for ISV software licensing) and also by the CFRM support in z/OS for identifying the CFCC partitions. You might want to consider establishing a naming convention for your LPAR ids, as suggested in “LPARs” on page 66.
17. Perform a Power-on Reset (POR) of the z990 server.
18. Images on the z990 server can now be IPLed.

5.3.2 Upgrading an existing z900 to a z990 processor

This procedure documents a recommended process that you should follow if you are upgrading from z900 to a z990 processor. The process is summarized in Figure 5-2 on page 104 and described in the steps that follow.

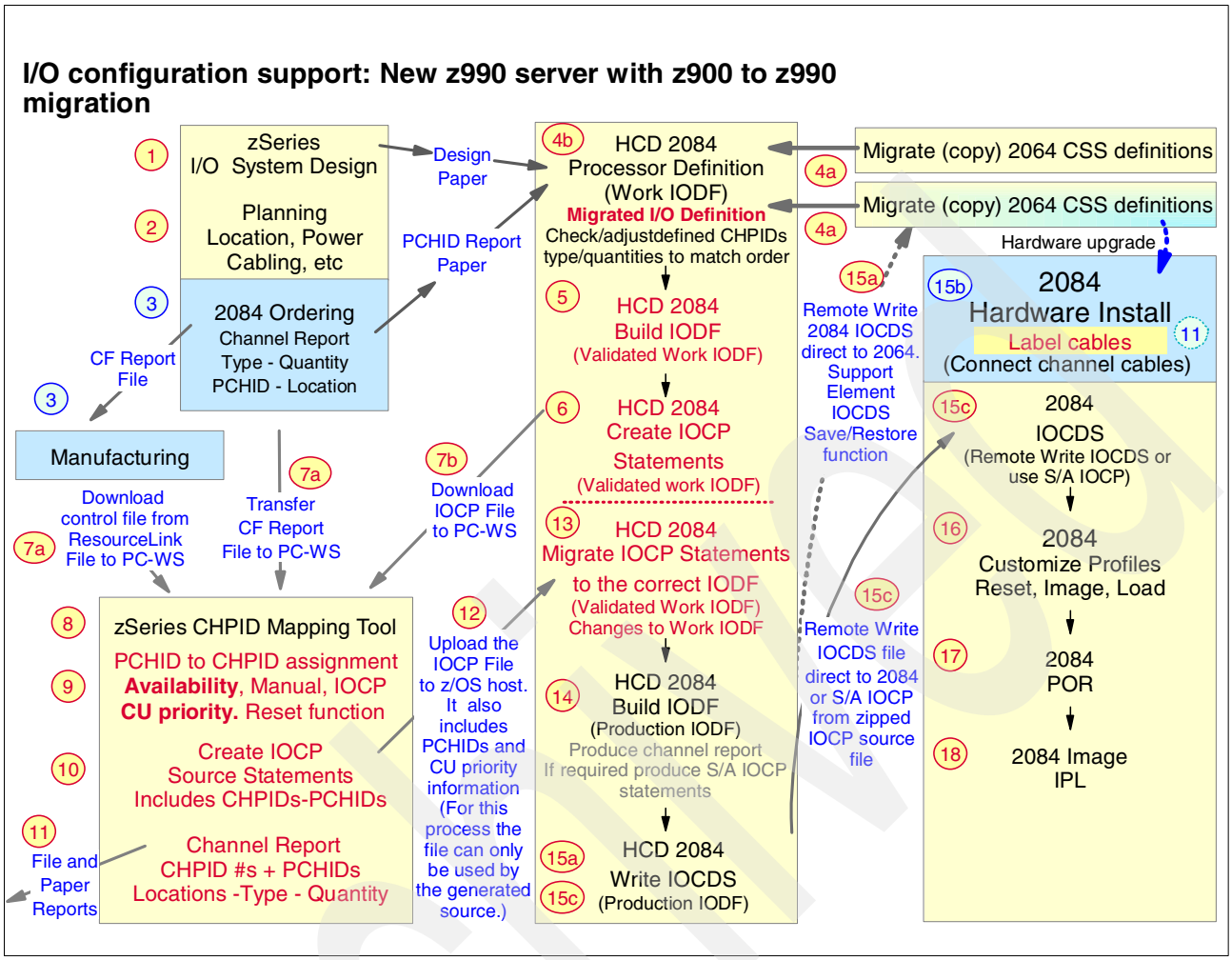


Figure 5-2 I/O configuration support: New z990 server with z900 to z990 migration

1. The I/O system design should be planned with your IBM field representative to cater for all your connectivity requirements. Decide on the number of Logical Channel Subsystems, number of partitions, and partition names that the target z990 processor is to support.
2. Planning, location, cabling, and configuration work should be completed with your IBM service representative.
3. Your IBM field representative will configure your z990 server using the IBM configurator (eConfig). This will be sent to manufacturing and the PCHID report will be supplied for reference.
4. Create a new work IODF (from your current production IODF) that contains the definitions for the installed z900. At this point, you need to decide whether the target machine will retain the identity of the installed z900 (that is, the Processor ID and CPCname), or if it will have a new identity. If you retain the original name, then you have the ability to write the IOCDS data set before the upgrade. If you do not retain the processor identity, then a remote write of the IOCDS (from another processor in the microprocessor cluster) will only be possible once the upgrade has been completed. Another advantage of retaining the Processor Id across the upgrade is that there will be no change to the CPC icon on your HMCs. (This may be a convenience for your operations staff). To retain the Processor Id, you must first delete the base z900 from the *new* work IODF (using HCD option **1.3.d)** in order to allow the z990 to be defined with the same name. A Processor Id must be unique within an IODF.

Use HCD to define the z990 as a new processor, using HCD option **1.3**. If you intend to keep the Processor Id of the existing z900, you should ensure that it, the Network Id, and the CPCname match that specified in the original IODF.

Use the HCD **p** line command to define, change, and/or browse the Logical Channel Subsystem definitions for this processor. Be careful to specify the maximum number of devices carefully for each LCSS, as this value can only be changed with a Power-on Reset. If the value is defined too low, then you will be limited in the number of additional devices that you can define dynamically. Specifying too high a value will cause unnecessary HSA storage to be allocated.

You should now switch to the production IODF that contains the definition for the installed z900. Use HCD option **1.3.y** to copy its channel subsystem definition into the newly defined z990 in the work IODF. You will need to specify the target work IODF data set name, the Processor Id, and the Logical Channel Subsystem number into which it is to be copied. HCD will copy all of the channel path connections, partition, switch, control unit, and I/O device definitions into the work IODF.

You should now print a CTC connection report. If you have any FICON CTCs that loop back (that is, both ends of the FICON CTC connection are on the same processor) either via direct connection or via a switch, then you may need to change the CUADD value on the control unit definitions if the target is an LPAR that resides in a LCSS other than LCSS 0. The high-order 4 bits of the CUADD field should reflect the target LCSS. The low-order 4 bits should reflect the MIF Image id of the target LPAR. The CUADD value is displayed in the Logical Address field on the HCD screen and can be changed with HCD option **1.4.c**.

Important: Be aware that you will also have to do a similar change for the control unit CUADD value on any other processors that will communicate with the z990 over shared FICON CTC links.

5. The work IODF now contains the definitions for the processor, channel subsystems, logical partitions, control units, I/O devices, and operating system configurations. Assuming you did not assign PCHID values, it still requires the physical channel identifiers (PCHIDs) to be assigned to the CHPIDs. This is done with a validated work IODF in conjunction with the CHPID Mapping Tool. To convert the current work IODF into a validated work IODF you should use HCD option **2.12**, which is new. It is important that no changes other than the assignment of the PCHIDs are now made to this IODF; if other changes are made, it will lose its validated status. (A validated work IODF has an internal token assigned to it. This token is transferred to the CHPID Mapping Tool and later imported again with the PCHID definitions. The imported token must still match that in the validated IODF. If a validated IODF is changed and then revalidated, a new token will be assigned to it.)
6. Using the validated work IODF, you should select HCD option **2.3** to build an IOCP source statement data set. With previous levels of HCD it was only possible to generate an IOCP data set from a production IODF. The latest level of HCD allows these data sets to be created from a validated work IODF also. However, these IOCP data sets are only usable at this stage by the CHPID Mapping Tool; it is not possible to use them for a z990 Power-on Reset. Only IOCP source from a *production* IODF can be used to build an IOCDS.

Important: Once you have built the validated work IODF, do not make any changes to the IODF prior to importing the CMT IOCP statements and having built a production IODF.

