Superdome Management Part II: Partition Management and Reconfiguration

By

Mingyan Bao
Jim Darling
Bryan Jacquot

Hewlett Packard Company
Manageability Solutions Laboratory
3404 E. Harmony Rd.
Fort Collins, CO 80528
(970) 898 – 7335
(970) 898 – 2151 fax
bao_mingyan@hp.com
jim_darling@hp.com
bryan_jacquot@hp.com

1. Introduction

This paper is the second of two papers that describe managing Hewlett Packard's new Superdome servers. It builds on the concepts introduced in the first paper, providing information related to creating and modifying partitions. The key concepts covered in this paper are:

1. The Complex Profile, what it contains, how it gets changed, how changes are synchronized via locks, and the rules for changing cell assignments to partitions.
2. The process of booting a partition and how this relates to joining cells into a partition.
3. The need for a "core cell" in each partition
4. The "reboot for reconfiguration" process, differentiated from a normal reboot, and its use to activate new cells in a partition.
5. The "shut down for reconfiguration" process, differentiated from a normal shutdown, and its use to make cells inactive so that their partition assignment can be changed.
6. The process to delete a partition.
7. The tools available in HP-UX to manage a Superdome server: the Partition Manager GUI, and the partition commands (such as parcreate(1M) and parmodify(1M)).

Through out this paper, the Partition Manager (parmgr) tool is used to illustrate various configuration tasks and processes.

2. Complex Profile

2.1 Introduction

There is a very important set of data that represents the configurable aspects of a Superdome complex. This data is known as the **Complex Profile**. The Guardian Service Processor (GSP) maintains the Complex Profile, though it is primarily through configuration changes made via Partition Manager or the partition commands (for example, the parcreate(1M) command) that the Complex Profile is modified.

The Complex Profile is divided logically into two portions based on the type of information each contains. The first portion is called the **Stable Complex Configuration Data (SCCD)**. The other portion is called the **Partition Configuration Data (PCD)**.

2.2 Stable Complex Configuration Data (SCCD)

The SCCD contains information that pertains to the entire Superdome complex. This includes the model number, model string, complex serial number, complex system name, product number, and the cell assignment table.

The Cell Assignment Table

The SCCD contains a very important set of data related to the configuration of partitions – the cell assignment table. This table has an entry for each cell in the complex. Cells are globally numbered within a complex from 0 to 15. Each entry in the table either identifies the partition that the cell is assigned to or indicates that the cell is free (not assigned to any partition).

A cell's assignment can be changed only when the cell is not active in a partition. This is to avoid creating an inconsistent state where a cell is being used as a partition of one partition but the SCCD identifies it as belonging to a different partition. The implications of this rule are explained in the sections that cover creating a partition, adding cells to a partition, and especially removing cells from a partition.

2.3 Partition Configuration Data (PCD)

The PCD contains information about each partition. One entry exists for each possible partition. Partitions are globally numbered from 0-15. The data in the PCD for a partition includes that partition's boot paths, core cell choices, partition name, use-on-next-boot flags, console path, and keyboard path.

Use-on-next-boot Flag
The use-on-next-boot flag is an important PCD field because it determines whether each cell in a partition may be active. There is one flag per cell. If the flag is set and the cell is assigned to the partition then the cell is used if possible when the partition is booted. Thus, if this flag is not set then a cell is not used. This can be useful when needing to remove a cell from a partition (see section **3.5.7 Other Ways to Remove a Cell**), and it is useful to avoid long boot times when a cell is broken (see section **3.4.3 Booting a Partition**).

Core Cell Choices
Every partition has a **core cell** that is selected by system firmware when the partition is booted. From an administration perspective the choice of core cell is mostly a "don't care". The core cell must be a cell that is connected to an I/O chassis that contains a core I/O card because the core cell provides console access for the partition. Relatively small amounts of memory in the core cell are used for special purposes by both firmware and the kernel. HP also recommends that a partition's primary (PRI), and high-availability alternate (HAA) boot devices should be attached to cards in the I/O chassis that is attached to the core cell (though this is not required).

There is a mechanism that allows the administrator to influence system firmware's choice of a partition's core cell. This mechanism is a list of core cell choices kept in the partition's Partition Configuration Data. Partition Manager, parcreate(1M) and parmodify(1M) allow the administrator to specify core cell choices. The core cell choices identify the order that firmware should use in finding a viable core cell.

Firmware's default selection algorithm is to use a cell that:

- Is the lowest numbered cell assigned to the partition that is attached to an I/O chassis with core I/O
- Is to be included in the partition (the use-on-next-boot flag is set)
- Doesn't have any detectable (via self tests) hardware problems that would keep the cell from being used

If there are core cell choices specified then firmware would start with those cells first in the search for a core cell, then move on to its default algorithm if necessary.

Specifying core cell choices is optional. It is most likely useful if:

- There is a reason for connecting a partition's boot devices to a cell other than the lowest numbered cell that is attached to an I/O chassis.
- There is a reason, such as a possible hardware problem, for not using the lowest numbered cell.
- The goal is to eventually remove the lowest numbered cell with core I/O from the partition.

2.4 Changing the Complex Profile

The Complex Profile is not changed directly by an administrator. Rather, it is changed as a result of performing a task via Partition Manager or one of the HP-UX partition commands. Therefore, the details of how changes are made to the Complex Profile are not normally of interest to an administrator. However, it is important to know that there is a mechanism used to ensure that only one set of changes can be made to any part of the Complex Profile at a given time.

This mechanism is a set of locks that are managed by the GSP. There is one lock for the SCCD and one lock for each partition's entry in the PCD. An administrator is never directly involved in acquiring locks

(the tools handle that), but an administrator might run into situations where they can't perform a task because the tool they are using can't get the locks needed to perform the task.

To illustrate the use of locks on the Complex Profile, consider the following example in which one administrator (Joe) is using Partition Manager to create a partition, and another administrator (Susan) tries to use the parmodify command to add a cell to a partition.

1. Joe selects the "Create Partition**"** task in Partition Manager.
2. Partition manager acquires the lock on the SCCD since creating a partition will require changing the cell assignment array for those cells to be put into the new partition. This lock is aquired right away so that no changes can occur to the cell assignments while Joe is working on specifying the attributes of the new partition.
3. Partition Manager gets a copy of the SCCD from the GSP and uses the information in the SCCD to let Joe know what cells are free cells and thus can be assigned to the new partition.
4. While Joe is interacting with Partition Manager, Susan executes parmodify(1M) to assign a cell to a partition that she manages.
5. The parmodify command attempts to lock the SCCD (since it needs to make a cell assignment change) but that request to the GSP to get the lock fails (because Joe's instance of Partition Manager already has the lock).
6. The parmodify command exits with an error message telling Susan that the SCCD is already locked.
7. In the meantime, Joe completes the create partition task, resulting in changes to the SCCD and clearing of the SCCD lock.

