

Planning SuperDome Configurations

HP 9000 Computers

Edition 1

E1000

United States

© Copyright 1983-2000 Hewlett-Packard Company. All rights reserved..



Legal Notices

The information in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty

A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

HEWLETT-PACKARD COMPANY
3000 Hanover Street
Palo Alto, California 94304 U.S.A.

Use of this manual and flexible disk(s) or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs, in their present form or with alterations, is expressly prohibited.

Copyright Notices

Copyright © 1983-2000 Hewlett-Packard Company. All rights reserved. Reproduction, adaptation, or translation of this document without prior written permission is prohibited, except as allowed under the copyright laws.

Copyright © 1979, 1980, 1983, 1985-93 Regents of the University of California. This software is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California.

Copyright © 1988 Carnegie Mellon University
Copyright © 1990-1995 Cornell University
Copyright © 1986 Digital Equipment Corporation.
Copyright © 1997 Isogon Corporation
Copyright © 1985, 1986, 1988 Massachusetts Institute of Technology.
Copyright © 1991-1997 Mentat, Inc.
Copyright © 1996 Morning Star Technologies, Inc.
Copyright © 1990 Motorola, Inc.
Copyright © 1980, 1984, 1986 Novell, Inc.
Copyright © 1989-1993 The Open Software Foundation, Inc.
Copyright © 1996 Progressive Systems, Inc.
Copyright © 1989-1991 The University of Maryland
Copyright © 1986-1992 Sun Microsystems, Inc.

Trademark Notices

Apple® and Macintosh® are trademarks of Apple Computer, Inc., registered in the United States and other countries.

MS-DOS® and Microsoft® are U.S. registered trademarks of Microsoft Corporation.

OSF/Motif™ is a trademark of the Open Software Foundation, Inc. in the U.S. and other countries.

REFLECTION® and WRQ® are registered trademarks of WRQ, Inc.

UNIX® is a registered trademark in the United States and other countries, licensed exclusively through The Open Group.

VERITAS® is a registered trademark of VERITAS Software Corporation.

VERITAS File System™ is a trademark of VERITAS Software Corporation.

X Window System™ is a trademark of the Massachusetts Institute of Technology.

Publication History

- First Edition: October 2000 (HP-UX 11i); see <http://www.docs.hp.com/> for updates.

1. Planning SuperDome Configurations

How To Use this Document	8
Building Blocks and Definitions	10
Outline of this Section	10
Glossary	11
What is a SuperDome System?	15
What Is a Complex?	16
The Complex Profile	16
What Actions Change the Complex Profile?	18
When Are Complex Profile Changes Broadcast to Cells?	18
What is the Guardian Service Processor?	20
What is a Partition?	22
What is a CPU Cabinet?	23
What is a Cell?	27
Cell Components	28
Cell Controller (CC)	30
Processors	31
Memory	32
Cell Compatibility	32
Cell Connectivity	32
Cell I/O	32
Crossbar Link to CPU Cabinet	33
Cell Types	33
Active Cell	33
Base Cell	33
Core Cell	33
What Happens when a Cell Boots	35
What is an XBC (Crossbar Controller)?	36
Crossbar Connections	38
Cross-Flex and U-Turn	39
What is an I/O Chassis?	41
I/O Cards	41
Core I/O	42
Configuring and Controlling an I/O Chassis	43
What is an I/O Expansion Cabinet?	44
The Console and the Support Management System	45
Support Management Station (SMS)	46
Rules and Guidelines for Configuring a Complex	47

Contents

Recommendations for Cabling Crossbar Controllers (XBCs)	48
When Do You Need To Think about Cabling?	48
Guidelines for Performance	48
Choosing Cells for Partitions	49
Before you go on, read:	49
Other terms and concepts:	49
Points to Note	49
Building a Complex from Scratch	50
Guidelines for Performance and High Availability	51
When To Add Cells	51
Where To Add Cells	52
Distributing Resources.	55
Guidelines for Expandability	56
Partitions, Cells and I/O Chassis	57
Before you go on, read:	57
Other terms and concepts:	57
Points To Note	57
Loading and Assigning I/O Chassis	58
Rules	60
Guidelines for High Availability.	62
Guidelines for Performance	64
Guidelines for Expandability	64
Checklist for Performance	65
Checklist for High Availability.	66

1 **Planning SuperDome Configurations**

What follows is a white paper intended to help system administrators and system architects plan, configure and reconfigure the structural components of a SuperDome complex.

It contains the following sections:

1. “How To Use this Document” on page 8.
2. “Building Blocks and Definitions” on page 10.
3. “Rules and Guidelines for Configuring a Complex” on page 47.

How To Use this Document

Terms:

- **Cell:** see “What is a Cell?” on page 27.
 - **Unassigned cell:** see “Cell Types” on page 33.
- **Complex:** see “What Is a Complex?” on page 16.
- **I/O chassis:** see “What is an I/O Chassis?” on page 41.
- **Partition:** see “What is a Partition?” on page 22.

Full Glossary on page 11 .

The primary purpose of this paper is to help you plan how to reconfigure a SuperDome **complex**, though you may also find it useful for mapping out a SuperDome you intend to order.

The heart of the document is the section “Rules and Guidelines for Configuring a Complex” on page 47, but if you are unfamiliar with the SuperDome architecture, start with the “Building Blocks and Definitions” on page 10.

There are two ways to reconfigure a SuperDome:

- Physically move components such as **cells** and **I/O chassis**; *or*
- Logically reassign cells from one **partition** to another.

As a customer, you must not make changes of the first type yourself: a Hewlett-Packard Customer Engineer or Service Engineer must be the person to move major components such as cells, I/O chassis and cabinets; that is, anything larger than an I/O card. Use this paper to help you plan such changes, but don't attempt to carry them out yourself.

But you can make changes of the second type, reassigning cells from one partition to another, or assigning **unassigned** cells. Such changes can affect performance and high-availability substantially, so it is important that you read the “Rules and Guidelines for Configuring a Complex” on page 47 carefully before proceeding. Then use chapter 4 of the manual *Managing SuperDome Complexes* to guide you through the necessary tasks.

You may be reading this paper in the form of an appendix to *Managing SuperDome Complexes*, or as PDF file delivered in the `/usr/share/doc` directory of an HP-UX 11i system. The most recent version is published on Hewlett-Packard's documentation website, `docs.hp.com`. To check that version for changes, go to `docs.hp.com`, then choose "Browse by Topic", then "HP-UX 11i Operating System" and then "White Papers" under "System Administration".

Building Blocks and Definitions

Outline of this Section

- “Glossary” on page 11
- “What is a SuperDome System?” on page 15
- “What Is a Complex?” on page 16
- “What is the Guardian Service Processor?” on page 20
- “What is a Partition?” on page 22
- “What is a CPU Cabinet?” on page 23
- “What is a Cell?” on page 27
- “What is an XBC (Crossbar Controller)?” on page 36
- “Crossbar Connections” on page 38
- “What is an I/O Chassis?” on page 41
- “What is an I/O Expansion Cabinet?” on page 44
- “The Console and the Support Management System” on page 45

Glossary

- **16-, 32-, 64-way-capable system:**

The three SuperDome models currently available; see “What is a CPU Cabinet?” on page 23.

- **CPU cabinet:**

SuperDome’s hardware “box”; see “What is a CPU Cabinet?” on page 23.

- **Cell; cell board:**

SuperDome’s hardware building blocks, containing memory, processors and other core components; see “What is a Cell?” on page 27.

- active cell:**

a cell in use in a partition; see “Active Cell” on page 33.

- base cell:**

a cell assigned to a partition; see “Base Cell” on page 33.

- core (root) cell:**

the cell that supports the console for its partition, and performs other key functions; see “Core Cell” on page 33.

- inactive cell:** a cell assigned to a partition which is not available to the operating system running on that partition; see “Active Cell” on page 33.

- unassigned (free) cell:** a cell that has not been assigned to any partition.

- viable core cell:** a base cell that is attached to an **I/O chassis** that contains a **core I/O** card. See “Core Cell” on page 33.

- **Cell Controller (CC):**

The chip on each **cell** that is responsible for maintaining data coherency across all the cells in a partition, connecting the cell to I/O (via the **System Bus Adapter**) and to the **crossbar controller (XBC)**. See “Cell Components” on page 28.

- **Cell I/O:**

Refers to the connection between **cells** and **I/O chassis**, which contain I/O cards; see “Cell I/O” on page 32.

- **Complex:**
A hardware configuration that can support multiple instances of an operating system (by means of **partitions**); see “What Is a Complex?” on page 16.
- **Complex Profile:**
The data structure managed by the GSP that represents the configuration of a complex. See “The Complex Profile” on page 16.
- **Core I/O:**
Comprises console support and 10/100 Base T LAN; see “Core I/O” on page 42.
- **Cross-Flex:**
The cabling used to connect the **crossbars** in a **64-way-capable** system; see “Crossbar Connections” on page 38.
- **Crossbar; crossbar controller (XBC):**
The backplane board that **cells** plug into, and its controlling chips; see “What is an XBC (Crossbar Controller)?” on page 36.
- **DIMM:**
Dual Inline Memory Module; see “Memory” on page 32.
- **Guardian Service Processor:**
The board that maintains configuration information about the **complex** and oversees configuration changes; see “What is the Guardian Service Processor?” on page 20.
- **iCOD:**
Instant Capacity on Demand. Allows you to activate inactive processors. See “Instant Capacity on Demand (iCOD)” on page 31.
- **I/O chassis (cardcage):**
A cardcage containing 12 I/O slots; see “What is an I/O Chassis?” on page 41
- **I/O expansion cabinet:**
An external cabinet containing up to six 12-slot **I/O chassis**; see “What is an I/O Expansion Cabinet?” on page 44.