7. Prior to invoking the CHPID Mapping Tool, you need to have two files available:
   a. Either the Hardware Configuration File (.HWC extension) for this physical processor that was created during manufacturing and downloaded from Resource Link, or the CFReport file (.CFR extension) created when this processor was configured by your IBM technical support team.[2]
   b. The IOCP source statement data set created in step 6. This data set then needs to be downloaded to a PC workstation as a text file, using your favorite 3270 file transfer software.
8. The CHPID Mapping Tool (CMT) is used to assign physical channel identifiers (PCHIDs) to all physical channels. At this point, the tool may flag that there are channel hardware types for resolution. For example, there may be CHPIDs that support more than one available channel type. A check box is displayed to match the correct channel feature to each CHPID.
9. The availability option of the Mapping Tool places the highest priority on configuring paths to multiple path control units across books, then MBAs, followed by STIs, and finally across I/O cards. The CMT allows priority to be assigned (from 0001 to 9999) for each control unit in the IOCP deck. More than one CU can be assigned the same priority. Assigning the same priority to more than one CU means that these units will be mapped together for availability. When this is complete, the "Process CU priority" button can be used to perform the availability mapping.

   Note: Three options are presented:

   - Reset CHPIDs by Availability
   - Reset CHPIDs assigned by Manual remap
   - Reset CHPIDs assigned by IOCP

   You can choose to reset any, none, or all three of these categories of assigned CHPIDs. Selecting none of the options will only process the unassigned CHPIDS.

   Important: Be aware that, for an established processor, choosing to reset CHPIDs assigned by IOCP may require you to recable your hardware.

10. The CMT should now be used to create the updated IOCP file, which now includes PCHIDs. The option to generate this file is available from the "Tool" pull-down menu. The file should be saved on the workstation.
11. The channel reports should now be generated by the CMT. This will reflect CHPIDs by CSS with their assigned PCHIDs

    Important: It is vital that the reports created by the CHPID Mapping Tool are saved and given to the installation team for the z990. These reports can also be used to help label the cables with PCHID numbers. We recommend that you produce *all* of the available reports, as they are cross-referenced in different ways.

12. The IOCP should be transferred back to the z/OS host where HCD is being run.

    Important: With z/OS, this file should be transferred into an existing data set with fixed length, 80-byte records (RECFM=FB,LRECL=80), or it will not be accepted by HCD.

[2] An IBM tool known as *eConfig* is used for this process. This tool is used by IBM and IBM Business Partner personnel.

13. The next step is to merge the PCHID values assigned by the CHPID Mapping Tool with the IODF being built. This is accomplished with the HCD migration function in HCD option **5.1**. If your IODF contains more than one processor definition, then you will need to identify the correct one in the “Processor Id” field. The data set that contains the output from the CHPID Mapping Tool should be specified in the “IOCP only input data set” field. The processing mode should be set to **2** (Save) and the migration option should be set to **3** (PCHIDs).

    Once the PCHID values assigned by the CHPID Mapping Tool have been merged with the IODF, it needs to be re-validated with HCD option **2.12**.

14. The work IODF can now be used to create a Production IODF with HCD Option **2.1**.

15. The hardware and IOCDS now need to be established for the z990 processor.

    If this is an upgrade to an existing z900 and this is not part of a microprocessor cluster, a new option has been created to allow you to write the z990 IOCDS to the existing z900 prior to the hardware being changed. This is known as the IOCDS Save/Restore function.

    Initially, the z990 IOCDS is written to the z900 using HCD option **2.2**. The IOCDS will be shown as INVALID. This is then saved to the hard disk of the HMC. Following the hardware upgrade, the IOCDS can be restored to the Support Element using the same tool. The status is now changed to VALID and you can tailor your RESET, IMAGE and LOAD profiles (step 16). Alternatively, this can be performed by standalone IOCP (see step 15 c).

    c. The hardware installation can now be performed.

    d. If the new z990 has been connected as part of a microprocessor cluster and has been defined with the network name and CPC name coded in the processor definition, the IOCDS can be created on the target machine using HCD option **2.2**. If the new z990 is not part of a microprocessor cluster, the IOCP source statements need to be created with HCD option **2.3** and transferred to the HMC.

       Traditionally this has been done with diskettes; however, some large I/O configurations may exceed the capacity of a diskette. If this situation should arise, the file should be transformed into a zip file and stored on a diskette. The diskette can then be used as input to the stand-alone IOCP program on the z990 for creation of the IOCDS.

    Tip: We recommend that you define and connect your z900 and z990 processors as part of a microprocessor cluster to allow an IOCDS to be created directly from any system without the need for manual file transfer.

16. The Reset, Image and Load profiles for the z990 operational environment can be customized from the HMC. When the Image profiles are created, care should be taken to assign the correct *LPAR id* to each logical partition. The LPAR id is not the same as the MIF Image id assigned in the HCD partition definition. The LPAR id is used primarily by the STSI and STIDP instructions (usually for ISV software licensing) and *also by the CFRM support* in z/OS for identifying the CFCC partitions.

17. Perform a Power -on Reset (POR) of the z990 server.

18. Images on the z990 server can now be IPLed.

5.3.3 Consolidating a zSeries or S/390 environment onto an installed z990

This process assumes that you have an already established z990 processor operational and now wish to consolidate another S/390 or zSeries environment onto it. It assumes that the existing workload is already running in LCSS 0. The procedure that follows documents the steps you will need to take to move the existing S/390 or z900 workload into LCSS 1. (The

work involved in consolidating a processor workload into an already used LCSS is much more complex and beyond the scope of this book). The process is summarized in Figure 5-3 and described in the steps that follow.

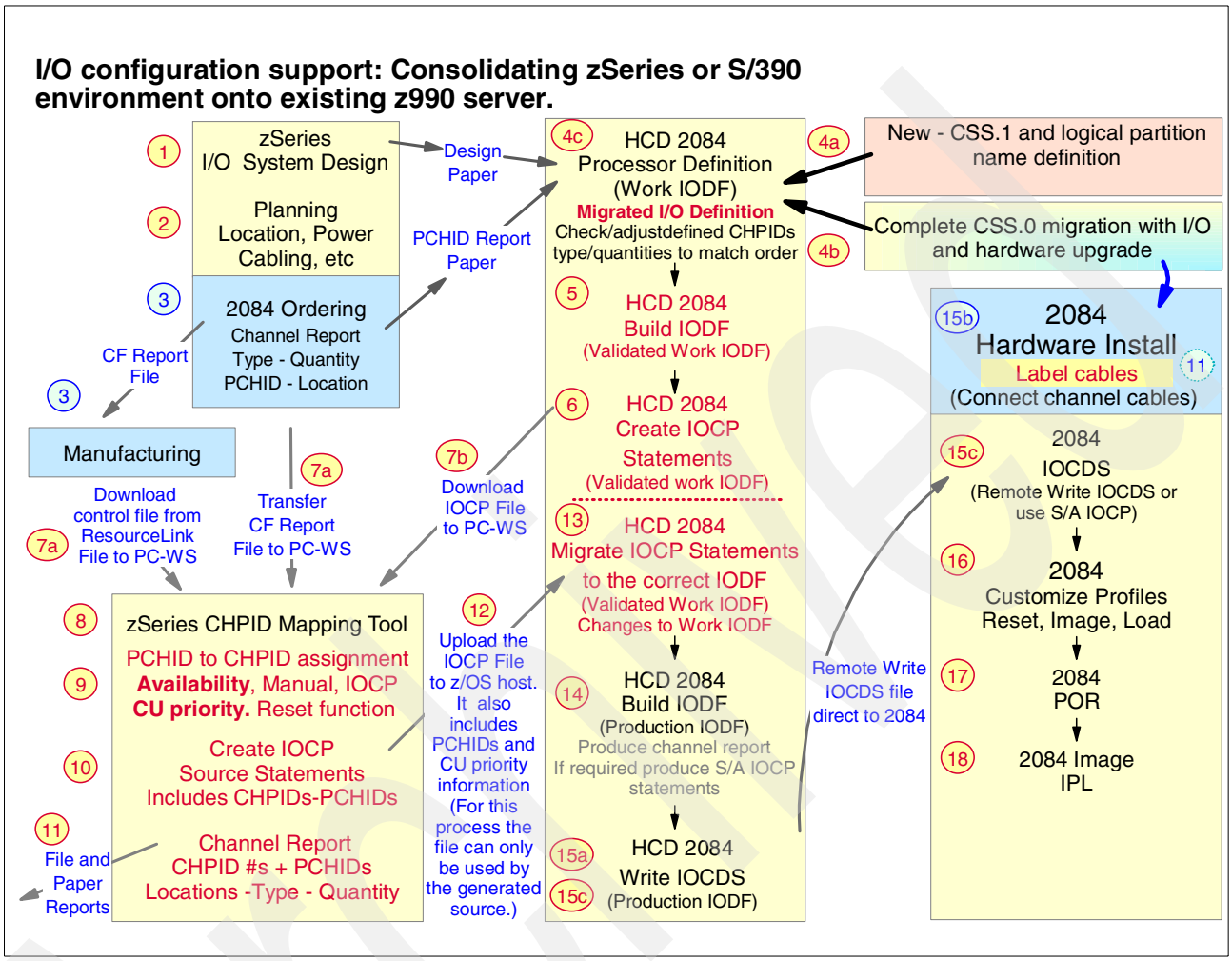


Figure 5-3 I/O configuration support: Consolidation of zSeries or S/390 environment to an established z990.

