

# Getting started with HP-UX Workload Manager (basic configuration and monitoring)



Introduction.....	2
Audience.....	2
What is HP-UX WLM?.....	2
Common WLM configurations.....	3
Providing a fixed amount of CPU.....	4
Providing CPU as it is needed.....	11
Providing CPU for a given time period.....	21
Automatically resizing PSETs.....	29
Automatically resizing vPars.....	36
Automatically resizing nPars using iCOD software.....	37
Ensuring CPU allocations after a Serviceguard failover.....	37
Isolating Oracle database instances.....	38
Example WLM configurations.....	46
Alternative methods for placing processes in workload groups.....	47
pmmrun: starting a process in a workload group.....	47
pmmmove: moving an existing process to a workload group.....	47
Monitoring wlm.....	48
ps [-P] [-R workload_group].....	48
wlminfo.....	48
GlancePlus.....	49
Status and statistics logs.....	50
Stopping WLM.....	50
For more information.....	50

# Introduction

Today, we live in a constantly changing world where IT departments need to increase their agility while improving their return on IT investment. The Adaptive Enterprise is HP's vision for helping customers synchronize business and IT to capitalize on change. Virtualization is a cornerstone of HP's approach to helping customers realize the promise of becoming an adaptive enterprise. It is an approach to IT that pools and shares resources so utilization is optimized and supply automatically meets demand.

Part of the virtualization model is the Virtual Server Environment. The HP Virtual Server Environment has the broadest range of server virtualization capabilities, including clustering, both for high availability and high performance, on-demand technologies such as iCOD and Pay Per Use, partitioning, resource management, and more. The HP Virtual Server Environment for HP-UX also offers an intelligent policy engine that ties all these technologies together, HP-UX Workload Manager (WLM). WLM, the only goal-based workload management product available in the UNIX environment, advises the administrator to reallocate or automatically reallocates system resources based on service-level objectives (SLOs) such as application performance, time of day, CPU utilization, and more.

## Audience

This paper is intended for individuals who are interested in using WLM but would prefer to learn as little of the WLM configuration file syntax as possible.

The paper explains how to configure and monitor HP-UX WLM Version A.02.02 running on HP-UX 11.0, HP-UX 11i V1.0 (B.11.11), or HP-UX 11i V2.0 (B.11.23).

## What is HP-UX WLM?

WLM is an optional HP product (available as a standalone product and in the mission-critical operating environment, or OE) that provides automatic resource allocation and application performance management through the use of prioritized service-level objectives (SLOs).

When using WLM, the system administrator:

- Defines workload groups for each workload
- Places the applications or users that define the workloads into their own workload groups (the workload's processes share the resources WLM makes available to the workload group)
- Creates one or more SLOs for each workload group

In defining an SLO, the system administrator specifies a priority and optionally a metric goal or a resource usage goal. As the applications run, WLM compares the application metrics or usage against the goals. WLM then automatically adjusts the amount of CPU available to the workload groups to achieve each goal, based on available CPU and SLO priority.

Thus, WLM is most effective in managing applications that are CPU-bound: It adjusts the CPU allocation of a workload group based on the performance of applications in that workload group. WLM can also manage real memory, although not in response to SLO performance. Disk bandwidth allocations are statically assigned in the WLM configuration file. If multiple users or applications within a workload group are competing for resources, standard HP-UX resource management determines the resource allocation.

You can use WLM with a number of other HP products to better manage your systems:

- **Virtual Partitions (vPars):** You can run WLM within and across virtual partitions. For systems with partitions, WLM provides a global arbiter that can take input from the WLM instances on the individual partitions. The global arbiter then moves CPUs between partitions, if needed, to better achieve the SLOs specified in the WLM configuration files that are active in the partitions.
- **Node Partitions (nPars):** You can also run WLM within and across node partitions. For nPars with iCOD software installed, WLM can simulate CPU movement between the partitions, activating CPUs on the partitions where CPU resources are most needed.
- **Processor Sets (PSETs):** You can base a workload group on a PSET, dedicating the processors to the applications that make up the workload group. In addition to simply setting up the PSET-based workload group and not touching it, WLM can add and remove processors to a PSET-based group based on your SLOs.
- **Instant Capacity on Demand (iCOD):** WLM can inform you that iCOD resources could help reduce the number of SLO failures, allowing you to manually activate an iCOD processor if appropriate. Alternatively, you can configure WLM to automatically activate and deactivate processors using iCOD software.
- **Pay Per Use (PPU):** Pay Per Use (PPU) version 4.x uses reserve CPUs to meet computing needs, with costs based on the number of active processors. Using WLM in such an environment, you can have the system's CPU capacity increased or decreased automatically: After you set your SLOs in the WLM configuration file, WLM and its Pay Per Use Toolkit (PPUTK) automatically optimize the number of active CPUs to the minimum number of CPUs needed to satisfy the SLOs. With PPU, you pay only for the amount of CPU you use; thus, minimizing the number of active CPUs minimizes your costs.