- **I/O slots:**
The slots in an **I/O chassis**; see “I/O Cards” on page 41.
- **IPL:**
Initial Program Load(er). See “What Happens when a Cell Boots” on page 35.
- **Local Bus Adapter:**
The chip that connects an individual I/O slot to the **System Bus Adapter**, and thence to a **cell**; see “Cell Connectivity” on page 32.
- **Monarch CPU:**
The processor that performs selftest and other functions when a cell is activated as part of a booting partition. See “What Happens when a Cell Boots” on page 35. Often used to refer specifically to the **core cell’s** monarch CPU.
- **PCI cards:**
Peripheral Interface Cards, often called simply “I/O cards”; see “I/O Cards” on page 41.
- **PDC:**
Processor Dependent Code. See “What Happens when a Cell Boots” on page 35.
- **Partition:**
A **cell**, or usually group of cells, running an instance of an operating system in a **Complex**; see “What is a Partition?” on page 22.
- **Partition Manager (parmgr):**
A menu-driven software tool (`/opt/parmgr/bin/parmgr`) that runs under HP-UX and is used to configure and reconfigure **complexes**. Can be executed locally on a **partition**, or remotely from a PC via a web browser.
- **Quad:**
A group of four cell slots that are connected to a single **crossbar controller (XBC)**. See “What is an XBC (Crossbar Controller)?” on page 36.

- **SAM:**
The menu-driven System Administration Manager tool used to configure HP-UX.
- **Single Computer Board (SBC) and SBC Hub (SBCH):**
The components of the **Guardian Service Processor (GSP)**. See “What is the Guardian Service Processor?” on page 20. There is one SBC per **complex**, and one SBCH per **CPU cabinet**.
- **Support Management Station (SMS); scan station:**
A workstation connected to SuperDome **complexes** to run diagnostics; see “Support Management Station (SMS)” on page 46.
- **System Bus Adapter:**
The chip that connects a **cell** to an **I/O chassis**; see “Cell Connectivity” on page 32.
- **XBC:**
See **crossbar controller**, and “What is an XBC (Crossbar Controller)?” on page 36.

What is a SuperDome System?

Terms:

- **Cell:** see “What is a Cell?” on page 27.
- **Complex:** see “What Is a Complex?” on page 16.
- **CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **Partition:** see “What is a Partition?” on page 22.

Full Glossary on page 11 .

SuperDome is a high-end server that can be (though it does not have to be) partitioned into several systems-within-a-system, each running its own operating-system “image” or instance. These systems-within-a-system are called **partitions**; the system that includes all the partitions is called a **complex**.

The primary hardware building-blocks are **cells**, which are housed in **CPU cabinets**; when configuring a system, you combine cells to create a partition.

At first release a SuperDome system comprises either one or two cabinets, each holding a maximum of eight cells (and a minimum of one).

What Is a Complex?

Terms:

- **Active, inactive, unassigned cell:** see “Active Cell” on page 33.
- **Cell:** see “What is a Cell?” on page 27.
- **Core cell:** see “Core Cell” on page 33.
- **CPU cabinet; 32-way-capable system; 64-way-capable system:** see “What is a CPU Cabinet?” on page 23.
- **Guardian Service Processor (GSP):** see “What is the Guardian Service Processor?” on page 20.
- **iCOD:** see “Instant Capacity on Demand (iCOD)” on page 31.
- **Partition:** see “What is a Partition?” on page 22.
- **XBC:** see “What is an XBC (Crossbar Controller)?” on page 36.

Full Glossary on page 11 .

A complex can be defined as a single hardware configuration that can support more than one instance of an operating system (by means of **partitions**).

- In hardware terms, a complex is the sum of all the hardware resources in, and attached to, one or more **CPU cabinets** that are cabled together.
- In software terms, it is the sum of all the partitions.

A maximum of two CPU cabinets (**32-way-capable systems**) can be combined into a complex. This is done by cabling together the crossbars (**XBCs**) from the two adjacent cabinets; this configuration is called a **64-way-capable system**.

The Complex Profile

Information about the complex is stored in the **complex profile**, which is written by the commands used to configure the complex (and indirectly by the **Partition Manager (parmgr)**, which invokes those commands) and maintained by the **Guardian Service Processor (GSP)**.

The complex profile is used by **PDC (Processor Dependent Code)**, the HP-UX kernel and the **Partition Manager (parmgr)**, as well as by the GSP. It consists of three parts:

- **Stable complex configuration information (Group A)**, including:
 - Attributes of the complex (its name, model number, serial number, etc)
 - cell-to-partition assignments and unassigned cells**
 - XBC connections**
 - other complex-wide information
- **Dynamic complex configuration information (Group B)**.
Complex profile revision information, used to make sure that all cells and partitions have the same information about the configuration of the complex.
- **Partition configuration information (Group C)**, including:
 - Partition name
 - Partition number
 - IP address
 - Primary boot path
 - Alternate boot path and High Availability alternate boot path (see “Paths to boot and root disk” on page 63)
 - Boot timer
 - KGM (Known Good Memory)
 - Autostart and other flags
 - Console and keyboard paths
 - Core cell** selection table
 - “Use on boot” flag for each cell, indicating whether or not each of the partition’s cells will be activated next time the partition boots. The default is to activate the cell.
 - “Failure policy” for each cell, indicating whether or not the cell is to be activated as part of the partition if selftest reveals a processor or memory-module failure, but the cell is otherwise viable. The default is to activate the cell.

- ❑ **iCOD** information indicating how many processors in the partition, if any, you have not purchased; see “Instant Capacity on Demand (iCOD)” on page 31.

What Actions Change the Complex Profile?

Table 1-1

Action	Affected Cells	Affected Part(s) of Profile
Create a partition	All cells in the partition	Groups A and C
Delete a partition	All cells in the partition	Groups A and C
Add cells to a partition	Cells being added	Group A
Remove inactive cells from a partition	Cells being removed	Group A
Remove active cells from a partition	All cells in the partition (requires rebooting the partition)	Group A

When Are Complex Profile Changes Broadcast to Cells? A copy of the complex profile is stored in each cell’s memory. The GSP updates the cells’ copies of the profile under the following circumstances:

- If the stable complex configuration information (Group A) has changed:
 - ❑ If no cell’s partition assignment has changed, the changes are broadcast immediately.
 - ❑ If only inactive cells’ assignments have been changed, the changes are broadcast immediately.
 - ❑ If an active cell’s partition assignment has changed, the changes are broadcast once the cell is inactive (that is, no cells will receive the new information until the affected cell is inactive).

- If the partition configuration information (Group C) has changed:
 - ❑ The changes are broadcast to the cells in that partition immediately, though the information will normally not be used until the next time the partition boots.

Point to note: The GSP will not update stable complex configuration information (Group A) for *any* cell unless it can update *all* cells, and it cannot update the partition assignment of an active cell.

What is the Guardian Service Processor?

Terms:

- **Complex:** see “What Is a Complex?” on page 16.
- **CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **Partition:** see “What is a Partition?” on page 22.
- **Support Management Station (SMS):** see “Support Management Station (SMS)” on page 46.

Full Glossary on page 11 .

One **CPU cabinet** in each **complex** contains the **Guardian Service Processor (GSP)**, comprising a **Single Computer Board (SBC)** and an **SBC hub (SBCH)**. The GSP maintains configuration information about the complex and oversees configuration changes; see “The Complex Profile” on page 16.

Specifically, the GSP is needed for:

- Access to the complex console.
- Getting information about the configuration of a **partition** other than the one you are on, or the complex.
- Changing partition or routing configuration.
- Propagating diagnostic chassis logs.
- Notification of events in partitions other than the one you are on.

More broadly, the GSP and its menu interface comprise a command center for managing the complex, providing the following capabilities:

- Always-on capability.
The GSP is alive so long as the circuit breakers are closed.
- Access control.
Provides three levels of capabilities.
- Multiple access methods:
 - Local RS232 port, providing support for directly connected terminal or laptop computer.
 - Remote-modem port.
 - Customer LAN port, providing support for telnet access.
 - HP private LAN, providing support for telnet access, diagnostics, and firmware updates from the **Support Management Station**.See “The Console and the Support Management System” on page 45 for a diagram.
- Multi-user access to the complex.
Multiple users can log in through the LAN port simultaneously and manage different partitions or examine the state of the complex.
- Multi-user access to a partition’s console.
Multiple users can telnet simultaneously to a given partition’s console.
- Virtual Front Panel (VPF).
Displays the boot/run state for a given partition.
- Power on/off, system reset and TOC (Transfer of Control) capabilities.

For information on using the GSP to perform specific system-management tasks, see chapters 2, 4 and 5 of the manual *Managing SuperDome Complexes* (you may be reading this paper in the form of an appendix to that manual).

What is a Partition?

Terms:

- **Cell:** see “What is a Cell?” on page 27.
- **Complex:** see “What Is a Complex?” on page 16.
- **CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **I/O chassis:** see “What is an I/O Chassis?” on page 41.

Full Glossary on page 11 .

A partition corresponds roughly to a single, standalone system. Each **complex** can be subdivided into several partitions, each containing one or (usually) more **cells** and running a single instance of the operating system (at first release, HP-UX 11i only).

Partitioning the resources of the complex in this way makes it easy to run multiple applications on the same physical system; you can allocate physical resources and tune the operating system running on each partition depending on the needs of the application (or the most important application) you intend to run on it.

Alternatively, you can configure the complex as a single partition, allowing all the resources to be focussed on a single set of tasks, for example a large online transaction-processing application.

At first release, the maximum number of cells in a single partition is sixteen (two **CPU cabinets** containing a maximum of eight cells each); the minimum is one. Similarly, the maximum number of partitions in a complex at first release is sixteen (sixteen single-cell partitions); the minimum is one (a single partition containing any number or cells from one to sixteen).

Each partition must contain at least one cell that is attached to an **IO chassis**; see “Partitions, Cells and I/O Chassis” on page 57 for more information.

You can increase or reduce the processing power of a partition by adding or deleting cells (at first release, you must shut down the operating system running on the affected partition(s) before moving cells, and before configuration changes will take effect). Though HP-UX 11i does include commands for some configuration tasks, HP recommends you use the Partition Manager (parmgr) to configure partitions.

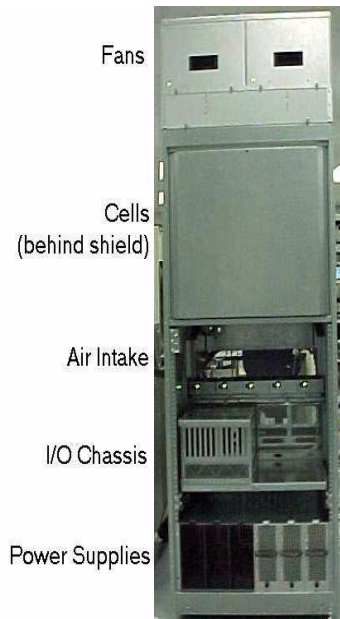
There are performance and high-availability issues you need to consider before you install a new cell or move an existing one or re-assign it to another partition; see “Choosing Cells for Partitions” on page 49.