1. The modifications to your existing I/O system design should be planned with your IBM field representative to cater for all your connectivity requirements. Decide on the number of Logical Channel Subsystems, number of partitions, and partition names that the z990 server is to support.
2. Planning, location, cabling, and configuration work should be completed with your IBM service representative.
3. Your IBM field representative will configure the modifications to your z990 server using the IBM configurator (eConfig). This will be sent to manufacturing and the PCHID report will be supplied for reference.
4. Create a new work IODF from your current production IODF that contains the definition for the installed z990. This work IODF will contain the definitions for the merged environment.

   Use HCD option **`1.3.s`** to define the new Logical Channel Subsystem if it is not already defined. Be careful to specify the maximum number of devices carefully for each LCSS, as this value can only be changed with a Power-on Reset. If the value is defined too low, then you will be limited in the number of additional devices that you can define dynamically. Specifying too high a value will cause unnecessary HSA storage to be allocated.

You should now switch to the production IODF that contains the definition for the z900 or S/390 environment that you want to consolidate. Use HCD option **1.3.y** to copy its channel subsystem definition into the target z990 in the work IODF. You will need to specify the target work IODF data set name, the Processor Id, and the Logical Channel Subsystem number into which it will be copied. It will copy all of the channel path connections, switch, control unit, partition, and I/O device definitions into the target work IODF. If there is any duplication of LPAR names (that is, an LPAR with the same name is already defined in LCSS 0), HCD will display a screen showing the conflicts.

At this point you will have the option of renaming the LPARs being added. You can only use this screen to change the names of the LPARs being added. If you need to change the names of the existing LPARs on the installed z990, you should cancel out of this step, make the changes to the LPAR names in the existing z990 in the work IODF with HCD Option **1.3.s.p.c,** and then redo this whole step of the procedure.

Important: You should consider very carefully the implications of changing an LPAR name. At the very least, it will affect your operations staff documentation and training, HMC profiles, and RMF reports.

If the OS configuration(s) needed for the existing z900 or S/390 are not already defined in the new work IODF, they will need to be copied separately. From the original production IODF, select option **1.1.r** to repeat (copy) them. You will need to specify the new work IODF in the HCD “Target IODF name” field.

You should now print a CTC connection report. If you have any FICON CTCs that loop back (that is, both ends of the FICON CTC connection are on the same processor) either via direct connection or via a switch, then you may need to change the CUADD value on the control unit definitions if the target is an LPAR that resides in a LCSS other than LCSS 0. The high-order 4 bits of the CUADD field should reflect the target LCSS. The low-order 4 bits should reflect the MIF Image id of the target LPAR. The CUADD value is displayed in the “Logical Address” field on the HCD screen and can be changed with HCD option **1.4.c**.

Important: Be aware that you will also have to do a similar change for the control unit CUADD value on any other processors that will communicate with the z990 over shared FICON CTC links.

5. The work IODF now contains the definitions for the processor, channel subsystems, logical partitions, control units, I/O devices, and operating system configurations. Assuming you did not assign PCHID values, it still requires the physical channel identifiers (PCHIDs) to be assigned to the CHPIDs. This is done with a validated work IODF in conjunction with the CHPID Mapping Tool. To convert the current work IODF into a validated work IODF use HCD option **2.12**, which is new.

   It is important that no changes other than the assignment of the PCHIDs are now made to this IODF; if other changes are made, it will lose its validated status. (A validated work IODF has an internal token assigned to it. This token is transferred to the CHPID Mapping Tool and later imported again with the PCHID definitions. The imported token must still match that in the validated IODF. If a validated IODF is changed and then revalidated, a new token will be assigned to it.)

6. Using the validated work IODF, select HCD option **2.3** to build an IOCP source statement data set. With previous levels of HCD, it was only possible to generate an IOCP data set from a production IODF. The latest level of HCD allows these data sets to be created from a validated work IODF also. However, these IOCP data sets are only usable at this stage

by the CHPID Mapping Tool; it is not possible to use them for a z990 Power-on Reset until they contain all the PCHID assignments for the CHPIDs.

Important: Once you have built the validated work IODF, do not make any changes to the IODF prior to importing the CMT IOCP statements and having built a production IODF.

7. Prior to invoking the CHPID Mapping Tool, you need to have two files available:
   a. Either the updated Hardware Configuration File (.HWC extension) for this physical processor that was created during manufacturing and downloaded from Resource Link, or the new CFReport file (.CFR extension) created when this processor was re-configured by your IBM technical support team.[3]
   b. The IOCP source statement data set created in step 6. This data set then needs to be downloaded to a PC workstation as a text file, using your favorite 3270 file transfer software.
8. The CHPID Mapping Tool (CMT) is used to assign physical channel identifiers (PCHIDs) to all physical channels. At this point, the tool may tell you that there are channel hardware types for resolution. For example, there may be CHPIDs that support more than one available channel type. A check box is displayed to match the correct channel feature to each CHPID.
9. The availability option of the Mapping Tool places the highest priority on configuring paths to multiple path control units across books, then MBAs, followed by STIs, and finally across I/O cards. The CMT allows priority to be assigned (from 0001 to 9999) for each control unit in the IOCP deck. More than one CU can be assigned the same priority. Assigning the same priority to more than one CU means that these units will be mapped together for availability. When this is complete, the “Process CU priority” button can be used to perform the availability mapping.

   Note: Three options are presented:

   - Reset CHPIDs by Availability
   - Reset CHPIDs assigned by Manual remap
   - Reset CHPIDs assigned by IOCP

   You can choose to reset any, none, or all three of these categories of assigned CHPIDs. Selecting none of the options will only process the unassigned CHPIDS.

   Important: Be aware that, for an established processor, choosing to reset CHPIDs assigned by IOCP may require you to recable your hardware.

10. The CMT should now be used to create the updated IOCP file, which now includes PCHIDs. The option to generate this file is available from the “Tool” pull-down menu. The file should be saved on the workstation.
11. The channel reports should now be generated by the CMT. This will reflect CHPIDs by CSS with their assigned PCHIDs.

    Important: It is vital that the reports created by the CHPID Mapping Tool are saved and given to the installation team for the z990. We recommend that you produce *all* of the available reports, as they are cross-referenced in different ways.

[3] An IBM tool known as *eConfig* is used for this process. This tool is used by IBM and IBM Business Partner personnel.

12. The IOCP should be transferred back to the z/OS host where HCD is being run.

    Important: With z/OS, this file should be transferred into an existing data set with fixed length, 80-byte records (RECFM=FB,LRECL=80), or it will not be accepted by HCD.

13. The next step is to merge the PCHID values assigned by the CHPID Mapping Tool with the IODF being built. This is accomplished with the HCD migration function in HCD option **5.1**. If your IODF contains more than one processor definition, then you will need to identify the correct one in the “Processor Id” field. The data set that contains the output from the CHPID Mapping Tool should be specified in the “IOCP only input data set” field. The processing mode should be set to **2** (Save) and the migration option should be set to **3** (PCHIDs).

    Once the PCHID values assigned by the CHPID Mapping Tool have been merged with the IODF, it needs to be revalidated with HCD option **2.12**.

14. The work IODF can now be used to create a Production IODF with HCD Option **2.1**.

15. When the hardware is installed, the IOCDS can be created for the target z990 processor. The IOCDS can now be written to the Support Element with HCD option **2.2**.

16. The Reset, Image and Load profiles for the new z990 images can be customized from the HMC. When the Image profiles are created, care should be taken to assign the correct *LPAR id* to each logical partition. The LPAR id is not the same as the MIF Image id assigned in the HCD partition definition. The LPAR id is used primarily by the STSI and STIDP instructions (usually for ISV software licensing) and *also by the CFRM support* in z/OS for identifying the CFCC partitions.

17. Perform a Power-on Reset (POR) of the z990 server.

18. Images on the z990 server can now be IPLed.

5.3.4 Upgrading an installed z990 with additional channels

This process is very similar to that of installing additional channels on a z900, with the following exceptions:

- You now need to specify a Logical Channel Subsystem for each new channel.
- You need to decide on the CHPID addresses that are to be added (this step was also needed with the z900 if you previously used the CHPID assignment support).
- You can use the CHPID Mapping Tool to assign the PCHIDs to the CHPIDs.

The process to define the new channels for the installed z990 is summarized in Figure 5-4 on page 112 and described in the sections that follow. This process assumes use of the CMT for availability mapping (recommended). This is not mandatory as the new CHPIDs can be mapped to PCHIDs manually in the IODF based on the PCHID report supplied by your IBM field representative.