## Common WLM configurations

WLM is controlled by a configuration you define. You can create this configuration in a text editor, through the WLM GUI, or through the WLM configuration wizard. This paper focuses on configuring WLM using the wizard.

This paper covers or points to additional information on the following types of WLM configurations:

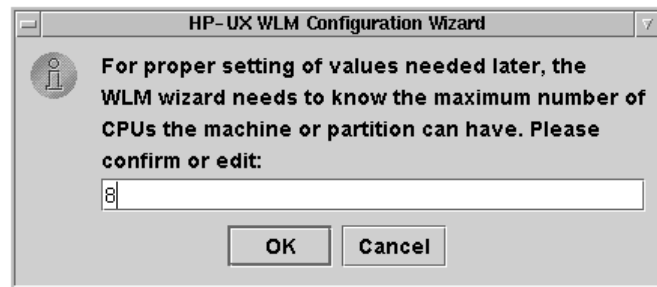
- Providing a fixed amount of CPU
- Providing CPU as it is needed
- Providing CPU for a given time period
- Automatically resizing PSETs
- Automatically resizing vPars
- Automatically resizing nPars using iCOD software
- Ensuring CPU allocations after a Serviceguard failover
- Isolating Oracle database instances

## Providing a fixed amount of CPU

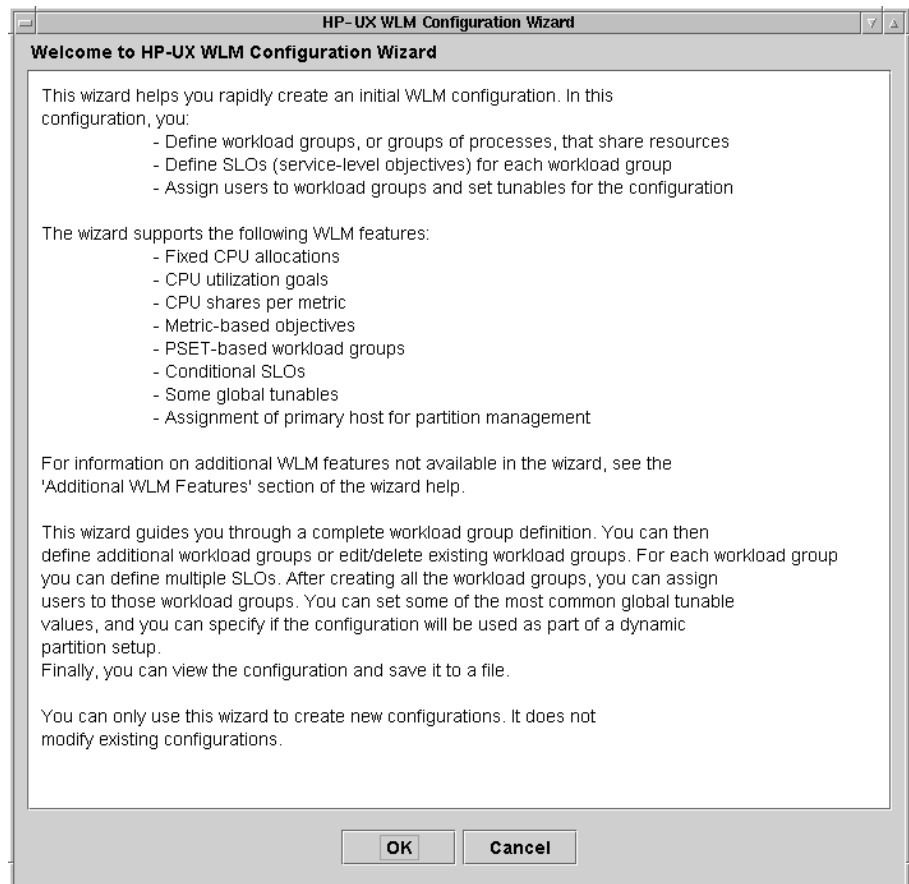
WLM allows you to give a workload group a fixed amount of CPU, reserving that amount of CPU for your workload group.