What is a CPU Cabinet?

Terms:

- **Cell:** see “What is a Cell?” on page 27
- **Complex:** see “What Is a Complex?” on page 16
- **I/O chassis:** see “What is an I/O Chassis?” on page 41
- **Guardian Service Processor:** see “What is the Guardian Service Processor?” on page 20.

Full Glossary on page 11

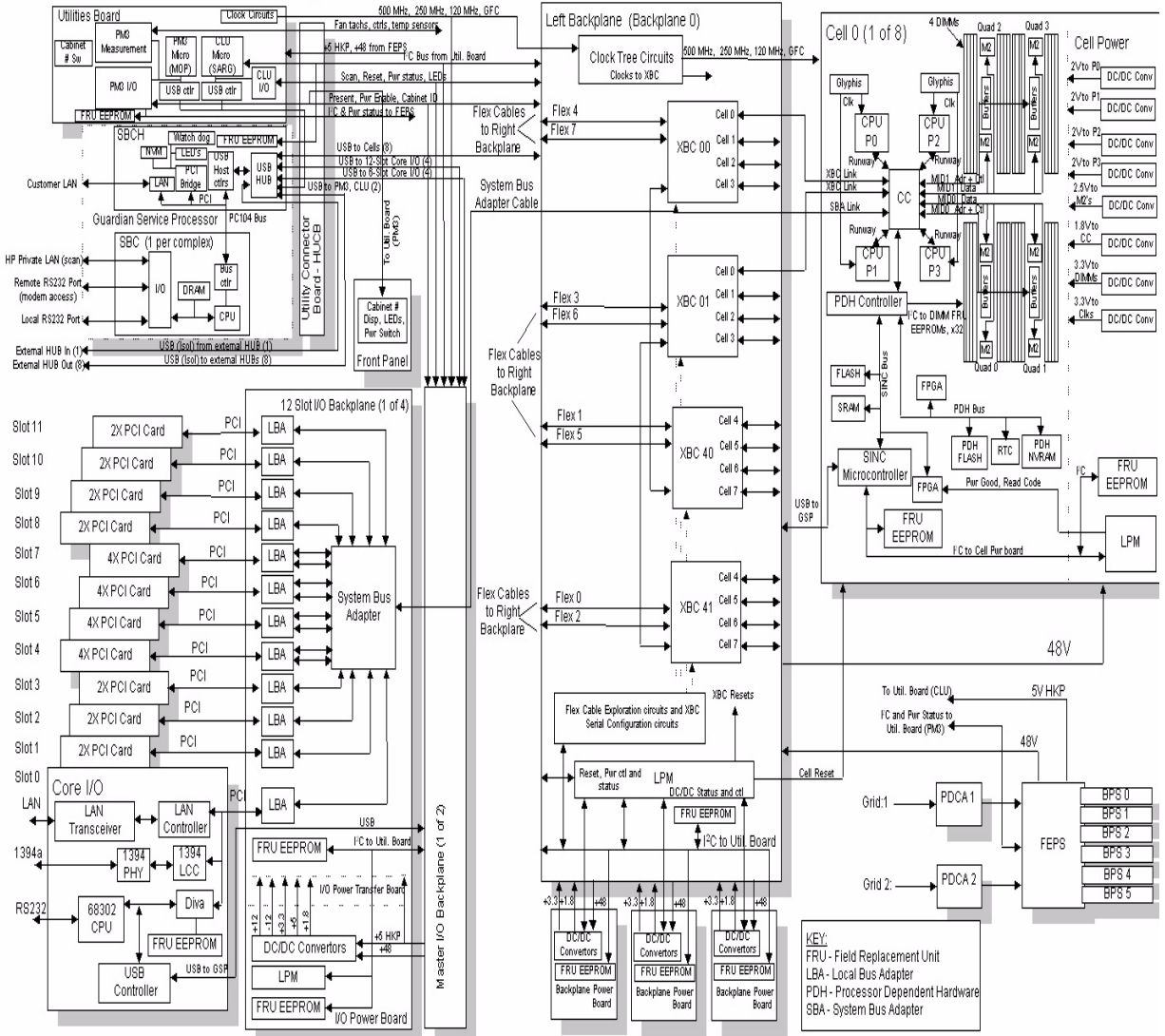


A CPU cabinet is SuperDome's hardware “box.” It contains:

- **cells;**
- The **Guardian Service Processor** (one per **complex**);
- **I/O chassis** (a maximum of four chassis for a maximum of 48 slots);
- five top-venting I/O fans;
- four top-venting cabinet fans;
- six power supplies, connected to the power source by one or (for redundancy) two cables.

This block diagram shows how all the components in a 32-way-capable system communicate.

SuperDome Cabinet showing 32 way capable system



NOTE

A CPU cabinet has no room for internal disks; all peripherals are external, attached to the I/O chassis.

Cabinets can be cabled together to form a **complex**. A complex comprises a maximum of two cabinets. A single cabinet is a **16-way-capable** (SD1600) or **32-way-capable** (SD3200) system; two contiguous cabinets cabled together are a **64-way-capable** system (SD6400).

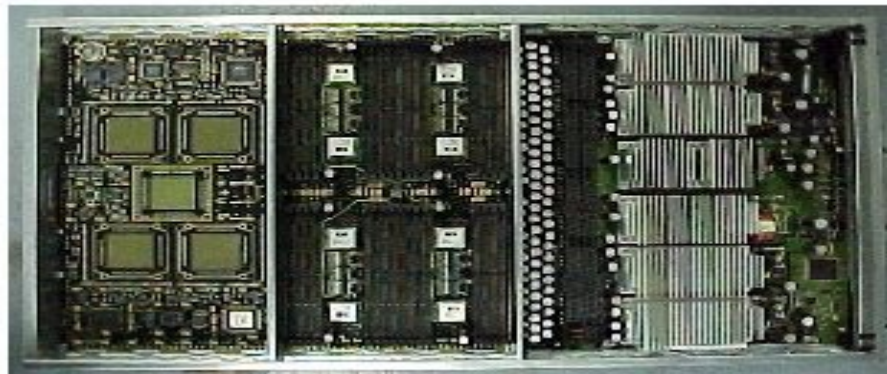


What is a Cell?

Terms:

- **CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **Cell Controller (CC):** see “Cell Controller (CC)” on page 30.
- **Complex:** see “What Is a Complex?” on page 16.
- **Complex profile:** see “The Complex Profile” on page 16.
- **Core I/O:** see “Core I/O” on page 42.
- **GSP:** see “What is the Guardian Service Processor?” on page 20.
- **I/O chassis:** see “What is an I/O Chassis?” on page 41.
- **I/O Expansion cabinet:** see “What is an I/O Expansion Cabinet?” on page 44.
- **Partition:** see “What is a Partition?” on page 22.
- **SuperDome system:** see “What is a SuperDome System?” on page 15.
- **System Bus Adapter:** see “Cell I/O” on page 32.
- **XBC link:** see “What is an XBC (Crossbar Controller)?” on page 36.

Full Glossary on page 11 .



*In the picture above, the rightmost section of the board is power bricks and capacitors; the center section is space for DIMMs, and the left section contains sockets for four processors surrounding a socket for the **cell controller (CC)**.*

A cell, or cell board, is the basic building block of a **SuperDome system**. When configuring or reconfiguring a **complex**, you assign cells to **partitions**. A cell provides processing power comparable to that of a mid-range server, but a SuperDome system supports this processing power with much greater memory capacity and I/O bandwidth. For more information, see:

- “Cell Components” on page 28.
- “Cell Compatibility” on page 32.
- “Cell Connectivity” on page 32.
- “Cell Types” on page 33.
- “What Happens when a Cell Boots” on page 35.

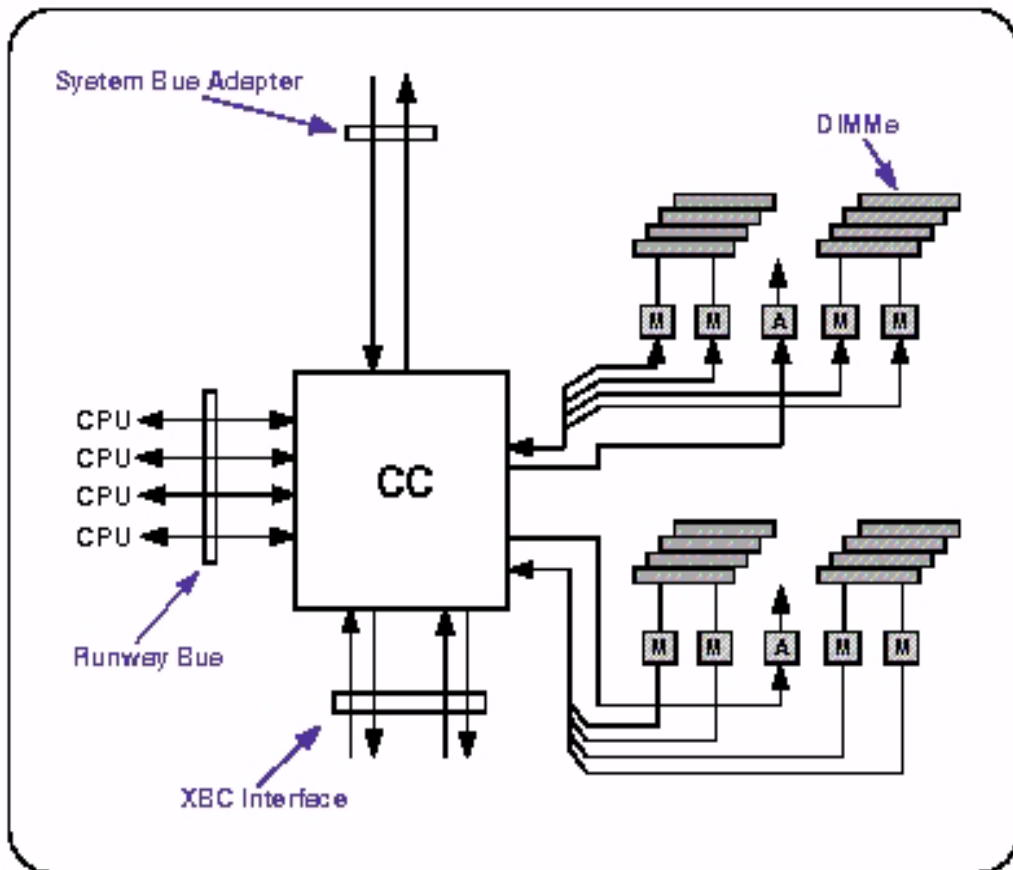
Cell Components

A cell is a single logic board containing:

- power converters;
- data busses;
- processors; see “Processors” on page 31;
- memory; see “Memory” on page 32;
- one **cell controller (CC)**; see “Cell Controller (CC)” on page 30.