More information related to the Complex Profile locks can be found in appendix **A.3 Trouble Shooting Tips**.

3. Partitions

3.1 Introduction

A partition consists of one or more cells that communicate coherently over a high bandwidth, low latency crossbar fabric. Each partition runs its own independent operating system. Different partitions may be executing the same or different revisions of an operating system.

Each partition has its own independent CPUs, memory and I/O resources consisting of the resources of the cells and I/O chassis that make up the partition. Resources may be removed from one partition and added to another without having to physically manipulate the hardware. This is accomplished by using Partition Manager or the HP-UX partition commands. See section **4. Management Tools** for more information about these tools.

3.2 Configuration Rules

A legal partition must conform to the following rules:

- All cells must have the same revision of system firmware.
- All cells must have the same processor revision (a.k.a. IODC_HVERSION)
- At least one cell must be attached to an I/O chassis that has a core I/O card.

There are also several HP recommended guidelines that should be followed:

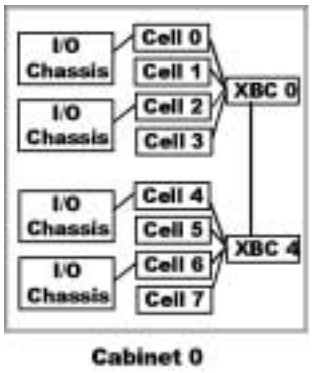
- A partition should have at least two cells – this is so if one fails, the other one can still keep the partition running.
- Memory configuration of all cells should be identical:
    - Same number of DIMMs, preferably a multiple of 8

- Same capacity (size) and locations of DIMMs
- At least 8 DIMMs per cell

- Boot device should be connected to the chassis that contains the active core I/O

3.3 Configuration Process

Below is an illustration of the cells, chassis, XBC chips, and their connections to each other in a complex:

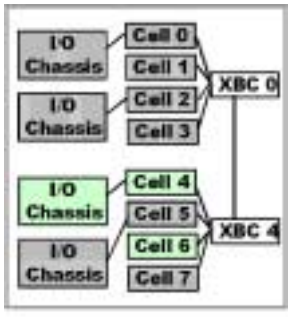


As shown, each cell in the complex is connected to an XBC chip. Cells 0, 2, 4, and 6 are connected to an I/O chassis.

This is the suggested process to follow when determining which cells to assign to partitions:

1. Start with the largest partition and decide which cells to assign to it first. Then move on to the next largest partition and so forth down to the smallest partition.
2. Use cells from a cabinet with all free cells if possible (applies to SD64000).
3. Assign cells to a partition from a XBC chip that has only free cells connected to it if possible.
4. When assigning two cells to a partition from the same XBC, use every other cell if possible. For example, use cells 4 and 6 rather than cells 4 and 5.
5. If necessary to follow this process, change the connections between I/O chassis and cells.

Consider the case where a partition with 16 CPUs is needed (thus 4 cells should be assigned to the partition), and two partitions with 8 CPUs are needed (thus 2 cells should be assigned to each of those partitions). The correct configuration is shown in the following diagram. Note that the I/O chassis that had been connected to cell 6 has to be reconnected to cell 5.



3.4 Creating a Partition

This section describes the process of creating a partition. This includes an overview of the process that cells go through to boot a partition, including the very important concepts of the **Boot Is Blocked** flag, and

inactive versus **active** cells. Finally, it concludes with a brief description of the process for creating the first partition on a Superdome server.

3.4.1 How to Create a Partition

There are two ways to create a partition. The administrator can use either Partition Manager's "Create Partition" task or the parcreate(1M) command. Partition Manager's "Create Partition" task walks the administrator through each step of the create partition process, and automatically invokes parcreate(1M) at the end when all the information is gathered. Once the partition has been created, it needs to be booted and an OS needs to be installed and configured.

Creating a partition consists of assigning **free cells** to a new partition. To move cells from an existing partition to a new partition it is first necessary to remove the cells from the existing partition, then (once they are free cells) assign them to the new partition. The other attributes that can be specified when creating a new partition are:

- **A partition name**. This is a descriptive value that has no relationship to any other names (for example, it is **not** related to the hostname that will be assigned once an operating system is installed and networking is configured). Every partition is assigned (by the tools) a unique partition id (a number 0..15), so the name you choose is an alternative for identifying the partition.
- **Cells' use-on-next-boot flag.** In most cases this flag should be left set to the default ("yes") for all of the cells in the partition. However, should a case arise where a cell is to be assigned to a partition but shouldn't be used when the partition is booted, this flag should be set to "no" for that cell.
- **Core cell choices.** In most cases the algorithm used by system firmware for selecting the partition's core cell should be used. However, should a case arise where there is a reason to deviate from this algorithm, the core cell choices should be set to tell system firmware what order to use when selecting the core cell.
- **Partition boot paths**. The primary, alternate, and high availability alternate boot paths can be set, though only via parcreate(1M), not via Partition Manager. When set via the parcreate command no checking is done that the device paths are valid or that any device actually exists. The alternative is to wait until the partition has been booted, then at the Boot Console Handler search for bootable devices and set the various boot paths.

Both Partition Manager and the parcreate command assure that a new partition has at least one cell that is connected to an I/O chassis that contains a core I/O card, that all cells in the partition have the same processor revision (that is, the same IODC_HVERSION value), and the same system firmware revision. Partition Manager also performs a number of high availability checks and generates warnings if any of those checks fail. For example, fewer than two cells in a partition results in a high availability warning, likewise for a cell with fewer than eight DIMMs. Neither tool checks for recommended cell combinations. The planning process should ensure that the set of cells selected for a partition meet the various HP recommendations, or understand the potential consequences of not adhering to those recommendations.

The following changes get made to the Complex Profile when a partition is created. Both parts of the Complex Profile are changed immediately, thus no more than a few seconds passes between executing the parcreate command and the partition being created.