To set up a fixed CPU allocation using the configuration wizard:

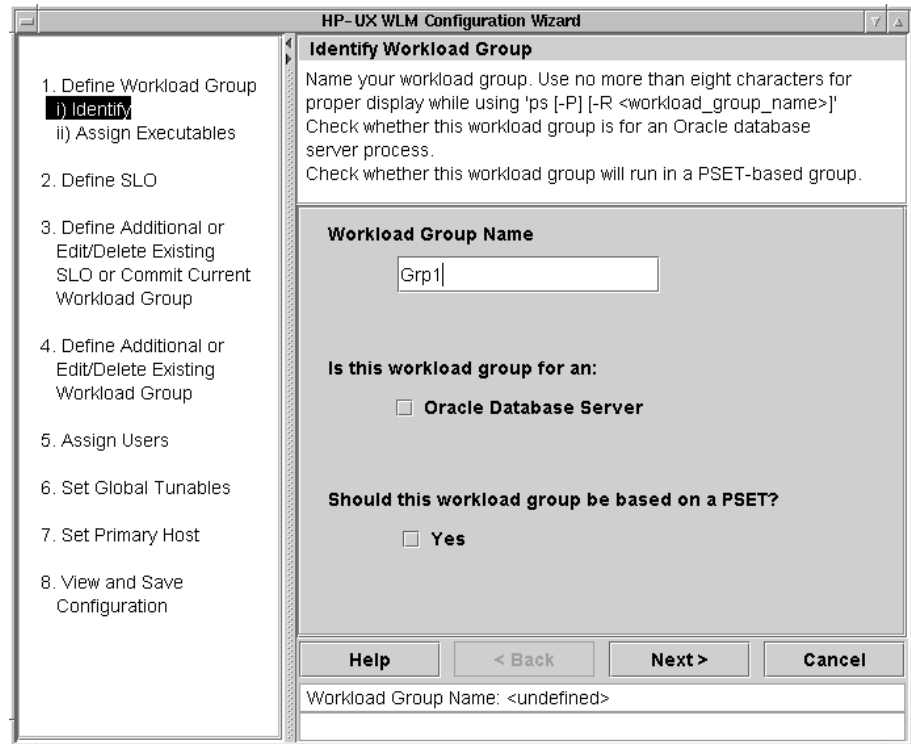
1. Log in as root.
2. Be sure your DISPLAY environment variable is set, then start the wizard:  
# /opt/wlm/bin/wlmcw
3. Enter the maximum number of CPUs the system will have.  
The wizard guesses at the maximum value. Confirm or edit the guess. The value you enter should take into account changes in the CPU count due to iCOD and other such adjustments to the CPU count.



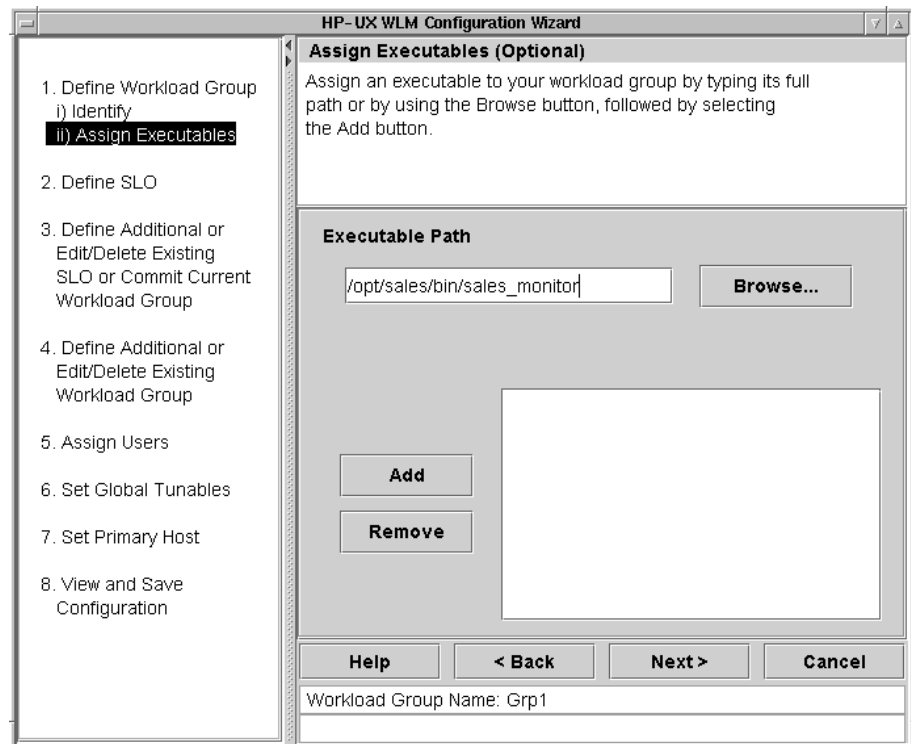
4. Read the introduction screen and press the **OK** button.