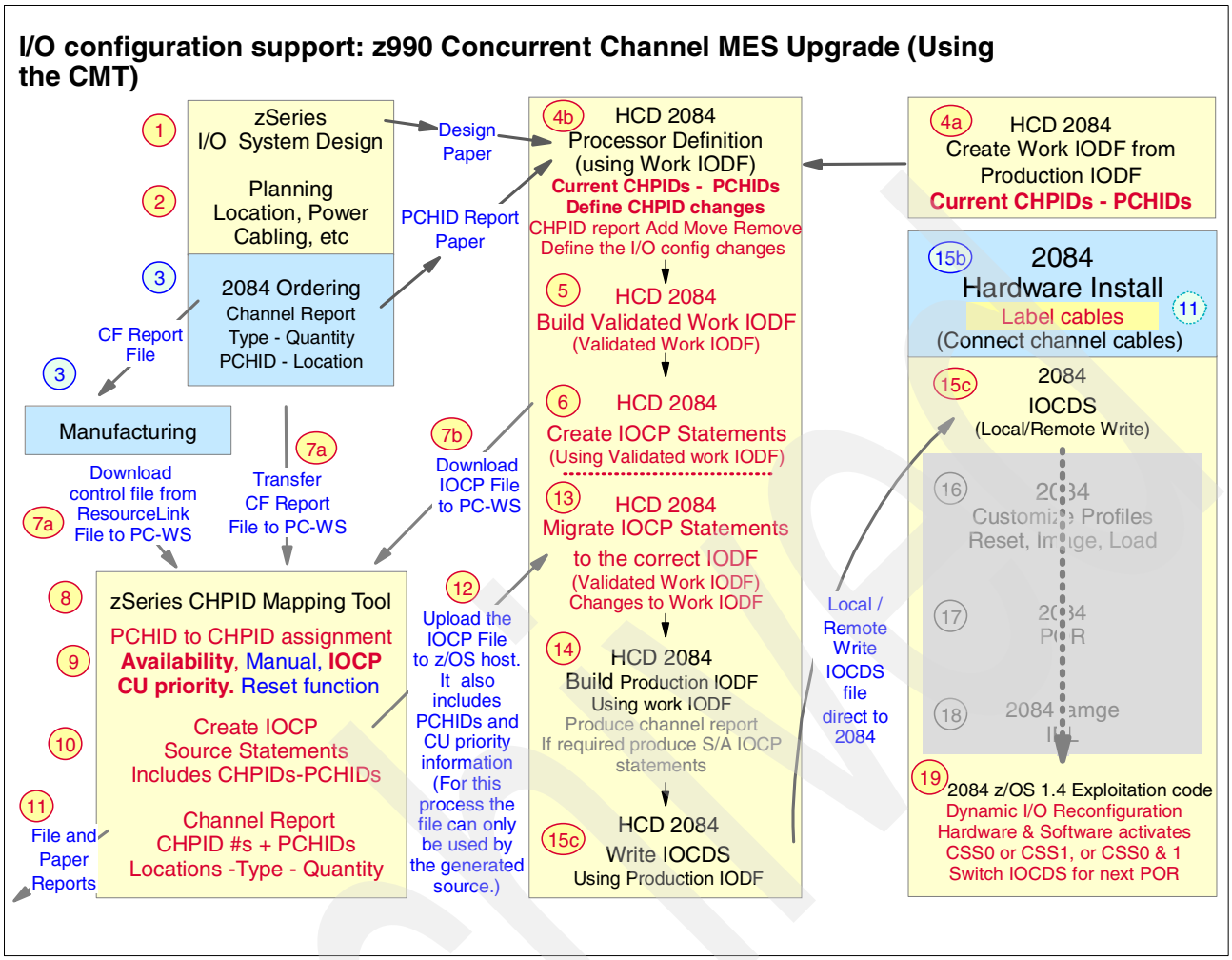


Figure 5-4 I/O configuration support: Upgrading an installed z990 server with additional channels

1. The additions to your existing z990 server should be planned with your IBM field representative to cater for all your connectivity requirements.
2. Planning, location, cabling, and configuration work should be completed with your IBM service representative.
3. Your IBM field representative will configure the modifications to your z990 server using the IBM configurator (eConfig). This will be sent to manufacturing and the PCHID report will be supplied for reference.
4. Create a new work IODF from your current production IODF that contains the definition for the installed z990.

   Use HCD Option **`1.3.s.s`** to add the new CHPIDs to the Logical Channel Subsystems that you have defined, as appropriate.

   Optionally, you may at this stage add or modify any control unit definitions and I/O device definitions for these new channels. You could also add the PCHID numbers and skip the use of the CMT; this can be practical when a relatively small number of channels are being added.
5. The work IODF must now be transformed into a *validated work IODF*. This is achieved with HCD option **`2.12`**.

Important: Once you have built the validated work IODF, do not make any changes to the IODF prior to importing the CMT IOCP statements and having built a production IODF.

6. An IOCP source statement data set must now be created with HCD option **2.2**.
7. Prior to invoking the CHPID Mapping Tool, you need to have two files available:
   a. Either the updated Hardware Configuration File (.HWC extension) for this physical processor that was created during manufacturing and downloaded from Resource Link, or the new CFReport file (.CFR extension) created when this processor was reconfigured by your IBM technical support team.
   b. The IOCP source statement data set created in step 6. This data set then needs to be downloaded to a PC workstation as a text file, using your favorite 3270 file transfer software.

   Important: This IOCP source statement data set must be transferred to the PC workstation as a text file.

8. The CHPID Mapping Tool (CMT) is now used to assign physical channel identifiers (PCHIDs) to the new physical channels. At this point, the Mapping Tool may flag that there are channel hardware types for resolution. For example, there may be CHPIDs that support more than one available channel type. A check box is displayed to match the correct channel feature to each CHPID.
9. The availability option of the Mapping Tool places the highest priority on configuring paths to multiple path control units across books, then MBAs, followed by STIs, and finally across I/O cards. The CMT allows priority to be assigned (from 0001 to 9999) for each control unit in the IOCP deck. More than one CU can be assigned the same priority. Assigning the same priority to more than one CU means that these units will be mapped together for availability. When this is complete, the “Process CU priority” button can be used to perform the availability mapping.

   Note: Three options are presented:

   - Reset CHPIDs by Availability
   - Reset CHPIDs assigned by Manual remap
   - Reset CHPIDs assigned by IOCP

   You can choose to reset any, none, or all three of these categories of assigned CHPIDs. Selecting none of the options will only process the unassigned CHPIDS.

   Important: Be aware that, for an established processor, choosing to reset CHPIDs assigned by IOCP may require you to recable your hardware.

10. The CMT should now be used to create the updated IOCP file, which now includes PCHIDs. The option to generate this file is available from the “Tool” pull-down menu. The file should be saved on the workstation.
11. The channel reports should now be generated by the CMT. This will reflect CHPIDs by CSS with their assigned PCHIDs.

Important: It is vital that the reports created by the CHPID Mapping Tool are saved and given to the installation team for the z990. We recommend that you produce *all* of the available reports, as they are cross-referenced in different ways.

12. The IOCP should be transferred back to the z/OS host where HCD is being run.

    Important: With z/OS, this file should be transferred into an existing data set with fixed length, 80-byte records (RECFM=FB,LRECL=80), or it will not be accepted by HCD.

13. The next step is to merge the PCHID values assigned by the CHPID Mapping Tool with the IODF being built. This is accomplished with the HCD migration function in HCD option **5.1**. If your IODF contains more than one processor definition, then you will need to identify the correct one in the “Processor Id” field. The data set that contains the output from the CHPID Mapping Tool should be specified in the “IOCP only input data set” field. The processing mode should be set to **2** (Save) and the migration option should be set to **3** (PCHIDs).

    Once the PCHID values assigned by the CHPID Mapping Tool have been merged with the IODF, it needs to be re-validated with HCD option **2.12**.

14. The work IODF can now be used to create a Production IODF with HCD Option **2.1**.
15. When the hardware is installed, the configuration can be implemented. If I/O Dynamic Reconfiguration is to be performed, skip to step 19. Otherwise, the IOCDS should now be written to the Support Element.
16. Make any required modifications to the Reset, Image and Load profiles of the z990 operational environment.
17. Perform a Power-on Reset of the z990 server.
18. IPL images on the z990 server.
19. If the configuration is to be implemented with Dynamic I/O reconfiguration, the correct level of software must be installed. See “Software considerations” on page 79 for details of the requirements.

    The HCD panels (option **2.3)** can now be used to perform the dynamic reconfiguration. Software activates should be performed for all LPARs. The final activation (for the last LPAR) is a software and hardware activate which should include writing the IOCDS and switching the IOCDS pointer.

5.4 LPAR definition notes

Storage granularity (“increment”) for CS and ES (in LPAR definitions) has several factors:

- The granularity for ES storage definition is 64 MB.
- The granularity for CS definitions depends on the larger of the *initial* or *reserved* storage for an LPAR.