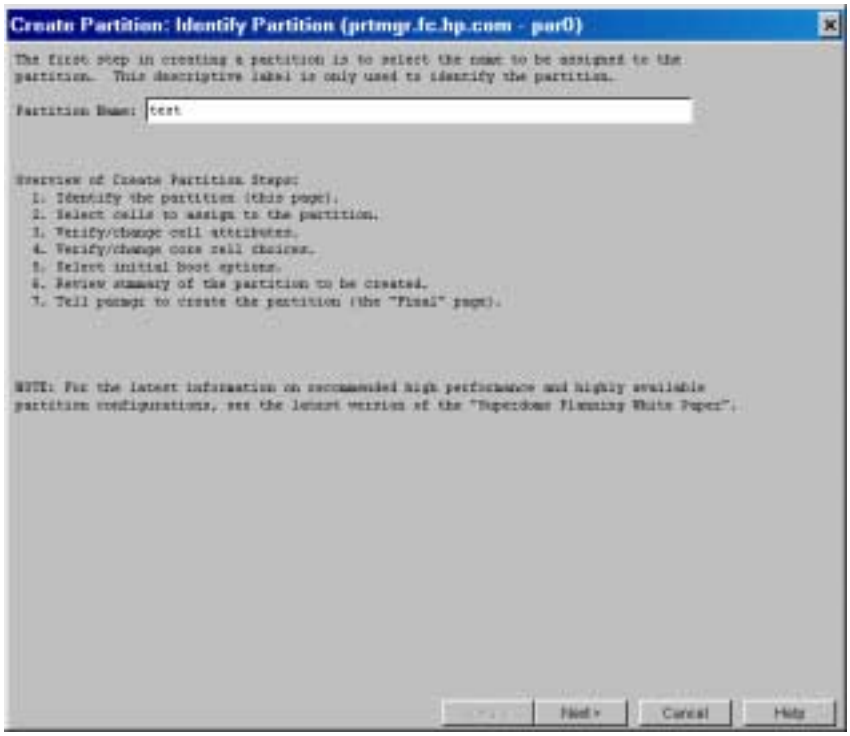
- The new partition's PCD is changed to include the partition name, the cells' use-on-next-boot flags, the partition's core cell choices, and the partition's boot paths.
- The SCCD is changed to reflect the new cell assignments.

Once a new partition has been created, it can be booted manually from the GSP's Command Menu using the boot ("BO") command. The alternative is to use options provided by Partition Manager and parcreate(1M) to have the new partition automatically as soon as it has been created. For the parcreate command this is the -B option. For Partition Manager, one of the steps in the "Create Partition" task wizard provides the option of booting automatically once the partition has been created.

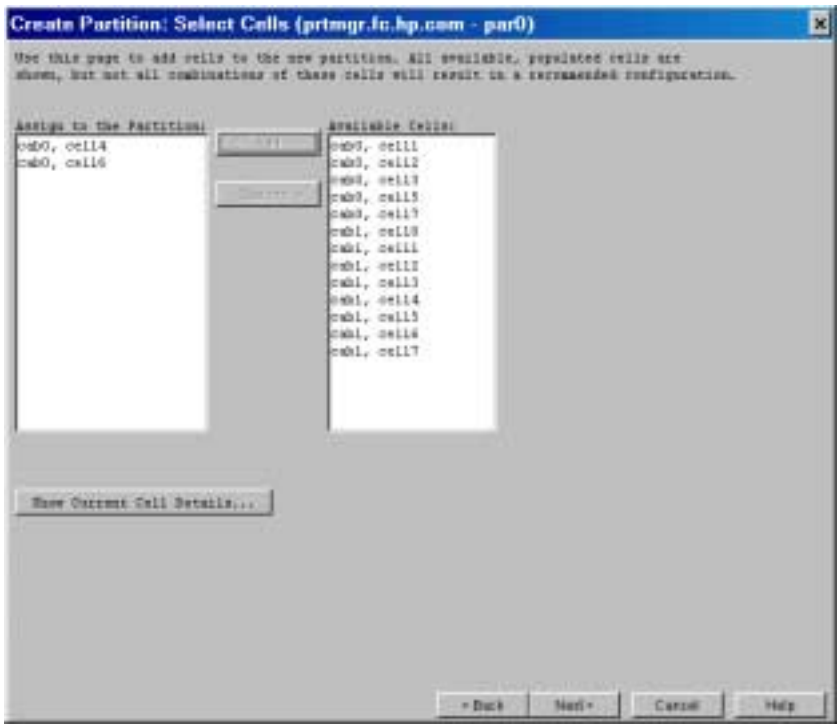
3.4.2 Example of Creating a Partition Using Partition Manager

The administrator can invoke the task to create a partition by going into Partition Manager and selecting the “Create Partition” task from the **Partition** menu. An alternative way to invoke this dialog is to select "Available Resources" in the left pane of Partition Manager's main window, then highlight one or more of the free cells that are displayed in the right pane of the main window, and then selection the "Create Partition" task from the **Partition** menu. When this approach is used the highlighted free cells are automatically designated to be assigned to the new partition (see the second screen shot below).

The “Create Partition” task is a task wizard that consists of seven steps. Shown below is the first page of the task wizard. It includes a selector for providing the name of the new partition, and it includes any overview of the entire task wizard.



The second page (or window) in the “Create Partition” task is shown below. This is the most important page in the task wizard as it is where the cells to be assigned to the new partition are identified. All of the free cells in the Superdome system are shown (see the right column labeled "Available Cells"). The user moves the cells to be assigned to the partition to the other column (labeled "Assign to the Partition"). If free cells were highlighted in Partition Manager's main window when the “Create partition” task was selected then those cells are automatically put in the left column.



In contrast, here is an example of the parcreate(1M) command. The comments explain the various options that have been used. This command creates a partition with two cells. The options specify cell 6 as the first core cell choice and cell 4 as the second core cell choice (the opposite of the default)

```
/usr/sbin/parcreate -P 'test' \   # partition name
    -c 4::: \                     # add cell 4
    -c 6::: \                     # add cell 6
    -r 6 -r 4                     # core cell choices
```

The option for specifying a cell deserves a bit of explanation (e.g., "-c 6:::"). The partition commands have been designed with an eye towards the future where additional cell attributes may exist. This is an example of that. The definition of the -c option is:

-c cell_id:cell_type:use_on_next_boot:failure_usage

The cell type (which would differentiate how cells are used) and Failure Usage flag (which would specify whether or not to use a cell if certain self-tests fail during the power-on self-test sequence - see the **Booting a Partition** section below for an explanation of this sequence) are examples where only one option is available today but additional options may be available in the future. Therefore, the construct "-c 6:::" in effect says to add cell 6 with the default values for cell type, use-on-next-boot, and Failure Usage.

3.4.3 Booting a Partition

When a partition is booted its cells go through a process that consists of several steps in order to become active and usable by the partition. The process of booting a partition is complicated by the fact that it is not just a simple matter of turning on the system and letting it boot (as occurs on servers that don't support partitions). Instead, the boot process on Superdome is broken down into two distinct phases – the **power-on self-test phase** and the **partition rendezvous phase**. These phases are described below.

Power-On Self-Test Phase

The power-on self-test phase (also referred to as **POST**) occurs when cells are powered on or reset. Powering on a cell happens when 48V is enabled in a cabinet or when a cell that has previously been powered off gets powered on. A cell reset happens when a partition is shutdown (see the sections **3.5.4**

Making Cells Active and **3.5.6 Doing a Shut Down for Reconfiguration**). In this phase a cell operates independently of all other cells (in contrast to the cooperation that occurs between cells in the **partition rendezvous phase**). The key steps that occur during this phase are:

1. The cell is powered on and the **Boot Is Blocked (BIB)** flag gets set. This is a hardware flag on the cell board. Its use is central to the process of booting a partition.
2. System firmware on the cell performs self-tests and discovery operations on the cell's hardware. This process can take up to several minutes. The time it takes is mostly a function of the amount of memory in the cell (the more memory, the longer this process takes). The steps in this process include:
    a. CPU self-tests
    b. Memory self-tests
    c. I/O discovery
    d. Fabric discovery (i.e., what connections exist between cells, chassis, and XBCs).
3. Once firmware has completed the self-tests and discovery operations it reports the hardware configuration of the cell to GSP and tells the GSP that it is waiting for BIB to be cleared.