This diagram shows how these components are connected.

Cell Block Diagram



This cell contains four double speed busses, each connecting a single processor to the **cell controller (CC)** chip, which controls the cell's memory and links the cell to other cells and to I/O.

Cell Controller (CC) The cell controller chip co-ordinates memory between the major components of a cell board -

- memory controllers (DIMMs),
- processors,
- the I/O bus (**System Bus Adapter**)

- and determines if a request requires communication with another cell or with the I/O subsystem.

The CC has two interfaces which leave the cell:

- the **crossbar interface (XBC link)**, through which all communication to other cells flows;
- the **System Bus Adapter**, which connects the cell to an I/O subsystem.

See “Cell Components” on page 28 for a block diagram.

Processors Each cell contains four processors. All the processors in the same **partition** must be of the same type (the partition will not boot if they are not). At first release, the only processor type available is PA-8600.

Instant Capacity on Demand (iCOD) All cells are delivered with four processors, but you pay only for the processors you use. For example, a cell ordered with two active processors will be delivered with two active and two inactive processors. If and when you need additional capacity, you can use the Partition Manager (parmgr) to turn on one or both of the inactive processors, and HP will bill you accordingly.

NOTE

iCOD is a separately orderable product which is not available with the earliest SuperDome shipments; those systems use a different method for turning on inactive processors. Contact your HP Sales Representative for more information.

Order for Activating Processors. The algorithm the HP tools use to activate new processors is:

1. There must be at least one active CPU per cell.
2. CPUs are activated round-robin across cells within a partition (that is, first processor 0 in each cell, then processor 3 in each cell, and so on following rule 3). This means that the number of active processors per cell will differ by at most one across the partition.
3. CPUs are enabled on a cell board in the order 0, 3, 1, 2. This spreads the CPUs across the two internal Cell Controller busses and allocates CPUs in the best thermal fashion.
4. When a failed CPU is replaced, choose one from the same cell when possible. If it is not possible, choose the next available CPU following rules 2 and 3.

Performance and High Availability For best performance, all cells in the same partition should contain the same number of active processors. For high-availability reasons, each cell should contain at least two active processors. See also “Choosing Cells for Partitions” on page 49.

Memory Each cell can contain up to 16 GB RAM in 2 GB increments.

Performance and High Availability For high availability reasons, a cell should contain at least 8 memory DIMMs (Dual Inline Memory Modules) for a minimum of 4 GB RAM. For the best performance, all the cells in a partition should contain the same amount of RAM. This is because all the memory in a partition is **fully cache-line-interleaved**, meaning that each cell's memory forms part of a common pool used by processes running on that partition. If one cell in the partition has more memory than the others, that cell will contribute disproportionately to the pool and the connections to and from it will be overworked, degrading performance.

See also “Choosing Cells for Partitions” on page 49.

Cell Compatibility

When a partition boots, PDC selects a **core cell** (see “Core Cell” on page 33). Other cells in the partition are allowed to boot if they match the core cell in the following respects:

- The firmware revisions have the same major number.
- The cache sizes of the processors are the same, and the processors are of the same type.

The Partition Manager (parmgr) will warn you if you try to configure an incompatible cell into a partition.

Cell Connectivity

Cells communicate with each other (within a **partition**); and with I/O devices by means of an I/O chassis (or, if the cell in question is not connected to an I/O chassis, via a cell that is connected to I/O).

Cell I/O At least one cell in a partition must be connected to an I/O chassis, and preferably more than one (see “Partitions, Cells and I/O Chassis” on page 57).

A cell can be connected to no more than one **I/O chassis**, and an I/O chassis can be connected to no more than one cell. Unless you add an **I/O expansion cabinet**, this means that a maximum of four cells (out of a possible eight in a CPU cabinet) can be directly connected to I/O. Adding an I/O expansion cabinet would theoretically allow all eight cells in a full CPU cabinet to be connected to I/O. Cells that have I/O are connected to an I/O chassis via the **System Bus Adapter**, which in turn is connected to the cards in the individual I/O slots, via **Local Bus Adapter** chips. See “I/O Cards” on page 41 for more information.

Crossbar Link to CPU Cabinet Cells are connected to the cabinet by means of the cell controller's **crossbar (XBC) link**. A maximum of four cells plug into a crossbar; there are two crossbar in a CPU cabinet. See “What is an XBC (Crossbar Controller)?” on page 36 for more information.

Cell Types

Active Cell An active cell is a cell that has been assigned to a **partition**, has power enabled, and is integrated into the operating system running on the partition.

An **inactive cell** is a cell that has been assigned to a partition, but was not activated when the partition booted (either because you de-activated it by means of its “use on next boot” flag in the **complex profile** or because of a failure). An **unassigned cell** does not currently belong to any partition.

Base Cell A base cell is a cell that has been assigned to a partition.

Core Cell A core cell (also known as the **root cell**) is the cell that the Processor-Dependent Code (PDC) has selected at boot time to perform the following set of roles and functions for the partition:

- Support the console
- Support the system clock
- Maintain copies of the GSP chassis logs
- Via the **monarch** CPU, perform the system boot and execute **IPL (Initial Program Load)** for the partition.
- Keep the “master copy” for some data structures
- Serve as the criterion for the compatibility of the other cells in the partition; see “Cell Compatibility” on page 32.

To qualify as a **viable core cell**, a cell must be a **base cell** and must be attached to an **I/O chassis** that contains a **core I/O** card. A partition's core cell, the one from which it boots, should normally be its lowest-numbered cell, but partitions can (and, for high-availability reasons, should) have more than one viable core cell.

NOTE

In a booted partition that has more than one viable core cell, the only core I/O card that is active is the core cell's.

How the Core Cell is Selected. If a partition has more than one viable core cell, PDC (Processor Dependent Code) decides when the partition is booting which cell should be the core cell. It does this on the basis of a prioritized list which is part of the Partition Configuration Data maintained by the GSP. If PDC detects any problem with the first cell on the list or its I/O, PDC will choose the second cell on the list; if both of these are faulty and there is a third, it will choose the third, and so on. If all the cells in the list are faulty (or there are no cells in the list) PDC will check all the cells in the partition that have core I/O (starting with the lowest- numbered cell) until it finds a viable core cell or it has exhausted all the possibilities.

You can use Partition Manager (parmgr) to modify this list, and so designate which should be the primary, secondary or later choice for the core cell; but normally you should accept Partition Manager's defaults unless you have a particular reason not to.

See also "Partitions, Cells and I/O Chassis" on page 57.

What Happens when a Cell Boots

This section explains how a cell becomes active as its partition boots. The sequence is as follows:

1. The system administrator enables power to the cell.
(This can be done by means of the HP-UX `frupower` command, or from SAM or the **GSP**.)
2. The cell is held in the **reset** state until power stabilizes.
3. The cell is released from reset and **boot is blocked** (that is, the cell is on but is not allowed to boot).
4. The cell's **monarch processor's PDC** (Processor-Dependent Code) sets a flag that signals the GSP that it can post a new **complex profile** (the complex profile is replicated in every cell's memory; see "The Complex Profile" on page 16).
5. The cell's monarch processor's PDC performs selftests, identifies the cell's hardware, checks for I/O (is the cell connected to an I/O chassis, and if so what devices are connected?) and identifies everything the cell is connected to via its crossbar link (**XBC**).
6. The cell reports its hardware configuration to the GSP.
7. The GSP ensures that the cell's copy of the complex profile has current information, reflecting both the cell's partition and the complex as a whole.
8. The GSP clears the "boot is blocked" flag, allowing the cell to boot.
9. The cell's monarch processor's PDC checks that the cell is compatible with the other cells in the partition (see "Cell Compatibility" on page 32).
10. If the cell is compatible, its monarch processor's PDC reads the complex profile and the cell boots as part of its partition.

If this is the **core cell**, its monarch CPU performs the system boot and executes IPL (Initial Program Load) for the partition.

What is an XBC (Crossbar Controller)?

Terms:

- **64-way-capable system; CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **Cell:** see “What is a Cell?” on page 27.
- **Cell Controller:** see “Cell Controller (CC)” on page 30.
- **Partition:** see “What is a Partition?” on page 22.