  Using the *larger* of these two values:

  – For values less than or equal to 32GB, the increment size is 64 MB.
  – For values greater than 32 GB and less than or equal to 64 GB, the increment size is 128 MB.
  – For values greater than 64 GB, up to the maximum of 128 GB, the increment size is 256 MB.
- While HSA is not directly defined by the user, the granularity for HSA storage allocation is 64 MB.

The maximum LPAR size is 128 GB.

This information is required for Logical Partition Image definitions and for z/OS and OS/390 Reconfigurable Storage Unit definitions.

LPAR Dynamic Storage Reconfiguration

Dynamic Storage Reconfiguration (DSR) on z990 processors allows an operating system running in a logical partition to nondisruptively add its *reserved storage* amount to its configuration if any unused storage exists. This unused storage can be obtained when another logical partition releases some storage, or when a concurrent memory upgrade takes place. With enhanced DSR, the unused storage does not have to be contiguous.

When an operating system running in a logical partition assigns a storage increment to its configuration, LPAR will check if there are any free storage increments and will dynamically bring the storage online. LPAR will dynamically take offline a storage increment and will make it available to other partitions when an operating system running in a logical partition releases an increment of storage.

6

Frequently asked questions

Q: I cannot find a price for the additional Logical Channel Subsystem feature. Why?
A: It is inherent in the z990 hardware/firmware. There is no price associated with it and the capability is always present (and must be used).

Q: The z990 memory certainly appears as a NUMA design. Why is this term not used in your document?
A: Non-Uniform Memory Architecture (NUMA) is not a precisely-defined term. In common use, it often implies the ability (of application programs) to selectively use different speed memory areas (or, perhaps, different modes of memory access). This is not the case with the z990, where memory appears uniform to application programs.

Q: Does the operating system see any NUMA effects?
A: No.

Q: Are we required to use the CHPID Mapping Tool to set up a system?
A: No, you can enter PCHID numbers directly in HCD or in an IOCP deck.

Q: Can I use z/OS.e on a z990?
A: No.

Q: Can I move an I/O cage from my z900 system to a z990?
A: No.

Q: If I have a model B16 (two books) with a total of 10 CPs, are these balanced across the two books?
A: No. All eight customer-usable PUs in the first book are first used, then PUs are assigned to the next book, and so forth.

Q: If a complete book fails, will the system keep running? Can I restart in a degraded mode?
A: This would be an unsupported and untested environment. In the very unlikely event that a complete book fails, we would consider that the system is down.

Q: IBM seems to make a big deal of the *nondisruptive* upgrade functions. As I understand it, all a *disruptive* upgrade involves is an IPL. Is this correct?
A: You are partly correct. A more disruptive change might require a Power-on Reset (POR). However, in some environments an IPL or POR needs to be planned well in advance, and nondisruptive changes are important in these installations.

Q: We might need to run a very old operating system we keep in reserve. Can the z990 run in S/370 mode?
A: No.

Q: Can I "rotate" which PUs I use, so that the spare PUs are regularly used?
A: No, there is no provision for this.

Q: Are PCHID numbers fixed to hardware (or potential hardware) ports? Are they absolutely fixed values similar to an I/O slot number and jack number?
A: Almost. The only exception, at this time, is that a ESCON port sparing action (that is, moving a defined ESCON channel to a new jack on the same adapter card) will take the PCHID number with it so that no IOCDS changes are needed.

Q: How is the spare ESCON port used?
A: The 16-port ESCON cards reserve one port as a spare. By default it is the last port (connection J15) on the card. (Actually, any unactivated port on the ESCON card can be used as a spare.) The switchover process, in case of a failure, is not automatic. The *Repair&Verify* procedure directs the IBM Service Representative to manually move the ESCON cable from the failed port to the newly-activated port.

Q: Is the z990 "big endian" all the time? Even when running Linux?
A: Yes.

Q: Can the z990 use both EBCDIC and ASCII?
A: Yes. This is mostly a software issue. EBCDIC is "built in" for a few machine instructions related to numeric conversions and packed decimal data. It is also implicit in devices that are obviously character-oriented, such as printers and terminals. Otherwise, the z990 hardware (or any S/360, S/370, S/390, or z/Architecture machine) does not care whether the software uses ASCII or EBCDIC or a mixture of both.

Q: *Exactly* what functions or instructions are missing in an IFL to prevent MVS from running?
A: IBM does not document this.

Q: I understand I can run z/VM with an IFL engine. Can I run full z/VM facilities, such as VTAM operation, compilers, batch emulation, and so forth?
A: No. Only a subset of z/VM capabilities can be used. The subset includes CMS, but *excludes* VTAM, compile and link operations, batch operation, and a number of other functions. The optional, priced features Directory Maintenance Facility (DirMaint™), Performance Reporting Facility (PRF), RealTime Monitor (RTM), RACF® for z/VM, and the Performance Toolkit for VM may be ordered and used on IFLs. Some products with usage-based licenses may be used. In general, the intention is to limit z/VM functions to those reasonably needed to manage a Linux environment.

Q: Can I run both 32-bit and 64-bit versions of Linux? At the same time?
A: Yes, provided you have the appropriate Linux distributions. You can do this under z/VM, or by using separate LPARs—or with a combination of z/VM and LPARs.

Q: Can a Linux system share disks with z/OS?
A: For practical purposes, no. The most recent versions of Linux produce a disk label format that z/OS can tolerate, but there is no useful sharing of data at this time.

Q: Does Linux on the z990 use ASCII or EBCDIC?
A: ASCII.

Q: I am still confused about Linux for the z990. Where do I get it?
A: Not from IBM. At the time of writing, there are three mainline "commercial" Linux distributions that can be used with the z990. These are from SuSE, Red Hat, and Turbolinux. Several other organizations, such as Marist College, offer prebuilt Linux packages that can be used with the z990. You can also download all the source code (and patches) and build your own Linux kernel. IBM sells middleware, such as DB2, that can be installed on top of these Linux distributions.

Q: I cannot find the feature codes to order cables for the z990. Where are they?
A: They do not exist. Cables have become sufficiently complex that ordering by simple feature codes does not work very well. IBM uses a service offering to provide cables for the z990.

Q: Our local "experts" talk about *Passat* and *Cargo* and *Compatibility* I/O cages and I find this confusing. Does the z990 have these?
A: No. These were development names for z900 I/O cages. The *Passat* holds "old" adapters such as parallel channels and OSA-2 adapters; it is sometimes known as a *Compatibility* cage. The *Cargo* cage holds the newer adapters (such as are also used with the z990). During development, the z800 I/O cage was known as *Cargo Lite*; it is similar to the z900 *Cargo* cage, but smaller in some respects. These names were used during product development stages and are no longer meaningful.

Q: I have acquired several old IBM 9034 ("Pacer") boxes. Can I use these to attach parallel channel devices?
A: Yes.

Q: Can I use the Optica converters with z990 systems? With older S/390s?
A: Yes, as far as we know they should work with any ESCON channel. However, be aware that byte multiplexor devices *may* be a special case. You should obtain the latest information from Optica.

Q: Can I purchase ESCON and other fiber cables from non-IBM suppliers?
A: Yes, but make certain you get the right cables and the right connectors. The number of cable types and connectors has grown rapidly. Also, there are fiber optic cables on the market with different levels of quality. IBM fiber optic cables are qualified to an application specification and for reliability.

Q: Do I need conversion cables on both ends of my existing fiber cables?
A: It depends on what connectors you have on the adapters at both ends of the fiber optic cable. If you have an ESCON cable with ESCON Duplex connectors, you will need a conversion kit at the processor end. If the switch or device at the other end has ESCON Duplex connectors, you will not need a conversion kit at this end. With mode conditioning patch (MCP) cables, you will require one at each end when you connect long-wave (LX) channels and devices with multimode cable.

However, these should be seen an a temporary measure, at best. Conversion kits and mode conditioning cables for fiber optic cables have long-term concerns. They are messy and difficult to manage under a raised floor. They are an interruption, with some loss, and a potential point of failure in the light path. They are prone to breakage or interruption when cables are moved, especially when bulk movements are involved. MCP cables are not supported for data rates over 1 Bpbs.

Q: Should I stack the Optica converters under the floor? How should I arrange them?
A: Arrange them any way that is convenient, but provide some air flow around the units. You probably want to avoid major movements of existing parallel channel cables, and this will be the determining factor in placement.

Q: Can I use the Optica converters to connect a parallel channel to an ESCON control unit?
A: No. They provide a one-way conversion for connecting ESCON channels to parallel control units. (The same is true for the Pacer units.)