> The fact that it is at this point that hardware information is made available is important because administration tools such as Partition Manager get details about cells from the GSP. Therefore, when a cell is powered off and while a cell is performing POST this information is not available.

4. Firmware then waits for BIB to be cleared.

During this phase a cell is said to be INACTIVE. The implication is that the cell's resources (its processors, memory, and any I/O attached to the cell) are not being used by an operating system. In the more general case, inactive is used to mean any cell that is not powered on or is powered on and has BIB set.

Cells won't always proceed through POST at exactly the same pace. Consider what happens when a cabinet is powered on. All of the cells in the cabinet begin POST at only roughly the same time – so right off the bat the cells are not exactly synchronized. As they proceed through POST they will naturally take different amounts of time to complete various steps. For example, a cell with 16 GB of memory will take longer to perform memory self-tests than a cell with less memory. Likewise, a cell attached to an I/O chassis spends more time in I/O discovery than a cell that is not attached to a chassis.

It is because of this variability in how long the POST phase takes, and also to provide control over when partitions boot, that the BIB flag exists. It provides a mechanism to get all of the cells at a common point in their boot process before going on to the second phase, the **Partition Rendezvous Phase**. It also gives the administrator the ability to control when partitions get booted rather than just having them automatically boot as soon as cells are powered on. For example, if the boot command is executed (from the GSP command menu), the GSP will wait until all of the cells in the partition have reported that they are waiting at BIB.

Partition Rendezvous Phase

The partition rendezvous phase occurs when a partition is booted. This happens when the boot command (`BO`) is executed from the GSP Command Menu, and when a partition is rebooted for reconfiguration (the process of rebooting a partition is discussed later in section **3.5.4 Making Cells Active**). In this phase the cells that are assigned to a partition must work together - system firmware on each cell can no longer operate independently.

The GSP initiates booting a partition by performing the following steps:

1. The GSP provides a copy of the Stable Complex Configuration Data to the partition's cells.
2. The GSP provides a copy of the partition's Partition Configuration Data to the partition's cells.

3. The GSP clears BIB for the partition's cells. **As soon as BIB is cleared a cell is considered to be ACTIVE.**

When BIB gets cleared system firmware on each cell goes into rendezvous mode. Using the information in the SCCD and PCD, the system firmware on each cell contacts the other cells in the partition. Working together, the cells do the following:

1. The cells agree that they all belong to the same partition.
2. The cells negotiate the selection of a core cell (this is like choosing a leader of the partition, though the core cell must be attached to an I/O chassis that contains a core I/O card).
3. The core cell manages the rest of the boot process, including displaying the Boot Console Handler user interface, and eventually handing off control to an operating system.

3.4.4 Creating the First Partition

Creating the very first partition on a Superdome server is a special case since there is no operating system booted from which to run Partition Manager or the parcreate(1M) command. For most customers this situation will never arise because Superdome servers are pre-configured by Hewlett Packard according to the customer's specification when the system is ordered. However, there could be occasions where it is necessary to start from scratch.

The way to create the very first partition on a Superdome server, and to then expand that partition and create additional partitions, is to perform the following steps:

1. Log on to the GSP console with the operator or administrator capabilities.
2. Go to the Command Menu and execute the `CC` command (this is the "Initiate a Complex Configuration" command).
3. Select the option "G – Build genesis Complex Profile". This option wipes out any existing partition definitions and creates a new version of the Complex Profile with one partition defined that contains a single cell (you specify which cell to use).
4. Boot the single-cell partition.
5. Install an OS (or perhaps use a previously configured disk with HP-UX installed).
6. Boot the OS.

Use Partition Manager or the partition commands to add cells to the existing partition or to create additional partitions.

3.5 Modifying a Partition

This section describes the process of modifying a partition. The focus is on adding cells to a partition and removing cells from a partition. Included is an explanation of the very important concepts of the **reboot for reconfiguration** and **shutdown for reconfiguration**.

3.5.1 How to Modify a Partition

There are two ways to modify a partition. The administrator can use either Partition Manager's "Modify Partition" task or the parmodify(1M) command. Partition Manager's "Modify Partition" task allows the administrator to specify as few or as many changes as they want to the given partition, and then automatically invokes parmodify(1M) to make the changes.

- Changing the partition name.
- Changing the use-on-next-boot flag for cells assigned to the partition.
- Changing the partition's core cell choices.
- Adding cells to the partition.
- Removing cells from the partition.
- Changing the partition's boot paths (only supported by parmodify, not by Partition Manager).

Changing the partition name, use-on-next-boot flags and core cell choices are fairly simple tasks, unlike adding and removing cells from a partition. Therefore, this paper concentrates on the add/remove cell tasks. Each of these tasks is described separately, though both parmodify and Partition Manager allow multiple changes to a partition (including adding and removing cells) in a single task.

3.5.2 Adding Cells To a Partition

Adding cells to a partition consists of assigning free cells to an existing partition, and then making the cells active. Note that only free cells can be added to a partition, in order to move cells from one partition to another partition it is necessary to first remove the cells from the partition that they are currently assigned to, and then to add them to the second partition. Also, a partition can be active (that is, an operating system can be booted) when cells are added to the partition, though a reboot is required to make the new cells active.

Both Partition Manager and the parmodify command assure that the new cells have the same processor revision (that is, the same IODC_HVERSION value), and the same system firmware revision as the cells that are already assigned to the partition. Partition Manager also performs a number of high availability checks and generates warnings if any of those checks fail (these are the same checks as are performed when a partition is created). Neither tool checks for recommended cell combinations. The planning process should ensure that the set of cells assigned to a partition met the various HP recommendations, or understand the potential consequences of not adhering to those recommendations.

The following changes get made to the Complex Profile when cells are added to a partition. Both parts of the Complex Profile are changed immediately, thus no more than a few seconds passes between executing the parmodify command and the partition being modified (the changes might not occur immediately if cells are also being removed - see the section **Remove Cells from a Partition** for more information).

- The modified partition's PCD is changed to set the values of the new cells' use-on-next-boot flags.
- The SCCD is changed to reflect the new cell assignments.