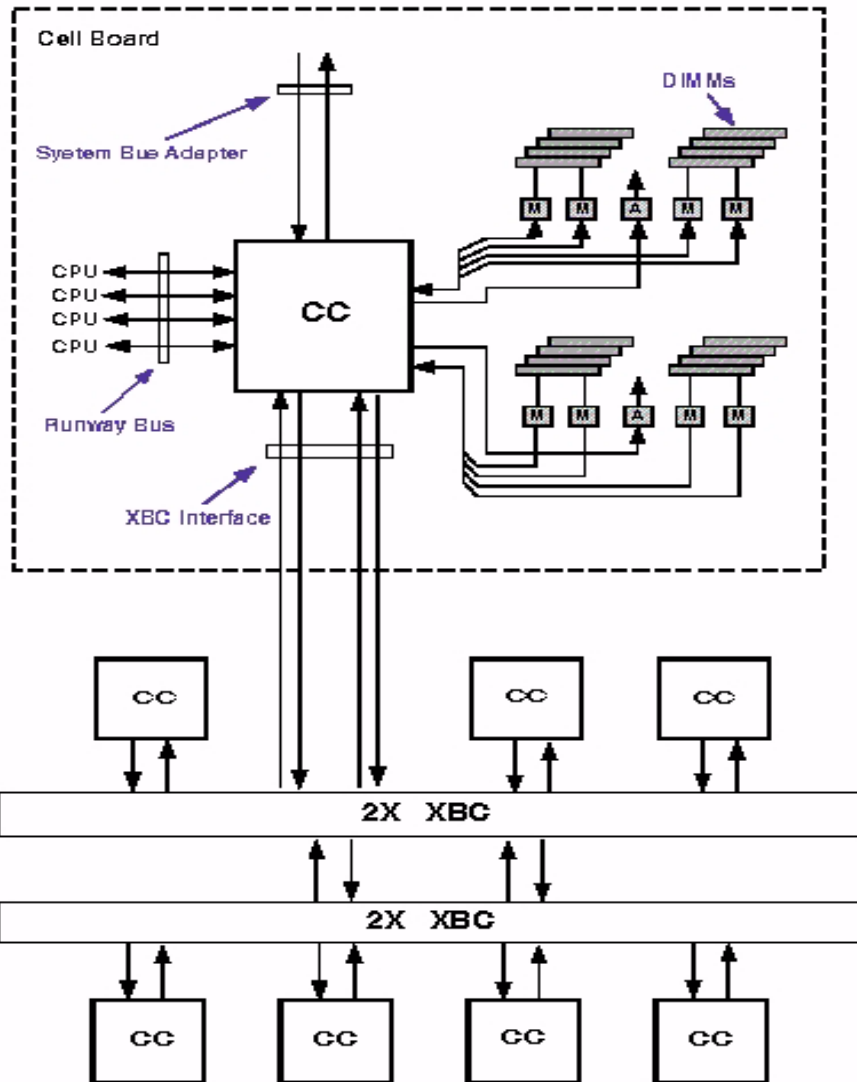
Full Glossary on page 11 .

Each **CPU cabinet** contains two crossbar backplane boards. Each **cell** plugs into one of these crossbars by means of a pair of connectors, forming a connection between the **cell controller (CC)** and the crossbar controller. The crossbar controller comprises two chips and is also known as the **XBC**.

There are two crossbars (and thus two XBCs) in a CPU cabinet, and a maximum of four cells plug into each, allowing a maximum of eight cells in a cabinet.

This diagram shows how cells connect to the backplane.

Cell Board/Backplane Connection



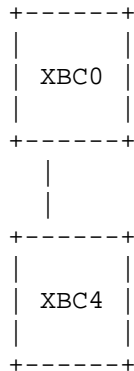
The two XBCs in a cabinet are connected to each other and, in a **64-way-capable system**, to the two XBCs in the other cabinet as well. These connections allow cells to communicate and to share memory both within and across crossbars, allowing **partitions** comprising more than four cells.

You need to be careful when configuring partitions that cross crossbar boundaries; not all configurations which are physically possible are supported. See “Choosing Cells for Partitions” on page 49.

Crossbar Connections

Three of the ports on a crossbar controller (**XBC**) are reserved for connections to other XBCs on other crossbars in the **complex**. One of these ports is used to connect to the other XBC in the same cabinet. This is a direct crossbar-to-crossbar connection (set in etch on the board) and is not configurable:

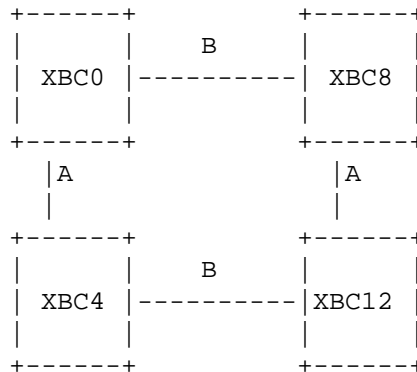
Direct Link



The direct XBC-XBC link between crossbars in a cabinet.

In a 64-way-capable-system, another of these three ports connects the crossbar to the corresponding XBC in the other cabinet:

Direct and Flex Links (64-way-capable-system)



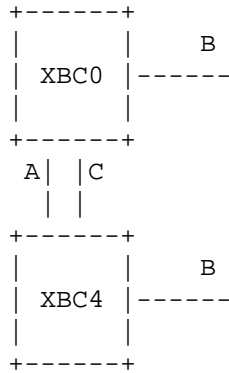
This figure shows the XBC-XBC links between crossbars (A) and the flex links between corresponding XBCs in two cabinets (B).

This leaves one more port for an XBC connection. The way this last port is connected defines the difference between a **U-Turn** and a **Cross-Flex** connection.

Cross-Flex and U-Turn

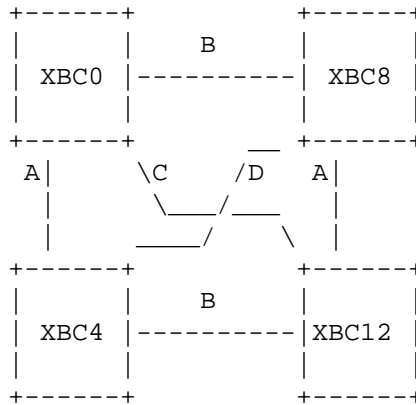
- If the last port is connected to the other XBC in the *same* cabinet (that is, a second XBC-XBC connection to the second XBC in the same cabinet), this is a **U-Turn** connection.
- If the connection is to the other XBC in another cabinet (such that XBC 0 is cabled to XBC 12), this is a **Cross-Flex** connection.

U-Turn (32-way-capable-system)



*In this figure, C shows a second XBC-XBC link (a flex link) between XBCs in the same cabinet. This is a U-Turn configuration for a **32-way-capable-system**. Note that the ports labeled B are not connected to anything.*

Cross-Flex (64-way-capable-system)



*This figure shows XBC-XBC links (C and D) connecting XBC 0 to XBC 12, and XBC 8 to XBC 12, in the other cabinet. This is a Cross-Flex configuration for a **64-way-capable-system**.*

See “Recommendations for Cabling Crossbar Controllers (XBCs)” on page 48 for guidelines on when to use which type of configuration.

What is an I/O Chassis?

Terms:

- **CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **Cell:** see “What is a Cell?” on page 27.
- **Partition:** see “What is a Partition?” on page 22.
- **GSP (Guardian Service Processor):** see “What is the Guardian Service Processor?” on page 20.

Full Glossary on page 11 .

An **I/O chassis** enables a **cell**, and hence a **partition** to communicate with I/O devices such as the system console, disk drives and the network. It contains slots for I/O cards and is sometimes referred to as a **cardcage**.

At first release, each **CPU cabinet** holds a maximum of four I/O chassis, two in each of the cabinet's two **I/O bays** (front and rear). I/O bays are the apertures in the CPU cabinet that the I/O chassis fit into. Partition Manager and other utilities report the location of an IO chassis in the form:

```
cabinet_#, bay_#, chassis_#
```

At first release, only 12-slot chassis are available.

See “Partitions, Cells and I/O Chassis” on page 57 for more information.

I/O Cards

At first release, each I/O chassis contains 12 **PCI** (Peripheral Card Interface) slots, numbered 0-11, right to left when looking at the chassis from the front.

- Slots 0, 1, 2, 3, 8, 9, 10, and 11 are 2X (33MHZ) 5-volt-only slots.
- Slots 4, 5, 6, and 7 are 4X (66MHZ) 3.3-volt-only slots.
- **Universal** PCI cards (cards that support both 5 and 3.3 volts) can plug into any slot, but keep in mind that:
 - ❑ Universal 2X PCI cards plugged into 4X slots will run only at 2X speed (33MHZ).
 - ❑ Universal 4X PCI cards plugged into 2X slots will also run only at 2X speed.

Core I/O Each partition must contain at least one cell that is attached to an I/O chassis containing **core I/O**, which comprises primarily:

- console support
- 10/100 BaseT LAN

Placement of Core I/O, Boot and Removable Media Cards:

- Put the **core-I/O card** in the rightmost slot (**slot 0**) of an I/O chassis (this is the only slot it can go in).
- Put the **boot device controller** in the same I/O chassis as the core I/O card. If the boot device controller is a 4X card, put it in **slot 4** of this chassis. If it is a 2X card, put it in **slot 1** of this chassis.
- Put the **removable media card** (e.g., for a DVD drive) in **slot 8** of this chassis.

Configuring and Controlling an I/O Chassis

Before a chassis is installed in its I/O bay and cabled to a cell, you will not be able to power it on. Once the chassis is cabled to a cell, powering on the cell will power on the chassis and powering off the cell will power off the chassis. Powering off a chassis that is attached to an active cell requires some planning.

Powering Off an I/O Chassis

You can power off an I/O chassis only if it is **inactive**. If the chassis is cabled to a cell, and the cell is active in a partition (that is, the partition has booted and the cell has booted as part of the partition) then the chassis is **active**.

This means that in order to power off a chassis (to do maintenance or replace it, for example) you need to shut down the partition and do one of the following:

- While the partition is down, use the **GSP** command menu to power off the chassis (a good approach if you need to replace the chassis and a spare is on hand - once the spare has been installed you can boot the partition).
- Disconnect the chassis from the cell, boot the partition, then go to the GSP command menu to power off the chassis (appropriate if the partition has to be back online as fast as possible and whatever needs to be done to the chassis might take some time).
- While the partition is down, from another partition set the “use on next boot” flag off for the cell attached to the chassis, then boot the partition; then you can power off the cell and chassis because the cell is not active.

What is an I/O Expansion Cabinet?

Terms:

- **CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **I/O chassis:** see “What is an I/O Chassis?” on page 41.

Full Glossary on page 11 .

NOTE

I/O expansion cabinets are not available with early shipments of SuperDome. Contact your HP Sales Representative for up-to-date information.

An I/O expansion cabinet contains up to six twelve-slot I/O chassis (cardcages). Unlike a **CPU cabinet**, it contains no slots for cells.