Q: I am a little confused about using my existing channel cables. Can you summarize the situation?
A: Cabling, especially for a large system, has become sufficiently complex that IBM has moved to the Services Offering (as announced with the z990) as a method of determining exactly what cables you need and then ordering the right ones. As a very brief summary:

- Existing parallel channels may be used, but only if you purchase converter boxes. You must use ESCON connections from the z990 to the converter boxes.
- Traditional ESCON cables (with the large duplex connector) cannot be connected to the z990. You need a conversion kit from the new connector (also used with z800 and z900 machines) to the traditional ESCON cable connector. Alternately, you could purchase new ESCON cables with the appropriate connectors. The control unit end of an ESCON cable still uses the large duplex connector.
- The earlier FICON cables (SC connectors) do not connect to the z990. A conversion kit is available to use these cables.
- The new Gigabyte Ethernet adapter uses a different fiber cable connector than the previous Gigabyte Ethernet adapter.

The conversion kits may be obtained as part of the Services Offering.

Q: Can a single 2074 be used with more than one Logical Channel Subsystem?
A: The two ESCON channels in the 2074 can be attached to two channel subsystems. A single ESCON channel (if directly connected to the 2074) can be used from only one channel subsystem. The use of an ESCON director allows greater sharing of a 2074 with a single channel.

Q: Can I define more CHPIDs (in HCD or IOCP) than I have physical channels on order?
A: You must specify a PCHID for every CHPID in order to create a working IOCDS. If you use the Channel Mapping Tool (which uses data from your system order as part of its input), you must have enough physical channels on order to supply a PCHID (corresponding to a physical channel) for every CHPID in your definition. However, if you manually specify PCHID values in your HCD or IOCP definitions (bypassing the CMT), you can specify PCHID values for channels that are not actually installed or on order.

Q: What is the difference between 32-bit and 31-bit operation?
A: They are the same. In this mode the registers are 32 bits long, but the maximum address is restricted to a 31-bit number. This design arose from compatibility concerns with programs using 24-bit addressing. The z990 can use 24-, 31-, or 64-bit addressing modes; it uses 32- or 64-bit general purpose registers.

Q: I cannot find the feature code or price for HiperSockets. Why?
A: They are basic parts of the system. No feature codes (or prices) are involved.

Q: I have an existing Ethernet structure that I want to use for SEs and HMCs. Can I use thin-wire (coax) Ethernet?
A: No. Neither the SEs nor the HMCs have provisions for attaching Ethernet coax or external Ethernet transceivers.

Q: Can I control my z990 from home, using the SE or HMC Web interfaces?
A: There are several factors to consider here, and the first is security. Do you want your system exposed on the Internet such that a simple userid/password is the only thing that prevents someone from taking over your machine? Assuming you accept the risk, there are two levels of Web control possible. The first level is a HMC subset that includes the *activate* and *load* (IPL) functions that are the basic elements of remote control. The second level is the Desktop on Call function that creates a complete image of the HMC desktop. A relatively large bandwidth is required to make this work well (mainly to reflect mouse pointer movements), and it is possible to overrun the connection bandwidth. However, with this connection you can perform any SE or HMC function remotely.

Q: My local "experts" talk about *driver levels*. What are these?
A: A *driver* is all the code that is loaded on a Support Element and HMC. This includes the visible Support Element functions plus all the code that is eventually loaded into the SAP, the cage controllers, CF LPARs, channel cards, and so forth.[1] Major internal upgrades to a z990 are made by installing a new driver. Minor upgrades and fixes are done by installing MCLs; these are replacements for various files in the current driver.

Q: You always describe LPARs. Can I run a z990 without LPARs?
A: No.

Q: Does the z990 have some "flashing lights" that provide a general sense of whether the system is doing anything?
A: No, sorry. However, you can use the System Activity Display (on the HMC) for a similar purpose.

Q: You did not discuss maintenance charges. These are an important element in the economics of purchasing a new system. Why?
A: Redbooks are not intended to discuss pricing. You should consult your IBM business partner or IBM representative for pricing details. However, at the risk of sounding like a marketing representative, we note that z990 maintenance charges are typically *much* less than the equivalent charges for older systems. This, combined with various strategies for software pricing, can produce a total cost (over three years, for example) that can be lower than the cost of keeping a fully-depreciated ("free") older system. The pricing of large computer systems, especially when third-party software is included, can be quite complex. We strongly suggest that you ask the appropriate marketing representatives for details instead of making your own assumptions about prices.[2]

Q: Why do I need ICKDSF Release 17 on all systems sharing DASD with a z990?
A: ICKDSF needs to recognize if another copy of ICKDSF (perhaps on another system) is currently accessing the same volume. This is permitted for media maintenance functions. It uses CPU ID information to control this sharing and the format of this information has changed for z990 systems. Release 17 recognizes both the old and new formats.

Q: What is the largest partition size supported?
A: 128 GB is the largest permitted partition size.

Q: Does it matter which book is connected to an I/O adapter? Does the operating system need to know this?
A: It does not matter (except possibly for high-availability planning) and an LPAR (or operating system) cannot sense this information.

Q: How are the number of LPARs related to the number of books?
A: There is no relation between the maximum number of LPARs and the number of books in

[1] These subsequent loads into various components are done during a Power-on Reset (POR) function.
[2] In the same way, we strongly suggest that you do not depend on the trade press for specific pricing information.

the system. (There may be a practical relationship, in terms of workload capacity, but there is no architectural relationship.)

Q: I am still confused about the different crypto options. Can you explain it a different way?
A: It might help to consider *secure crypto* (where secret keys are protected by special hardware and are not viewable in memory or anywhere else) and *non-secure crypto*[3] (where the secret keys are in the clear somewhere in programs, memory, and DASD).

The crypto assist instructions and the PCICA adapter deal with non-secure crypto and can perform any operation needed in this mode; the user must manage secret keys through programming. The PCIXCC adapter deals with secure crypto and has special hardware to conceal secret keys. Special functions (such as a TKE unit) are used to enter the secret keys. Some customers and applications (such as banking) usually demand secure crypto facilities. Other applications, such as some Web applications, may not need secure crypto facilities.

Q: Can memory for an LPAR span multiple books?
A: Yes.

Q: Do crypto adapters require a CHPID?
A: No.

Q: You do not say much about the “special” PCHIDs for ICB connections. How are these displayed?
A: The “0480” is the PCHID in these examples:

```
D M=CHP(F2)
IEE174I 16.10.15 DISPLAY M 569
CHPID F2:  TYPE=21, DESC=INTEGRATED CLUSTER BUS PEER, ONLINE
COUPLING FACILITY 002064.IBM.02.000000051515
                  PARTITION: 04  CPCID: 00
NAMED AQCF2       CONTROL UNIT ID: FFF5

SENDER PATH       PHYSICAL           LOGICAL       CHANNEL TYPE
  F2 / 0480        ONLINE             ONLINE        CBP

COUPLING FACILITY DEVICE     SUBCHANNEL    STATUS
                   FFCB         371A         OPERATIONAL/IN USE
                   FFCC         371B         OPERATIONAL/IN USE
                   FFCD         371C         OPERATIONAL/IN USE
                   FFCE         371D         OPERATIONAL/IN USE
                   FFCF         371E         OPERATIONAL/IN USE
                   FFD0         371F         OPERATIONAL/IN USE
                   FFD1         3720         OPERATIONAL/IN USE


D CF
IXL150I  16.11.06  DISPLAY CF 359
COUPLING FACILITY 002064.IBM.02.000000051515
                  PARTITION: 04  CPCID: 00
                  CONTROL UNIT ID: FFF5
NAMED AQCF2
COUPLING FACILITY SPACE UTILIZATION
 ALLOCATED SPACE                 DUMP SPACE UTILIZATION
  STRUCTURES:     131840 K        STRUCTURE DUMP TABLES:        0 K
  DUMP SPACE:       6144 K                  TABLE COUNT:        0
 FREE SPACE:      716288 K       FREE DUMP SPACE:           6144 K
TOTAL SPACE:      854272 K      TOTAL DUMP SPACE:           6144 K
                               MAX REQUESTED DUMP SPACE:        0 K
```

[3] These terms, secure crypto and non-secure crypto, are not standard terms.

```
   VOLATILE:         YES         STORAGE INCREMENT SIZE:      256 K
    CFLEVEL:          12
    CFCC RELEASE 12.00, SERVICE LEVEL 06.08
    BUILT ON 01/29/2003 AT 14:08:00

    CF REQUEST TIME ORDERING: NOT-REQUIRED AND ENABLED
COUPLING FACILITY SPACE CONFIGURATION
                          IN USE          FREE         TOTAL
CONTROL SPACE:          137984 K      716288 K      854272 K
NON-CONTROL SPACE:           0 K           0 K           0 K

SENDER PATH       PHYSICAL            LOGICAL       CHANNEL TYPE
  F2 / 0480        ONLINE              ONLINE        CBP

COUPLING FACILITY DEVICE     SUBCHANNEL     STATUS
                   FFCB         371A        OPERATIONAL/IN USE
                   FFCC         371B        OPERATIONAL/IN USE
                   FFCD         371C        OPERATIONAL/IN USE
                   FFCE         371D        OPERATIONAL/IN USE
                   FFCF         371E        OPERATIONAL/IN USE
                   FFD0         371F        OPERATIONAL/IN USE
```

Q: How can I determine (from a program) the number of active CPs in my z990?
A: The STSI instruction will return a decimal number (EBCDIC characters) in the range 301 to 348, where the last two digits indicate the number of active CPs. You may encounter informal references to *300 model numbers*, and this is the source of these numbers.