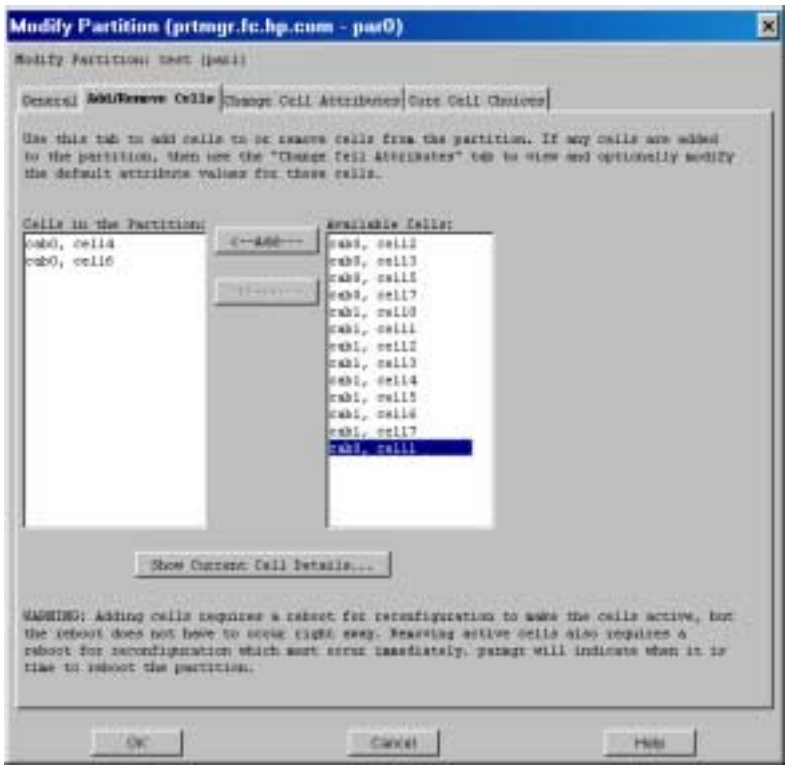
Once the partition has been modified the newly added cells are **inactive**. The process of adding a cell to a partition does not make the cell active. Making the cell active is a separate process that is discussed later in the section **3.5.4 Making Cells Active**.

3.5.3 Example of Adding Cells using Partition Manager

There are two ways to add a cell using Partition Manager:

- Select a partition and use the "Modify Partition" task from the **Partition** menu.
- Select a cell in “Available Resources” and use the “Assign Cell to Partition” task from the **Cell** menu. Only one cell can be selected to use this task.

Both options result in using the “Modify Partition” task dialog. In the second case the selected cell is automatically identified to be added to the partition (see the description of the "Add/Remove Cells" tab of the “Modify Partition” task below). This screen shot shows Partition Manager’s “Modify Partition” dialog with the “Add/Remove Cells” tab in that dialog displayed.

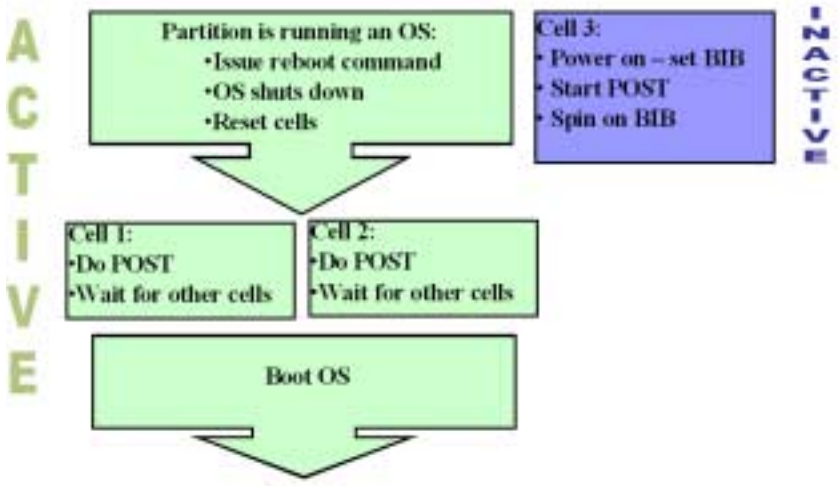


If the “Modify Partition” task was used then the “General” tab is displayed when this dialog comes up, so the administrator has to change to the “Add/Remove Cells” tab to add cells to the partition. If the “Assign Cell” task was used then the “Add/Remove Cells” tab is displayed automatically and the cell selected in “Available Resources” is automatically moved to the “Cells in the Partition” list (not shown in this example).

3.5.4 Making Cells Active

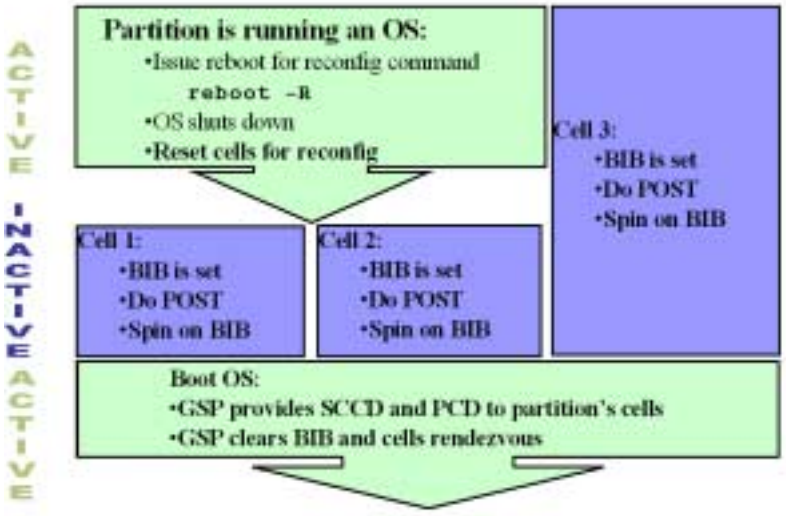
Once a cell (or cells) has been added to a partition a reboot has to be done for the operating system to be able to use the cell's processors, memory, and I/O (if the cell is attached to an I/O chassis). However, a normal reboot won't accomplish the job, a new, special form of reboot known as a reboot for reconfiguration must be done.

The following illustration shows what happens when a normal reboot is done. This process is the same regardless of whether the reboot(1M) or shutdown(1M) command is used, both can be used to cause HP-UX to be shutdown, and then restarted (or rebooted). The partition in this example contains two active cells, and a third cell that has been added to the partition is shown.



When the OS is told to reboot, all of the activity is shut down and the kernel issues a firmware call to reset the cells in the partition. When this reset is done, firmware in each cell performs POST, at the end of POST firmware waits for the other cells to also complete POST, then the process of booting the partition takes place (i.e., a core cell is selected, etc.). **The key here is that this process does not allow a new cell to join the partition.** However, it is important to preserve this process, so a new form of rebooting is needed.

The following illustration shows the process of doing a **reboot for reconfiguration**.



The starting point is the same as before: a two-cell partition that is active, and a third cell that is inactive but has been assigned to the partition. The reboot for reconfiguration is initiated by using the new option "-R" on the reboot(1M) or shutdown(1M) command.

```
# Reboot the partition for reconfiguration using the shutdown command
shutdown –R

# Reboot the partition for reconfiguration using the reboot command
reboot -R
```

When this option is specified, the OS does the normal shut down of activity, and then the following sequence of events happens.