You need an I/O expansion cabinet if:

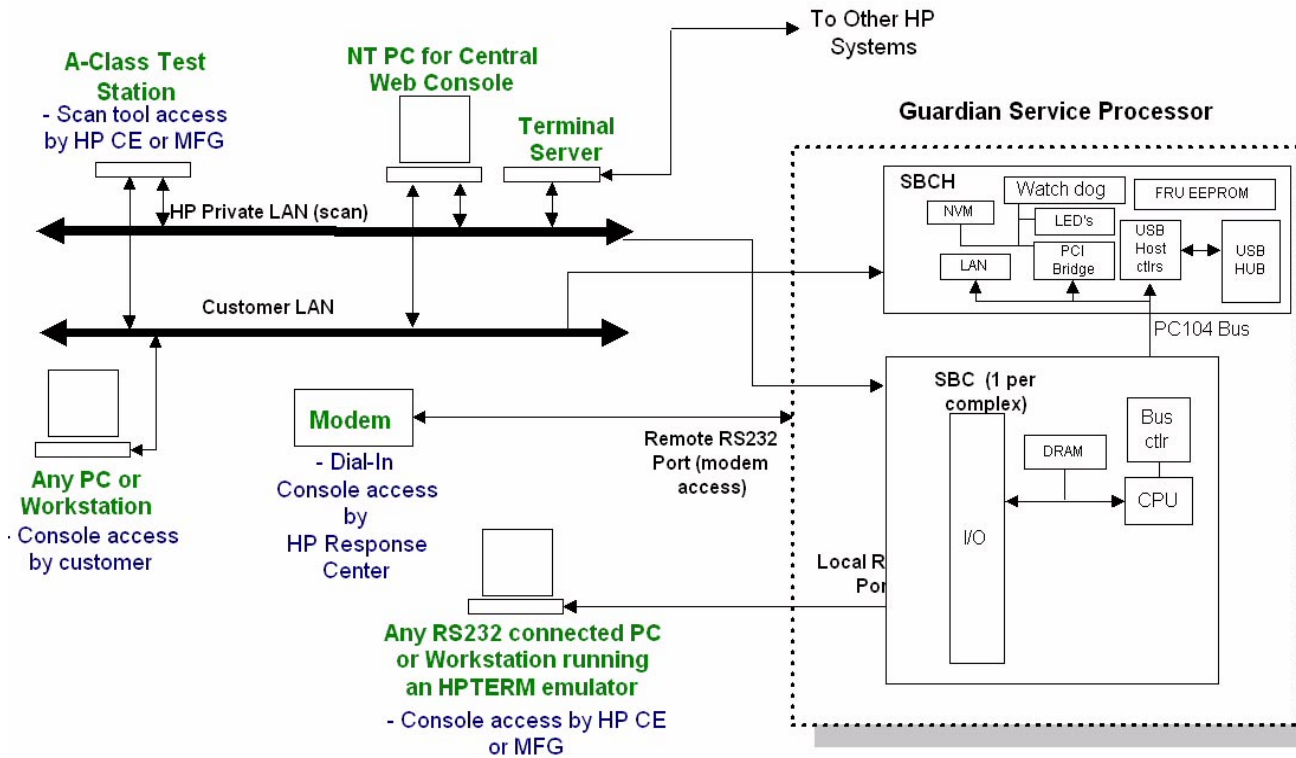
- you need more than 48 I/O slots per CPU cabinet; or
- you need to attach I/O to more than four cells in a CPU cabinet.

For more information see:

- “Cell I/O” on page 32.
- “Partitions, Cells and I/O Chassis” on page 57.

The Console and the Support Management System

Block Diagram of Console Connections to SuperDome



Support Management Station (SMS)

The Support Management Station (SMS), sometimes called a scan station, is an HP-UX workstation (A500 or equivalent) that runs SuperDome diagnostics and testing tools. These tools are used by HP Customer Engineers or Service Engineers to monitor the health of your SuperDome systems and to aid in system upgrades and hardware replacement.

An SMS can monitor up to sixteen SuperDome complexes; the intention is that there should be one per site or data center. It is best to install the SMS near the complex or complexes it manages, so that the diagnostic tools and the hardware under inspection are all in one spot.

The SMS is connected to the complexes it monitors by means of a dedicated LAN; a second LAN card can be connected to the local site LAN. If the SMS monitors more than one complex, it will require a hub for the dedicated LAN.

Rules and Guidelines for Configuring a Complex

This section contains information on the following topics:

- “Recommendations for Cabling Crossbar Controllers (XBCs)” on page 48.
- “Choosing Cells for Partitions” on page 49.
- “Partitions, Cells and I/O Chassis” on page 57.
- “Checklist for Performance” on page 65.
- “Checklist for High Availability” on page 66.

Recommendations for Cabling Crossbar Controllers (XBCs)

Before you go on, read:

- “What is a Partition?” on page 22.
- “What is a Cell?” on page 27.
- “What is an XBC (Crossbar Controller)?” on page 36:
 - “Crossbar Connections” on page 38.

Other terms and concepts:

- **32-way, 64-way-capable system:** see “What is a CPU Cabinet?” on page 23.
- **CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **Complex:** see “What Is a Complex?” on page 16.

When Do You Need To Think about Cabling?

Use this section when you are creating a new **complex**, or adding a **CPU cabinet** to a complex (or removing a CPU cabinet).

The crossbar controllers (**XBCs**) within a CPU cabinet are connected to each other by one direct connection which is not configurable - it's always there. You need to decide whether to use **U-Turn** or **Cross-Flex** cabling to connect the two remaining ports; see “Crossbar Connections” on page 38 for explanations and diagrams.

Guidelines for Performance

- **Always configure a 32-way-capable system with U-Turn.**

U-Turn provides twice the bandwidth between XBCs in the same cabinet.

- **Always configure a 64-way-capable-system with Cross-Flex.**

Cross-Flex increases bandwidth between cabinets in a 64-way-capable-system, providing the best performance for large partitions.

U-Turn is not recommended in a 64-way-capable-system because it would reduce the bandwidth between the cabinets; for example, there would be no direct connection between XBC 4 and XBC 8 in a U-Turn 64-way-capable-system, resulting in two hops for any communication between cells on these crossbars.

Choosing Cells for Partitions

Before you go on, read:

- “What is a Partition?” on page 22.
- “What is a Cell?” on page 27:
 - “Core Cell” on page 33
- “What is an XBC (Crossbar Controller)?” on page 36:
 - “Crossbar Connections” on page 38.

Other terms and concepts:

- “Recommendations for Cabling Crossbar Controllers (XBCs)” on page 48.
- **16-, 32- 64-way-capable-system, CPU cabinet:**
see “What is a CPU Cabinet?” on page 23.

Points to Note

- A partition is similar to a conventional single system. This means that the partition as a whole has access to all the resources in it: processors, memory and I/O are all shared among all the cells in the partition.
- A **16-way-capable system** should have at least two cells, a **32-way-capable system** should have at least four cells, and a **64-way-capable system** should have at least eight cells.

Building a Complex from Scratch

When building a complex from scratch, begin with the largest partition and proceed to the smallest.

Put the first cell of the largest partition in slot 0 of the left CPU cabinet, then start the next largest partition in the lowest-numbered of the remaining empty slots, and so on. Systems configured at the factory are built according to the following algorithm:

- Allocate partitions from largest to smallest.
- When starting a new partition:
 1. fill empty cabinets before adding to a partially populated cabinet;
 2. within a partially populated cabinet, next fill empty **quads** (that is, slots 0-3 or 4-7 in a cabinet);
 3. fill remaining slots from left to right by first filling in the even-numbered slots, then the odd.

Exception: Partitions larger than six cells in a 64-way-capable system are *not* built in a simple left-to-right fashion. For example, if you have a factory-configured eight-cell partition, you will probably notice that it is spread across three crossbars, with six cells in one cabinet and two in the other. A 64-way-capable system has only one link between any two crossbars (see the Cross-Flex cabling diagram under “Cross-Flex and U-Turn” on page 39), and confining the eight-cell partition to two crossbars would overload that link. Testing shows that spreading this partition across three crossbars results in much better performance.

If you are planning to configure or reconfigure a large partition (seven cells or more) in a 64-way-capable system, consult your HP Service Engineer or Customer Engineer for specific guidance.

Starting-slots for new partitions. The factory chooses starting slots in the following order, using the lowest priority-number that corresponds to an empty slot with enough empty slots to the right of it to build the partition:

Priority	Cabinet-Slot	Priority	Cabinet-Slot
1	0-0	9	0-1
2	1-0	10	1-1
3	0-4	11	05
4	1-4	12	1-5
5	0-2	13	0-3
6	1-2	14	1-3
7	0-6	15	0-7
8	1-6	16	1-7

Guidelines for Performance and High Availability

In general, what you do to improve performance will also improve availability. Exceptions are noted below.

Read the guidelines that follow in conjunction with “Partitions, Cells and I/O Chassis” on page 57.

When To Add Cells

- **Use all the possible resources within existing cells before adding cells to a partition.**

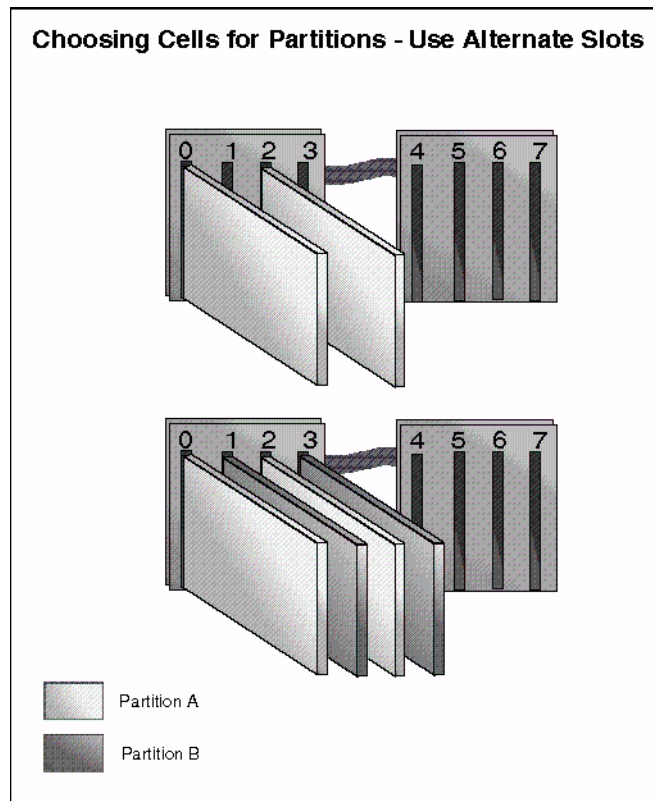
Example: If you need eight processors and 32 GB of RAM, configure a partition containing two cells with four active processors and 16 GB of RAM each, rather than four cells with two active processors and 8 GB of RAM each.