A

New and changed instructions

The following list contains the new and changed instructions. An asterisk (*) in the first column denotes a new instruction. The letters PRIV indicates a privileged instruction. These are all six-byte instructions with part of the operation code in the first byte and part in the last byte. The general changes are discussed in “Instruction set” on page 22.

	Format	Op codes	Mnemonic	Name....	Notes.....
*	RXY	E3 5A	AY	Add	32 bits
	RXY	E3 08	AG	Add	64 bits
	RXY	E3 18	AGF	Add	64<--32
*	RXY	E3 7A	AHY	Add halfword	16 bits
	RXY	E3 0A	ALG	Add logical	64 bits
	RXY	E3 1A	ALGF	Add logical	64<--32
*	RXY	E3 5E	ALY	Add logical	32 bits
	RXY	E3 98	ALC	Add logical with carry	32 bits
	RXY	E3 88	ALCG	Add logical with carry	64 bits
*	SIY	EB 54	NIY	AND	8 bits
	RXY	E3 80	NG	AND	64 bits
	RXY	E3 46	BCTG	Branch on count	64 bits
	RSY	EB 45	BXLEG	Branch on index low or equal	64 bits
	RSY	EB 44	BXHG	Branch on index high	64 bits
*	RRE	B9 2E	KM	Cipher message	
*	RRE	B9 2F	KMC	Cipher message with chaining	
*	RXY	E3 59	CY	Compare	32 bits
	RXY	E3 20	CG	Compare	64 bits
	RXY	E3 30	CGF	Compare	64:32
*	RSY	EB 14	CSY	Compare and swap	32 bits
	RSY	EB 30	CSG	Compare and swap	64 bits
*	RSY	EB 31	CDSY	Compare double and swap	32 bits
	RSY	EB 3E	CDSG	Compare double and swap	64 bits
*	RXY	E3 79	CHY	Compare halfword	16 bits
*	RXY	E3 55	CLY	Compare logical	32 bits
	RXY	E3 31	CLGF	Compare logical	64:32
	RXY	E3 21	CLG	Compare logical	64 bits
*	SIY	EB 55	CLIY	Compare Logical (immediate)	8 bits
*	RSY	EB 21	CLMY	Compare logical characters under mask (low)	
	RSY	EB 20	CLMH	Compare logical characters under mask (high)	
	RSY	EB 8F	CLCLU	Compare logical long unicode	
*	RRE	B9 3E	KIMD	Compute intermediate message digest	

*	RRE	B9 3F	KLMD	Compute last message digest	
*	RRE	B9 1E	KMAC	Compute message authentication code	
*	RXY	E3 06	CVBY	Convert to binary	32 bits
	RXY	E3 0E	CVBG	Convert to binary	64 bits
*	RXY	E3 26	CVDY	Convert to decimal	32 bits
	RXY	E3 2E	CVDG	Convert to decimal	64 bits
	RXY	E3 0D	DSG	Divide single	64 bits
	RXY	E3 1D	DSGF	Divide single	
	RXY	E3 97	DL	Divide logical	32<--64
	RXY	E3 87	DLG	Divide logical	64<--128
*	SIY	EB 57	XIY	Exclusive OR (immediate)	8 bits
	RXY	E3 82	XG	Exclusive OR	64 bits
*	RXY	E3 57	XY	Exclusive OR	32 bits
*	RXY	E3 73	ICY	Insert Character	8 bits
*	RSY	EB 81	ICMY	Insert characters under mask (low)	
	RSY	EB 80	ICMH	Insert characters under mask (high)	
	RXY	E3 04	LG	Load	64 bits
	RXY	E3 14	LGF	Load	64<--32
*	RXY	E3 58	LY	Load	32 bits
*	RXY	ED 65	LDY	Load	floating long
*	RXY	ED 64	LEY	Load	floating short
*	RSY	EB 9A	LAMY	Load access multiple	
*	RXY	E3 71	LAY	Load address	32 or 64 bits
*	RXY	E3 76	LB	Load byte	signed byte to 32 bits
*	RXY	E3 77	LGB	Load byte	signed byte to 64 bits
	RSY	EB 2F	LCTLG	Load control	64 bits PRIV
	RXY	E3 15	LGH	Load halfword	64<--16
*	RXY	E3 78	LHY	Load halfword	16 to 32 bits
	RXY	E3 16	LLGF	Load logical	64<--32
	RXY	E3 90	LLGC	Load logical character	64<--8
	RXY	E3 91	LLGH	Load logical halfword	64<--16
	RXY	E3 17	LLGT	Load logical 31 bits	
	RSY	EB 04	LMG	Load multiple	64 bits
*	RSY	EB 98	LMY	Load multiple	32 bit words
	RSY	EB 96	LMH	Load multiple high	
	RXY	E3 8F	LPQ	Load pair from quadword	
*	RXY	E3 13	LRAY	Load real address	32 bits PRIV
	RXY	E3 03	LRAG	Load real address	64 bits PRIV
	RXY	E3 1E	LRV	Load reversed	32 bits
	RXY	E3 0F	LRVG	Load reversed	64 bits
	RXY	E3 1F	LRVH	Load reversed	16 bits
*	SIY	EB 52	MVIY	Move (immediate)	8 bits
	RSY	EB 8E	MVCLU	Move long unicode	
	RXY	E3 86	MLG	Multiply logical	128<--64
	RXY	E3 96	ML	Multiply logical	64<--32
*	RXY	E3 51	MSY	Multiply single	32<--32x32
	RXY	E3 1C	MSGF	Multiply single	
	RXY	E3 0C	MSG	Multiply single	64 bits
*	SIY	EB 56	OIY	OR (immediate)	8 bits
	RXY	E3 81	OG	OR	64 bits
*	RXY	E3 56	OY	OR	32 bits
	RSY	EB 1C	RLLG	Rotate left single logical	64 bits
	RSY	EB 1D	RLL	Rotate left single logical	32 bits
	RSY	EB 0A	SRAG	Shift right single	64 bits
	RSY	EB 0B	SLAG	Shift left single	64 bits
	RSY	EB 0C	SRLG	Shift right single logical	64 bits
	RSY	EB 0D	SLLG	Shift left single logical	64 bits
*	RXY	ED 67	STDY	Store	floating long
*	RXY	ED 66	STEY	Store	floating short
*	RXY	E3 50	STY	Store	32 bits

```
  RXY     E3 24     STG     Store                          64 bits
* RSY     EB 9B     STAMY   Store access multiple          32 bit words
* RXY     E3 72     STCY    Store character
* RSY     EB 2D     STCMY   Store characters under mask (low)
  RSY     EB 2C     STCMH   Store characters under mask (high)
  RSY     EB 25     STCTG   Store control                  64 bits PRIV
* RXY     E3 70     STHY    Store halfword                 16 bits
  RSY     EB 24     STMG    Store multiple                 64 bits
  RSY     EB 26     STMH    Store multiple high
* RSY     EB 90     STMY    Store multiple
  RXY     E3 8E     STPQ    Store pair to quadword
  RXY     E3 2F     STRVG   Store reversed                 64 bits
  RXY     E3 3E     STRV    Store reversed                 32 bits
  RXY     E3 3F     STRVH   Store reversed                 16 bits
* RXY     E3 5B     SY      Subtract                       32 bit words
  RXY     E3 09     SG      Subtract                       64 bits
  RXY     E3 19     SGF     Subtract                       64<--32
* RXY     E3 7B     SHY     Subtract halfword              16 bit integers
  RXY     E3 0B     SLG     Subtract logical               64 bits
* RXY     E3 5F     SLY     Subtract logical               32 bit words
  RXY     E3 1B     SLGF    Subtract logical               64<--32
  RXY     E3 89     SLBG    Subtract logical with borrow   64 bits
  RXY     E3 99     SLB     Subtract logical with borrow   32 bits
* SIY     EB 51     TMY     Test under mask                8 bits
  RSY     EB 0F     TRACG   Trace                          64 bits PRIV
```

Not included in this list are:

- The STFL instruction, which has its output extended to indicate the presence of new facilities and assist functions
- The STIPD, described in the text
- The STCPS instruction, which functions as a NO OP on this system

The exact format and function of the instructions are explained in the Principles of Operation manual associated with the z990.

B

I/O adapter indicator lights

Adapter lights and indicators

The LEDs on the OSA-Express cards (and the FICON Express card) can be useful for various debugging purposes. Figure B-1 on page 130 illustrates the indicators present on the OSA Express cards. The A, B, and C status LEDs are common to all OSA-Express adapters and have the following meanings:

LED A (green)	LED B (amber)	Status
OFF	OFF	No power, or card processor is looping
OFF	FLASHING	Disgnostics are running
FLASHING	OFF	Tests are complete and CHPID is online
FLASHING	FLASHING	Hardware error detected. Not online.