1. The kernel sends an unchanged version of the SCCD to the GSP, identifying the local partition to be booted.
2. The GSP waits for all of the cells in the partition to have BIB set before it makes the new SCCD the current SCCD (the GSP doesn't check that no changes have been made to the SCCD).
3. In the meantime, the kernel issues the firmware **reset for reconfiguration** call (instead of the normal cell reset call). This call results in BIB being set for the cells in the partition.
4. System firmware in each cell performs POST as a result of the reset. When POST is completed each cell notifies the GSP that it is at BIB.
5. Once all of the cells in the partition have reached BIB the GSP writes out the new (though unchanged) SCCD to all cells in the complex and then boots the partition, including the cell that was recently added to the partition.

3.5.5 Remove Cells From a Partition

Removing cells from a partition consists of making cells those cells free cells. The tricky part of this is that a cell's partition assignment can't be changed when the cell is active. Therefore, since removing an inactive cell is easy, this paper focuses on removing an active cell.

A straightforward way to remove a cell is to first do a shut down for reconfiguration of the partition to make all of the cells inactive. Then use Partition Manager or parmodify(1M) on another partition to remove the cells. Finally, boot the partition after the cells have been removed. The GSP can be told to boot the affected partition by Partition Manager's "Modify Partition" task or by the -B option to parmodify(1M). There are other alternatives to this process, which are discussed in the section **3.5.7 Other Ways to Remove a Cell**.

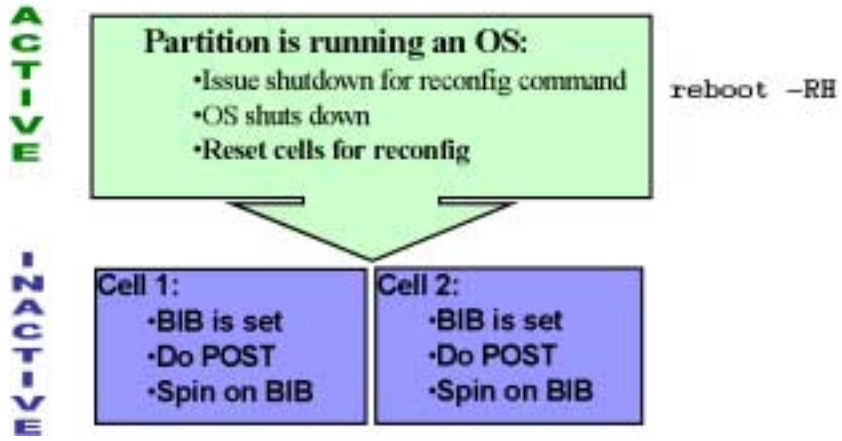
3.5.6 Doing a Shut Down for Reconfiguration

The key point here is that the process of removing cells requires a **shut down for reconfiguration**. A shut down for reconfiguration is like a reboot for reconfiguration except that the partition doesn't automatically reboot.

The following illustration shows a normal shut down.

The key is that with a normal shut down the kernel is left running in a very tight loop. However, from the GSP's and firmware's perspective, the OS is still in control of the cells in the partition, and BIB is still unset, **so the cells are still active.**

The following illustration shows the difference by doing a shut down for reconfiguration. It is just like a reboot for reconfiguration, except that the GSP is not told to boot the partition. Instead the OS only does the firmware reset for reconfiguration. This results in BIB being set, the cells performing POST, and then spinning on BIB. At that point the cells in the partition are inactive, thus allowing cells to be removed from the partition.



Use the new options "–RH" on the shutdown or reboot command to do a shut down for reconfiguration. Remember that using only "–R" results in a reboot for reconfiguration.

```
# Shut down the partition for reconfiguration using the shutdown
# command
shutdown –RH

# Shut down the partition for reconfiguration using the reboot command
reboot -RH
```

3.5.7 Other Ways to Remove a Cell

There are two other ways to approach removing a cell from a partition. These are in effect different alternatives for making the cells to be removed inactive. These alternatives are listed in order of preference for how HP recommends that this be done, with the first choice being to do a shut down for reconfiguration as has already been described.

1. Alternative 1:
   - Set the use-on-next-boot flag to false for the cells to be removed
   - Do a reboot for reconfiguration - the cells to be removed are left inactive.
   - Use Partition Manager or parmodify(1M) on any partition and remove the cells.
2. Alternative 2:
   - Use Partition Manager or parmodify(1M) to change the **local** partition (to remove the cells).
   - Do reboot for reconfiguration of the local partition.

Alternative 1

The idea here is to minimize the time that the affected partition is shut down. The use-on-next-boot flags for the cells to be removed are unset and the partition is rebooted (using the reconfiguration option – without the reconfiguration option the use-on-next-boot flag is not checked when the partition boots). Then the cells that are now inactive can be removed at any time. A possible scenario might be as follows:

1. During normal working hours an administrator unsets the use-on-next-boot flag for the cells to be removed, and leaves instructions for the night shift to do a reboot for reconfiguration.
2. Overnight, when it is okay to do a reboot, the partition is rebooted for reconfiguration. This leaves the cells to be removed inactive.
3. The next morning, the administrator removes the cells from the partition (without disrupting the partition).

Alternative 2

This alternative tends to be the one that people think of first though it should be avoided if possible. Partition Manager or parmodify(1M) must be run on the affected partition (both tools, in somewhat

different ways, will not remove active cells from another partition). In both cases, a modified SCCD is given to the GSP. The GSP will have to wait before writing out the new SCCD because the cells being removed from the partition are active. In order to complete the change, the affected partition must be rebooted for reconfiguration.

The key issue with this alternative is that it results in a pending change to the SCCD. If something should interrupt the reboot for reconfiguration of the affected partition the pending change won't occur (the GSP will continue to wait for the cells to become inactive). When this happens there is no way to find out what changes are pending and it prevents any other changes from being made to the SCCD. See the appendix **A. 3 Trouble Shooting Tips** for more information about what to do in such a situation.

3.5.8 Removing Cells Using Partition Manager

There are two ways to remove a cell using Partition Manager:

- Select a partition and use the "Modify Partition" task from the **Partition** menu.
- Select a cell in the partition and use the "Unassign Cell from Partition" task from the **Cell** menu.

Both options result in using the "Modify Partition" task dialog, much like the two options for adding cells use the same task.

3.6 Deleting a Partition

Deleting or removing a partition results in all of the cells assigned to the partition becoming free cells.

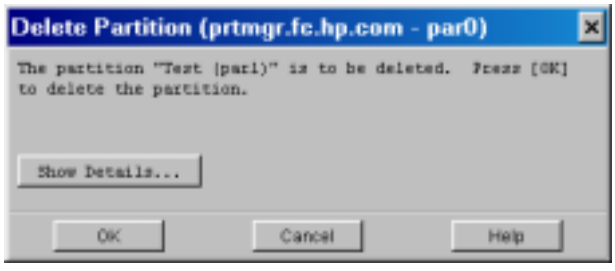
There are two ways to delete a partition. The administrator can use either Partition Manager's "Delete Partition" task, or the parremove(1M) command. Partition Manager uses the parremove(1M) command. The SCCD will be changed to reflect the fact that all the cells in the partition are now free.