This guideline will help ensure the best performance; but there may be cases in which you want to add cells for the sake of high availability.

Where To Add Cells

- **If the cells in a given partition will not fill all four of the slots in a given crossbar, plug the cells into alternate slots.**

Explanation: Pairs of slots (0 and 1, 2 and 3, etc.) share ports on the crossbar controller (**XBC**). To even out traffic, a partition that uses only two slots in a crossbar should use slots that don't share ports, such as 0 and 2, or 1 and 3. For example, configure two two-cell partitions in a single cabinet by assigning slots 0 and 2 to the first partition and slots 1 and 3 to the second.



Rule of thumb: When starting to add cells to a crossbar that has no occupied slots, fill the even slots first, then the odd slots.

- **If a partition comprises four cells or fewer, all the cells should be connected to the same crossbar.**

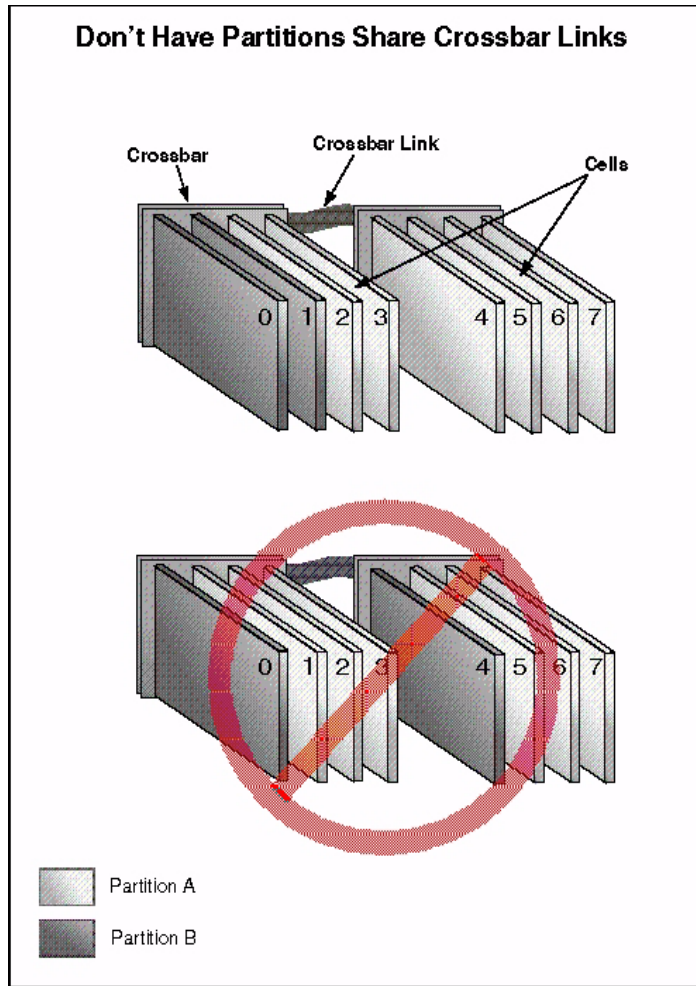
For example, two four-cell partitions should be configured such that the cells of each plug into a single crossbar.

Explanation: There are two crossbars to a CPU cabinet, and a maximum of four cells can plug into each. The crossbars are linked by means of their crossbar controllers (XBCs) but traffic that does not cross an XBC link is faster than traffic that does. In addition, a partition that depends on only one XBC is less vulnerable to hardware failure than a partition that depends on two.

Exception: A 64-way-capable system containing a large partition may need to break this rule. For example, if you have a system with an eight-cell partition and two four-cell partitions, one of the four-cell partitions is probably spread across two crossbars, one in each cabinet. In this large configuration the goal is to spread the load for the complex as a whole across the available links, getting good performance for all partitions rather than optimal for some and much worse for others. See the **Exception** under “Building a Complex from Scratch” on page 50.

- **Do not configure partitions such that two or more partitions share the same crossbar link.**

Explanation: A crossbar link is the link between the two crossbar controllers (XBCs) in a CPU cabinet. A partition comprising more than four cells must use a crossbar link, since there is one XBC per crossbar and a maximum of four cells can plug into one crossbar. A six-cell partition, for example, must use a crossbar link. When loading an empty cabinet, you should configure a six-cell partition such that it occupies all the slots (0-3) on the left crossbar and two of the slots on the right crossbar; you could then use the remaining two slots on the right crossbar for a two-cell partition. But you should not configure a six-cell partition such that it occupies three slots on each crossbar and then configure a second, two-cell partition from the remaining slots, because both partitions would then be using the same crossbar link. That could result in slower traffic and the loss of both partitions if the link fails.



Exception: A 64-way-capable system containing a large partition may need to break this rule. See the **Exceptions** under the previous guideline and “Building a Complex from Scratch” on page 50.

Distributing Resources

- **Each partition should consist of at least two cells.**
- **Each partition should contain at least two viable core cells.**

See “Partitions, Cells and I/O Chassis” on page 57 for more information.

- **Each cell should have at least two active CPUs.**
- **Configure cells into partitions by powers of two if possible.**

Because of the way memory interleaving is implemented (that is, memory sharing among cells) you will get the best memory performance from partitions comprising two, four, eight or sixteen cells. (Single-cell partitions also provide good memory performance, but are not recommended for other reasons.)

- **Configure the same number of active processors and the same amount of memory for each cell in a partition.**

Partitions that conform to this guideline will perform better than those that do not, because an uneven distribution of resources will result in an uneven workload, overworking some connections and underusing others.

- **Each cell should have at least eight memory DIMMs (Dual Inline Memory Modules), for a minimum of 4 GB RAM per cell at first release.**

Explanation: Memory is packaged in four-DIMM increments. A failure of one DIMM will cause the remaining three in the package to shut down. If the cell has only four DIMMs in all (2 GB at first release), it will be unbootable until the defective memory is replaced, whereas a cell that has at least eight DIMMs (4 GB at first release) can be rebooted even if four DIMMs are inactive.

- **Add memory to cells in increments of eight DIMMs (two packages of four DIMMs each).**

At first release, this means each cell’s total memory should be evenly divisible by 4 GB.

Explanation: There are two memory busses on the cell board. When the busses are loaded equally, memory traffic can be spread evenly across them to provide the maximum bandwidth.

Guidelines for Expandability

When populating a CPU cabinet, plan, as far as possible, not only for your immediate needs but also for what you may want to add in the future.

For example, if you define four 2-cell partitions in a 32-way-capable-system, you are leaving no room for expansion in that cabinet. If you have only the one 32-way-capable-system and you need the four partitions, then this configuration may be the only option, but suppose you need only two 2-cell partitions to begin with. In that case it would be best to assign two cells from each crossbar to each partition. Now you have room to expand either or both partitions to four cells while still keeping each partition within the confines of a single crossbar.

Partitions, Cells and I/O Chassis

Before you go on, read:

- “What is a Cell?” on page 27:
 - “Cell I/O” on page 32.
 - “Core Cell” on page 33.
- “What is an I/O Chassis?” on page 41:
 - “Core I/O” on page 42
- “What is an I/O Expansion Cabinet?” on page 44.
- “What is a Partition?” on page 22.

Other terms and concepts:

- **16-, 32-way, 64-way-capable system; CPU cabinet:** see “What is a CPU Cabinet?” on page 23.
- **Complex:** see “What Is a Complex?” on page 16.
- **GSP:** see “What is the Guardian Service Processor?” on page 20.
- **System Bus Adapter:** see “Cell I/O” on page 32.

Points To Note

- In a **partition** containing more than one **cell**, all I/O is accessible from all the cells.

For example, in a four-cell partition in which two cells are attached to **IO chassis**, code being run by a processor in one of the cells that is not attached to an I/O chassis will have the same access to disks and other I/O devices as the cells to which the I/O chassis are physically attached.

Each partition must have I/O, meaning that at least one cell in each partition must be attached to an I/O chassis, but this does not mean every cell in every partition needs to be attached to an I/O chassis (in fact, that is not possible without an **IO expansion cabinet**.)

NOTE

I/O expansion cabinets are not available with early shipments of SuperDome. Contact your HP Sales Representative for up-to-date information.

- A cell must be active in a partition before the partition can use an I/O chassis attached to that cell.

That is, the cell that is attached to the I/O chassis must not only have been assigned to the partition, but also powered on and booted; see “What Happens when a Cell Boots” on page 35.

- A **16- or 32-way-capable system** should have at least two I/O chassis, and a **64-way-capable system** should have at least four I/O chassis.
- The failure of a cell or an I/O chassis will bring down the partition it belongs to, but you can provide redundancy that will allow you to reboot the partition with a minimum of downtime; see “Guidelines for High Availability” on page 62.

When planning or reconfiguring a **complex**, first map out the partitions, then plan the I/O for each partition as you would for a single system, then assign the I/O to individual cells following the rules and guidelines below.

Loading and Assigning I/O Chassis

As shipped from the factory, I/O chassis will be loaded into the **CPU cabinet** and assigned to cells in the following order:

I/O Bay#	Chassis#	Position
1	3	rear right
0	1	front left
1	1	rear left
0	3	front right

It’s a good idea to keep to this order when you add I/O chassis to an installed system.

Given the above layout, populate an empty CPU cabinet as follows.

Attach the first I/O chassis (I/O chassis 3 on the right of I/O bay 1, in the rear of the CPU cabinet, or in the rear of the left CPU cabinet if this is a 64-way-capable system) to the lowest-numbered cell in the lowest-numbered partition in the complex. This chassis should contain a **core I/O** card.

Then continue by assigning the second I/O chassis to the first cell in the next partition; continue until all partitions in (or beginning in) this CPU cabinet have an I/O chassis inside the CPU cabinet attached to their lowest-numbered cell; each of these chassis should contain a core I/O card. Then start over with the first partition and assign additional chassis (including chassis in an expansion cabinet, if any) to the partitions as needed.

When loading and assigning chassis, keep the following considerations in mind (some of these are discussed in more detail in the rules and guidelines that follow):

- A partition's **core cell** (the cell it boots from by default) should normally be its lowest-numbered cell.
- Each partition's core cell should be attached to an I/O chassis inside the CPU cabinet if possible.
- When configuring additional (alternate) core cells for a partition, use the partition's next lowest-numbered cells, and, if the partition crosses CPU cabinet boundaries, use the cells in the left cabinet first.