LED C (green)	
OFF	CHPID is online and communicating
ON	CHPID is offline (for maintenance) or a wrap plug is present
RAPID FLASH	Power-on tests are running
FLASHING	CHPID is online

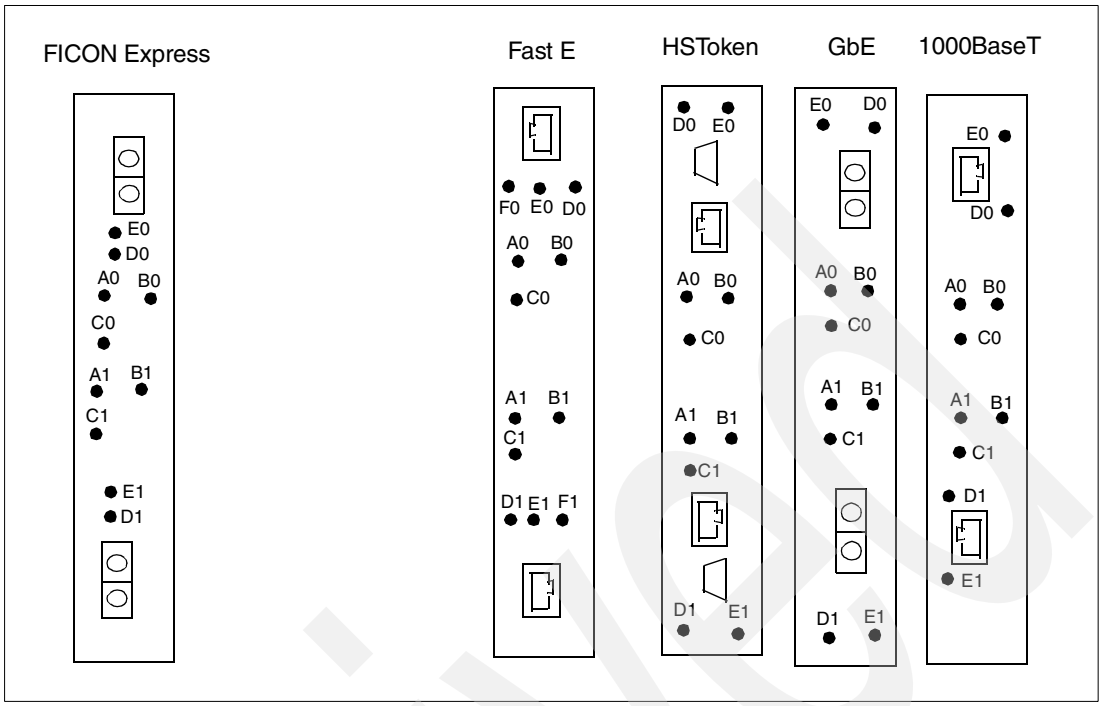


Figure B-1 Indicators on FICON Express and OSA-Express cards

OSA-Express Fast Ethernet

In addition to the common OSA Express status LEDs there are specific Fast Ethernet card indicators. These indicators have the following meanings:

```
LED D (amber)   Status
   ON           Sending or receiving data
   OFF          Idle

LED E (green)   Status
   ON           Operating at 100 Mpbs
   OFF          Operating at 10 Mpbs

LED F (green)   Status
   ON           Operating in full duplex
   OFF          Operating in half duplex
```

OSA-Express 1000BaseT Ethernet

This adapter provides gigabit Ethernet (as well as 100 Mbps and 10 Mpbs Ethernet) over copper cables. It replaces the OSA-Express Fast Ethernet adapter, and complements the OSA-Express Gigabit Ethernet adapter (which used fiber cables).

OSA-Express Gigabit Ethernet

In addition to the common OSA Express status LEDs there are specific Gigabit Ethernet card indicators, which have the following meanings:

```
LED D (amber)   LED E (green)   Status
   any             ON           Link is active
   ON              ON           Data activity on link
   ON              OFF          Error detected. Hardware is stopped.
   any             FLASHING     Internal code loaded, but card is disabled
```

OSA-Express High Speed Token Ring

In addition to the common OSA Express status LEDs there are specific OSA Express Token Ring card indicators, which have the following meanings:

LED D (amber)	LED E (green)	Status
ON	ON	Adapter reset
OFF	ON	Diagnostics failed. Adapter check. Fatal error.
ON	OFF	Adapter open and operational
ON	OFF	No power or initialization in progress
FLASHING	OFF	Diagnostics OK. Awaiting OPEN command
ON	FLASHING	Beaconing hard error
OFF	FLASHING	Wire fault. OPEN failed
FLASHING	FLASHING	Awaiting initialization. Diagnostics not started yet.

FICON Express

A sketch of a FICON Express card is included in Figure B-1 on page 130. There are several status LEDs on the card that can be quite useful for diagnostic purposes. These are explained as follows:

LED A (Green)	LED B (Amber)	Status
OFF	OFF	No power to card, or card processor is looping
OFF	FLASHING	Power-on self tests running
FLASHING	OFF	Tests complete; CHPID online
FLASHING	ON	Hardware error detected
ON	FLASHING	Invalid indication

(Any combination that does not have one blinking light indicates an error or lack of power)

LED C (Green)	Status
OFF	CHPID is online and communication with a PU
OFF	CHPID is offline for maintenance or wrap test

(LED C rapidly flashing indicates that power-on tests are running)

LED D (Green)	LED E (Amber)	Status
OFF	OFF	Wake-up failure or cannot communicate
OFF	ON	Power-on test failure.
OFF	SLOW FLASH	Wake-up failure or cannot communicate
OFF	FAST FLASH	Power-on text failure
OFF	IRREGULAR FLASH	Power-on test in progress
ON	OFF	Failure while functioning
ON	ON	Failure while functioning
ON	SLOW FLASH	Normal - inactive
ON	IRREGULAR FLASH	Normal - active
ON	FASH FLASH	Normal - busy
SLOW FLASH	OFF	Normal - link is down or not started
SLOW FLASH	ON	Not defined
SLOW FLASH	SLOW FLASH	Offline for download
SLOW FLASH	FAST FLASH	Restricted offline mode (waiting for restart)
FAST FLASH	OFF	Debug monitor in restricted mode
FASH FLASH	ON	Not defined
FAST FLASH	SLOW FLASH	Debug monitor in test fisture mode
FASH FLASH	FAST FLASH	Debug monitor in remote debug mode
FAST FLASH	IRREGULAR FLASH	Debug monitor output active

Related publications

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

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 133.

- *IBM @server™ zSeries Connectivity Handbook,* SG24-5444
- *IBM @server™ zSeries 990 Technical Guide,* SG24-6947

Other resources

These publications are also relevant as further information sources:

- *z/OS ICSF Overview,* SA22-7519
- *z/OS ICSF System Programmer’s Guide,* SC23-3974
- *z/OS ICSF Application Programmer’s Guide,* SA22-7522
- *z/OS ICSF Administrator’s Guide*, SC23-3975
- *Support Element Operations Guide*, GC38-0608
- *PR/SM Planning Guide*, SB10-7036
- *zSeries 990 Installation Manual for Physical Planning*, GC28-6824

Referenced Web sites

These Web sites are also relevant as further information sources:

http://www-1.ibm.com/servers/eserver/zseries/connectivity

How to get IBM Redbooks

You can order hardcopy Redbooks, as well as view, download, or search for Redbooks at the following Web site:

ibm.com/redbooks

You can also download additional materials (code samples or diskette/CD-ROM images) from that site.

IBM Redbooks collections

Redbooks are also available on CD-ROMs. Click the CD-ROMs button on the Redbooks Web site for information about all the CD-ROMs offered, as well as updates and formats.

Index

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

IBM IBM @server zSeries 990 Technical Introduction

(0.5” spine)
0.475”<->0.873”
250 <-> 459 pages

IBM @server zSeries 990 Technical Introduction

Redbooks

System overview

Hardware description

Frequently asked questions

The IBM @server zSeries z990 provides major extensions to existing zSeries architecture and capabilities. The concepts of books and channel subsystems are added to the architecture, and the maximum number of LPARs is increased. These architectural extensions provide the base for much larger zSeries machines. This IBM Redbook provides an overview of these changes, and goes into more detail in selected areas.

By removing memory, processor, and channel constraints, z990 can be used for major server consolidation activities. More granular options for nondisruptive maintenance, growth, and alterations provide better 7 x 24 x 365 operation. In general, no changes are needed for existing applications—allowing immediate growth into z990.

Readers are assumed to have a general understanding of existing S/390 and zSeries hardware. Terms, acronyms, abbreviations, and concepts associated with existing systems are used without further introduction.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-6863-00 ISBN 0738427322