In the general case all of the cells in a partition must be inactive before the partition can be deleted. Therefore, a shut down for reconfiguration must be done on the partition to make it inactive. The only exception to this rule is that Partition Manager and parremove(1M) can be used to remove the partition that the tool is being run on. If this is done then the shut down for reconfiguration should be done as soon as the delete partition task has completed (a pending change to the SCCD is in effect until the shut down for reconfiguration is done, just like when removing an active cell).

To remove a partition using Partition Manager:

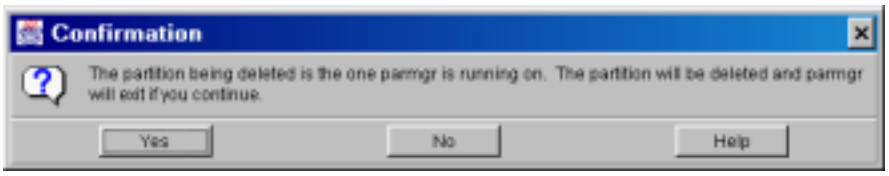
- Select the partition to be removed
- Select the "Delete Partition" task

The following screen shot shows the dialog that is displayed when deleting an inactive partition from another partition in the complex.



Selecting [OK] will remove the partition from the complex.

If Partition Manager is running on the partition to be removed then the following dialog appears.



After selecting [Yes] Partition Manager will exit and a shut down for reconfiguration should be done.

4. Management Tools

This section provides an overview of Partition Manager and the partition commands.

4.1 Partition Manager

Partition manager is a graphical tool that allows administrators to make changes to the complex. It also displays system configuration and hardware information. There are four ways to invoke Partition Manager.

- Through SAM (the System Administration Manager tool included with HP-UX).
- Typing the Partition Manager command, parmgr(1M), at an HP-UX shell prompt.
- Via a web browser running on a PC.
- Selected tasks in Partition Manager can be accessed directly using the -t option of the Partition Manager command.

4.1.1 Running Partition Manager from a Web Browser Running on a PC

In order to run Partition Manager through a PC web browser (both Internet Explorer and Netscape are supported), the administrator must first run the built-in web server on a partition in the complex to be managed. The command to start the web server is:

/usr/obam/server/bin/apachectl start

Additional information about configuring the web server and browser requirements can be found in Partition Manager's online help.

To get to the Partition Manager launch page use the URL shown below with "partition-hostname" replaced with the actual **hostname** (**not** the partition name) of the partition running the web server.

http://partition-hostname:1188/parmgr/

Note: Accessing Partition Manager via a web browser is only supported for web browsers running on a PC.

When this URL is entered the following window is displayed.

Before running partition manager from a web browser, the Java Plug-In (JPI) needs to be installed on the PC. Complete directions and links for installing the Java software are available by selecting the [Configure Browser] button on the Partition Manager web page (the middle button in the screen shot above).
A login screen appears when the [Run Partition Manager] button is selected, thus a user must be able to log in as superuser in order to run partition manager. For each session of the web browser, the first invocation of Partition Manager may take upwards of 30 seconds for the log in screen to appear. This is due to the time it can take to download the necessary Java elements to the PC. After the first time, the start up time is negligible. This is because the web browser caches the downloaded JAVA elements.

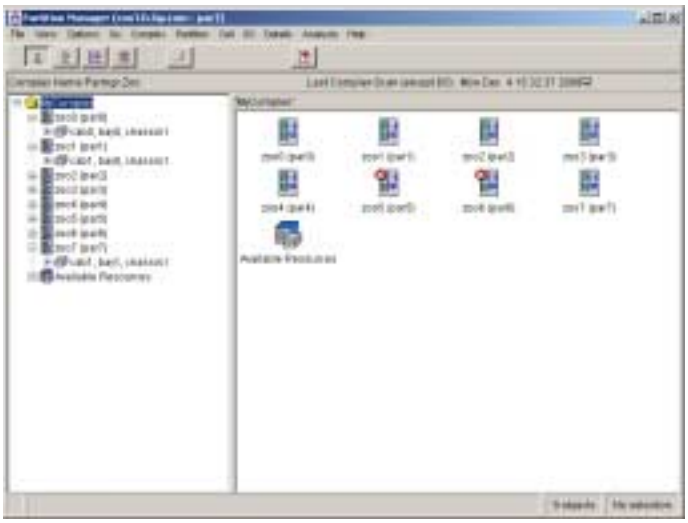
4.1.2 Using the Partition Manager Command

To run the main Partition Manager tool from an HP-UX shell prompt, logon to a partition as superuser and type the following commands:

```
export DISPLAY=<your display>:0.0   # set the display
/opt/parmgr/bin/parmgr &            # run Partition Manager
```

4.1.3 Sample Partition Manager Windows

The following screen shot shows Partition Manager's main screen.



In this view large icons are used to represent the objects shown on the right pane. Note the X in the circle on two of the icons. This indicates that these partitions have hardware (a cell or chassis) that is not in the "normal" active state.

Most of the focus of Partition Manager is configuring partitions. However, it is also important to be able to get an overview of all of the hardware in the complex. This screen shot shows the Complex Details screen (use the "Complex Details" task in the **Details** menu).

Shown here is the General tab. The other tabs show:

- An overview of all of the cells,
- A more detailed view of the CPUs and memory in all cells,
- An overview of all I/O chassis,
- Information specific to the cabinets in the complex
- Power and cooling information (for example, under this tab it is easy to determine if all power and cooling systems are at N+).

This is just one example of Partition Manager's capability to provide detailed information. There is also detailed information about a selected partition, cell, or I/O chassis. All of these dialogs include a [print] button to make it easy to capture this information (all of the tabs in a given dialog are printed, Postscript format is used, though the output can be sent to a printer or a file). The "Complex Details" screen also includes a [Save…] button that saves the information in this dialog in ASCII form.

4.1.4 Launching Individual Tasks Partition Manager also supports accessing certain individual tasks directly. For example:

```
# Start the "Create Partition" task
/opt/parmgr/bin/parmgr –t create

# Display the "Complex Details" property sheet
/opt/parmgr/bin/parmgr –t complex_details
```

The screen for the specified task will show up instead of the main Partition Manager screen. The full set of tasks that can be accessed in this manner are listed in the parmgr(1M) man page

4.2 Commands

The partition commands include the following.