For example, if you want to attach core I/O to three cells in a partition that comprises seven cells in the left cabinet and two in the right, attach the three chassis containing core I/O cards to the partition's three lowest-numbered cells in the left cabinet.

- The I/O chassis in the rear of the CPU cabinet are easier to work with.
- It is better not to cross **System Bus Adapter** cables.

The four cables on the left should attach to the four left cells and the four cables on the right to the four right cells.

- Allow for future expansion.

For example, if you plan to add cells to this CPU cabinet in the future, and they will form a new partition, leave a slot to add an I/O chassis (to be attached to the new partition's core cell). See "Guidelines for Expandability" on page 64.

Rules

- Each I/O chassis can be connected to only one cell, and each cell can be connected to only one I/O chassis.

See “Cell I/O” on page 32 for more information.

- Each partition must contain at least one cell that is attached to an I/O chassis containing core I/O. This cell should be the lowest-numbered cell in the partition (the leftmost cell in the partition when you are looking at the cabinet from the front). The I/O chassis should have the following cards:

- The core I/O card itself.

Put the core I/O card in the rightmost slot (slot 0; this is the only slot it can go in).

- The boot device controller.

Put the boot device controller in slot 4 of this chassis if it is a 4X card, or in slot 1 if it is a 2X card.

- A removable-media controller card (e.g., for a DVD drive).

Put the card for a removable-media device in slot 8 of this chassis.

- A networking card.

Use this for the partition’s connection to the main LAN (e.g., your site LAN). (The networking connection in the core I/O card is not the best choice for this.)

This cell will be the **core cell**, used for booting the partition; and it will be by default **PDC**’s first choice for booting a partition that contains more than one **viable core cell** (that is, more than one cell attached to an I/O chassis containing core I/O; see “Guidelines for High Availability” on page 62).

See also “Core Cell” on page 33.

- A cell can be attached to an I/O chassis in the same CPU cabinet, or in an expansion cabinet, but not to an I/O chassis in another CPU cabinet.
- A **32-way-capable system** can support only one I/O expansion cabinet; a **64-way-capable system** can support one or two expansion cabinets.

- An I/O expansion cabinet can be used by only one complex.
This means that the two CPU cabinets in a 64-way-capable system can share an expansion cabinet, but two 32-way-capable systems (that is, single cabinets not combined into a 64-way-capable system) cannot.

NOTE

The two CPU cabinets in a 64-way-capable system can share the *first* expansion cabinet, but the second can be used by cells in the right CPU cabinet only.

Guidelines for High Availability

- Make sure that more than one cell in each partition has an I/O chassis containing core I/O (partitions containing only one cell, or connected to only one I/O chassis, are not recommended).

The cells attached to chassis containing core I/O should be the partition's lowest-numbered cells. Each I/O chassis that contains core I/O should also contain cards for a boot device and networking, placed as described under "Rules" on page 60, but only one chassis with core I/O also has to have a removable-media controller card (this chassis should normally be the one attached to the lowest-numbered cell).

This redundancy will allow you to reboot the partition and continue to use it if any one cell connected to core I/O should fail in any respect (e.g., a failure in the cell itself, the I/O chassis it's attached to, or the core I/O card in the I/O chassis).

- As far as possible, keep all the I/O chassis for a single partition in a single cabinet.

Add an expansion cabinet only when all the slots in all the I/O chassis in the CPU cabinet(s) have been used up; and in a 64-way-capable system, add a second expansion cabinet only when all the available slots in the first expansion cabinet have been used up.

But see "Guidelines for Expandability" on page 64 for a possible exception.

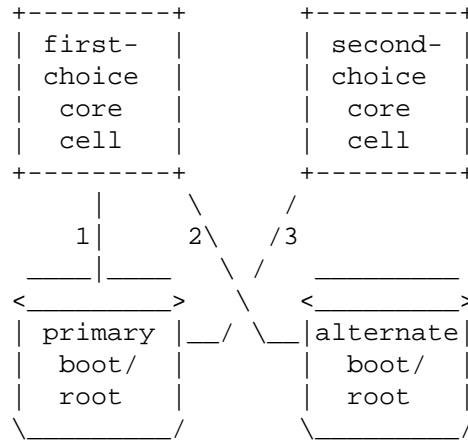
- If you are using an I/O expansion cabinet, make sure each partition has at least one cell attached to core I/O *inside the CPU cabinet*.
- Put a core I/O card and a boot card in every I/O chassis.
- Within a partition, spread redundant cards across I/O chassis such that if one I/O chassis fails, you still have sufficient connections to run the system.
- Attach critical disk devices to cards in more than one I/O chassis, and hence to more than one cell.

Some disk devices (such as a disk array with two SCSI controllers) can be connected to more than one card. Use such devices for root and boot disks, and also for disks containing mission-critical data. This will allow you to get to the data on the disk even if the cell to which it is primarily connected is disabled (or one card or any single connection fails).

Paths to boot and root disk

For a partition performing mission-critical tasks, configure the following paths to the boot device:

1. Primary attached to one viable core cell.
2. Secondary attached to the same viable core cell.
3. Primary attached to a second viable core cell:



This allows for automatic reboot upon any single failure.

In the above diagram, the “first-choice core cell” should be the lowest-numbered cell in the partition; the “second-choice core cell” should be the second-lowest-numbered cell in the partition.

- Attach more than one viable core cell in each partition to the network.
 This will allow the partition to reboot and continue to function even if one core cell is disabled, or one card fails.
- Place cards in an I/O chassis in the following order (bearing in mind that the core I/O card, if there is one, must always be in the first slot):
 1. boot device cards
 2. data device cards
 3. networking cards
 4. other cards

Guidelines for Performance

- If more than one cell in a partition is attached to an I/O chassis, spread out the I/O devices amongst the cells.

Performance will probably not be as good if all or most I/O operations go through one cell.

- Do not use the core I/O card as the partition's main connection to the network (to your site LAN, for example).
- Within a partition, assign 4X cards evenly across I/O chassis to even out the bandwidth. See "I/O Cards" on page 41 for more information.
- Make sure all 4X I/O cards are in 4X slots (see "I/O Cards" on page 41).
- Within an I/O chassis, spread the cards out as evenly as possible.

The **System Bus Adapter** that connects the chassis to its cell is split into two halves, one serving each half of the chassis and each with its own cache, so it is best to distribute the load between the two halves.

Guidelines for Expandability

There is one case in which you may want to consider ordering an I/O expansion cabinet even if all the I/O chassis inside the CPU cabinet have not been used up.

Suppose for example you have a CPU cabinet containing a single partition comprising six cells and using three out of the four possible I/O chassis inside the CPU cabinet. If the slots in these three chassis are all used and you need to attach more I/O devices, it may not be a good idea to begin filling up the fourth I/O chassis.

If you ever intend to create a new partition using the two remaining cell slots in this cabinet, you should reserve the fourth I/O chassis for that partition, so as to meet the high-availability requirement that every partition in (or beginning in) a CPU cabinet should have a cell attached to core I/O inside that cabinet.

In this case, you should order an I/O expansion cabinet to accommodate the I/O needs of the original partition. (I/O expansion cabinets will be available shortly after the first SuperDome shipments.)

Checklist for Performance

This section summarizes the recommendations in “Recommendations for Cabling Crossbar Controllers (XBCs)” on page 48, “Choosing Cells for Partitions” on page 49 and “Partitions, Cells and I/O Chassis” on page 57. Explanations are in those sections or as noted in parentheses below.

- ❑ A 32-way-capable-system should be connected in a **U-Turn configuration**; a 64-way-capable system should be cabled in a **Cross-Flex** configuration.

Processors and Memory

- ❑ Each cell in a partition should have the same number of active processors.
- ❑ Each cell in a partition should have the same amount of memory (see “Memory” on page 32).
- ❑ Each cell’s total memory should be evenly divisible by 4 GB.

Cell Placement

- ❑ If the cells in a given partition will not fill all four of the slots in a given crossbar, plug the cells into alternate slots.
- ❑ If a partition comprises four cells or fewer, all the cells should be connected to the same crossbar.
- ❑ Do not configure partitions such that two or more partitions share the same crossbar link.

Distributing Resources

- ❑ Use all the possible resources within existing cells rather than adding cells to a partition.
- ❑ All 4X I/O cards should be in 4X slots (see “I/O Cards” on page 41).
- ❑ Do not use the core I/O card as the partition’s main connection to the network (to your site LAN, for example).
- ❑ If more than one cell in a partition is attached to an I/O chassis, spread out the I/O devices amongst the cells.
- ❑ Within a partition, assign 4X cards evenly across I/O chassis (see “I/O Cards” on page 25).
- ❑ Within an I/O chassis, spread the cards out as evenly as possible.

Checklist for High Availability

In many cases, best practices for high availability are the same as those for performance, though the underlying reasons are different.

This section summarizes the recommendations in “Recommendations for Cabling Crossbar Controllers (XBCs)” on page 48, “Choosing Cells for Partitions” on page 49 and “Partitions, Cells and I/O Chassis” on page 57. Explanations are in those sections.

- Each cell should have at least eight memory DIMMs (Dual Inline Memory Modules), for a minimum of 4 GB RAM per cell at first release.
- Processors and Memory**
- Each cell should have at least two active processors.
- Cell Placement**
- If a partition comprises four cells or fewer, all the cells should be connected to the same crossbar.
 - Do not configure partitions such that two or more partitions share the same crossbar link.
- Distributing Resources**
- Each partition should consist of at least two cells.
 - Each partition should contain at least two viable core cells.
 - Each cell should have at least two active CPUs.
 - As far as possible, keep all the I/O chassis for a single partition in a single CPU cabinet.
 - If you are using an I/O expansion cabinet, make sure each partition has at least one cell attached to core I/O *inside the CPU cabinet*.
 - Put a core I/O card and a boot card in every I/O chassis.
 - Within a partition, spread redundant cards across I/O chassis such that if one I/O chassis fails, you still have sufficient connections to I/O devices to run the system.
 - Attach any critical disk device to cards in more than one I/O chassis, and hence to more than one cell.
 - Attach more than one cell in each partition to the network.