Command	Description
parcreate	Create a new partition; `root` permission is required. See the *parcreate*(1M) man pages for details.
parmodify	Modify an existing partition; `root` permission is required. See the *parmodify*(1M) man pages for details.
parremove	Remove an existing partition; `root` permission is required. See the *paremove*(1M) man pages for details.
parstatus	Display partition information and hardware details for a Superdome Complex. See the *parstatus*(1M) man pages for details.
parunlock	Unlock Complex Profile data (use this command with caution); `root` permission is required. See the *parunlock*(1M) man pages for details.
fruled	Turn the amber attention LEDs on or off for cells, cabinets, and I/O chassis. See the *fruled* (1M) man page for details.
frupower	Turn power on or off for cells and I/O chassis; `root` permission is required. See the *frupower* (1M) man page for details.

A. Appendix

A.1 References

A.1.1 Books:

- Managing Superdome Complexes: A Guide for HP-UX System Administrators, Part # B2355-90702
- Managing Systems and Workgroups: A Guide for HP-UX System Administrators
- Superdome Management Part I: Superdome Architecture and Service Processor-Based Mgmt, Interworks 2001.

A.1.2 HP-UX Man Pages

- Commands: fruled(1), frupower(1M), parcreate(1M), parmodify(1M), parremove(1M), parstatus(1), partition(1), parunlock(1M)
- Partition Manager: parmgr(1M)

A.2 Other Superdome Information Links

Superdome web site:
`http://www.unixservers.hp.com/highend/superdome`

Documentation (including the Managing Superdome Complexes manual) is available at:
`http://www.docs.hp.com/`

Education:
`http://education.itresourcecenter.hp.com/`

A.3 Trouble Shooting Tips

A.3.1 The Partition is in the Wrong Shutdown/Reboot State

If an administrator mistakenly does a normal shut down instead of a shut down for reconfiguration, the easy way to get the partition in the right state is to logon to the GSP, go to the Command Menu, and issue the

Reset for Reconfiguration (RR) command. This command results in setting BIB for the partition's cells and commencing POST on each cell.

Caution: Be careful about using the RR command. The GSP doesn't know what the OS is doing so an RR on a partition with a normally running OS will immediately crash the OS when the cell reset is done.

A.3.2 How to Force an Unlock of the Complex Profile

If for some reason, a profile was not unlocked by the command that locked it, there is a need to clean up the lock by doing a forced unlock. Either parunlock(1M) or the GSP Rekey Complex Profile Lock (RL)command will perform this task. For example, using parunlock(1M):

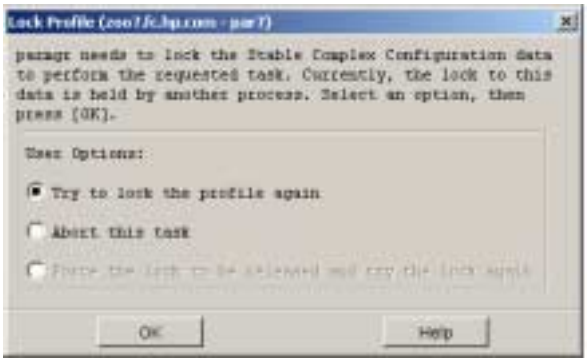
```
parunlock –p 0    # unlock the PCD for partition 0
parunlock –s      # unlock the SCCD
parunlock –A      # unlock ALL the profile parts.
```

Caution: Before using parunlock(1M), find out if a forced unlock is really needed. There might be a task that has a legitimate need for the lock.

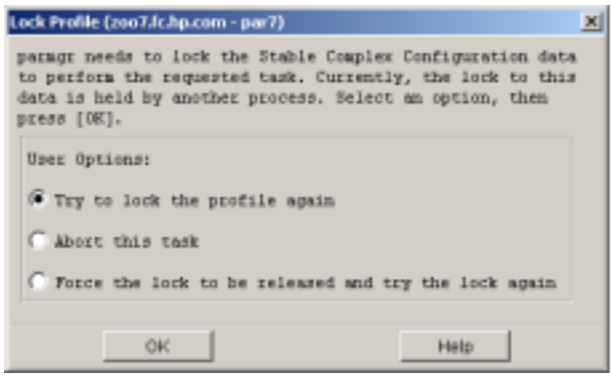
A.3.3 Locking and Unlocking the Complex Profile

The tool or command that is used to modify the complex configuration handles the locking and unlocking. Lock contention happens when more than one process is trying to lock the same portion of the profile. An example would be for an administrator to start up a configuration task that locks the SCCD portion of the profile, but is called away before completing the task. Then another administrator tries to make changes that also require locking the SCCD. Another example is if the administrator is performing a task that requires a long time to complete.

The following screen shot shows the dialog displayed when Partition Manager attempts to lock the SCCD when it is already locked by another process.



The administrator can abort the task and try later, or the administrator can try again. For example, find out who else is making a change, and either have them complete the task or cancel the other task. If the "try again" option is selected and the SCCD is still locked then the third option of forcing an unlock becomes available.



Use the "force unlock" option with care. It could interfere with a task someone else is doing. It is possible that the lock is stale, i.e., the tool that acquired the lock abnormally terminated and did not release the lock. If the administrator is aware of such a case then using the force unlock option is one way to clear the stale lock. Another way is to use the parunlock(1M) command, and a third way is to use the Rekey Complex Profile Lock (RL) command in the GSP's command menu.

If the force unlock option is used and, for example, there was another instance of Partition Manager running, then when the user of that Partition Manager selects [OK], the task will fail because that Partition Manager no longer has the profile locked. That user will then be forced to repeat the task that they were trying to perform.

Note that the same thing happens if Partition Manager attempts to lock a partition's PCD and it is already locked. The only difference in the dialog is that the text at the top of the dialog identifies the PCD that needs to be locked.

A.3.4 Complex Reconfiguration

One of the key rules for changing the partition assignment of cells is that the cell must be inactive. However, it is possible for the management tools to give a modified SCCD to the GSP that changes the assignment of active cells. When this happens the GSP waits until those cells become inactive before making the new SCCD the current SCCD. This process of waiting for cells to become inactive before pushing out a new SCCD is called a **complex reconfiguration.** While in this state no further changes can be made to the SCCD.

Partition Manager checks for this condition each time it is run. If this condition exists then Partition Manager displays the following message:

Note that Partition Manager has no way of finding out what changes are pending, even if the changes were made by an instance of Partition Manager (the changes also could have been something done from another partition or via one of the partition commands).

Once [OK] has been selected Partition Manager will display its main window, but the data displayed reflects the current SCCD, not the pending SCCD. Also, Partition Manager will be unable to perform any task that requires a change to the SCCD. This is because Partition Manager will not be able to lock the SCCD until the pending changes have been completed.

The best thing to do in this case is to exit Partition Manager and find out what is keeping the pending SCCD from being pushed out. This is invariably related to one or more cells that need to be inactive but aren't. The most common cause of this is a partition that has not been shut down or rebooted properly (i.e., a shut down or reboot for reconfiguration was not done). Use the Virtual Front Panel from the GSP to get detailed information about the status of each partition in the complex.