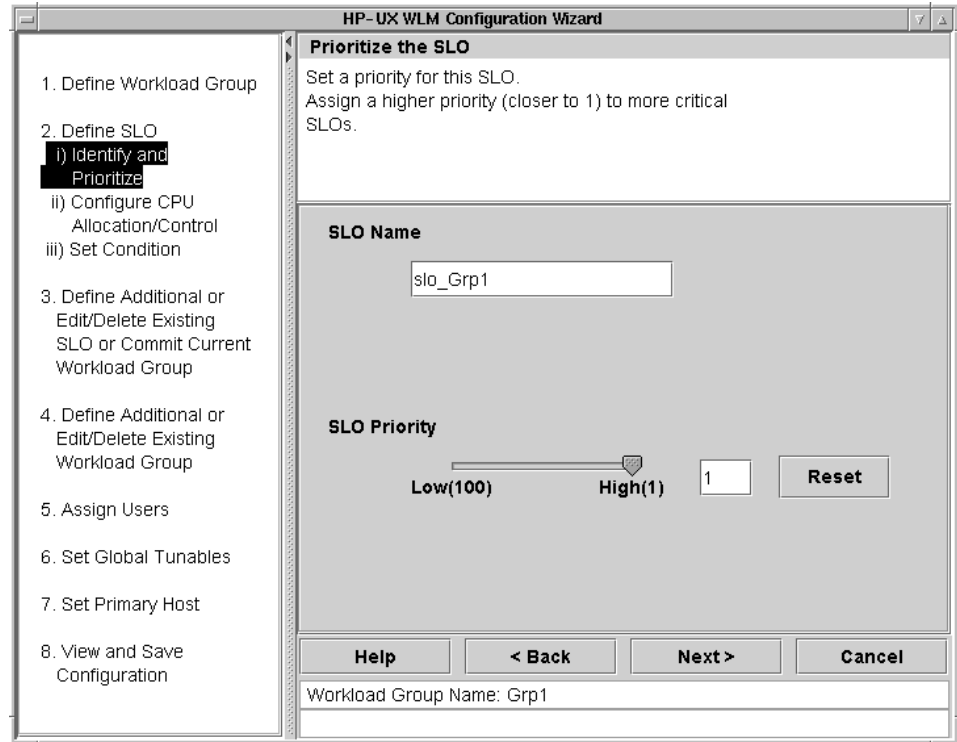
5. Name the workload group and press the **Next** button. In the screen below, the group is named Grp1.



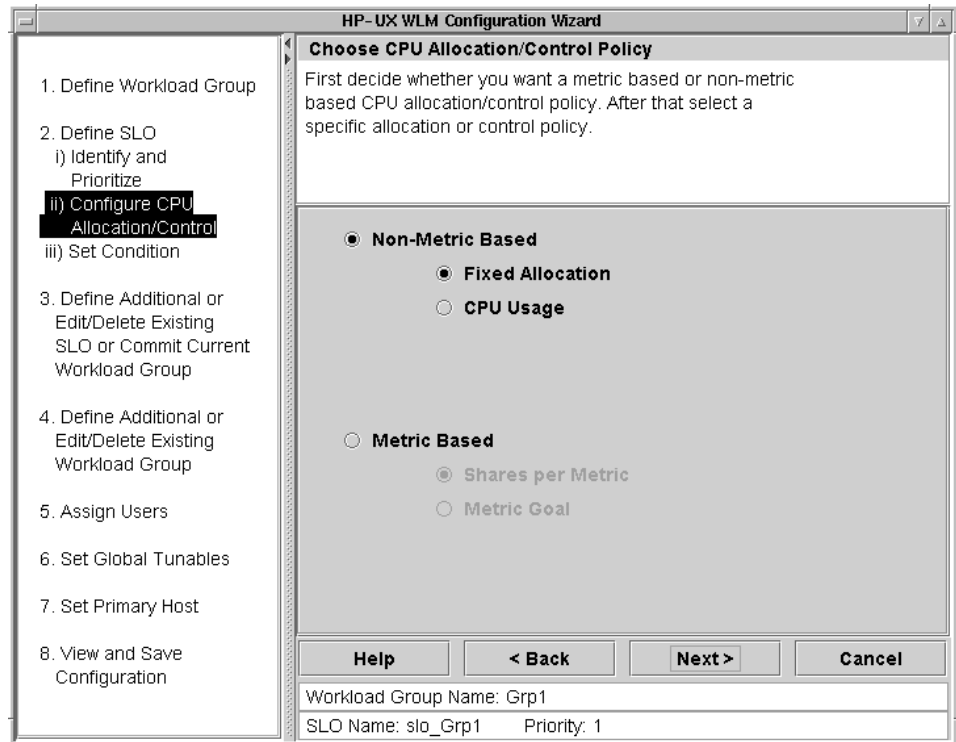
6. Assign a workload to the workload group you defined in the previous step. Specify the **Executable Path** for an application that you want in the workload group and select the **Add** button. Add more applications to the workload group in the same fashion. Select the **Next** button when ready to continue.



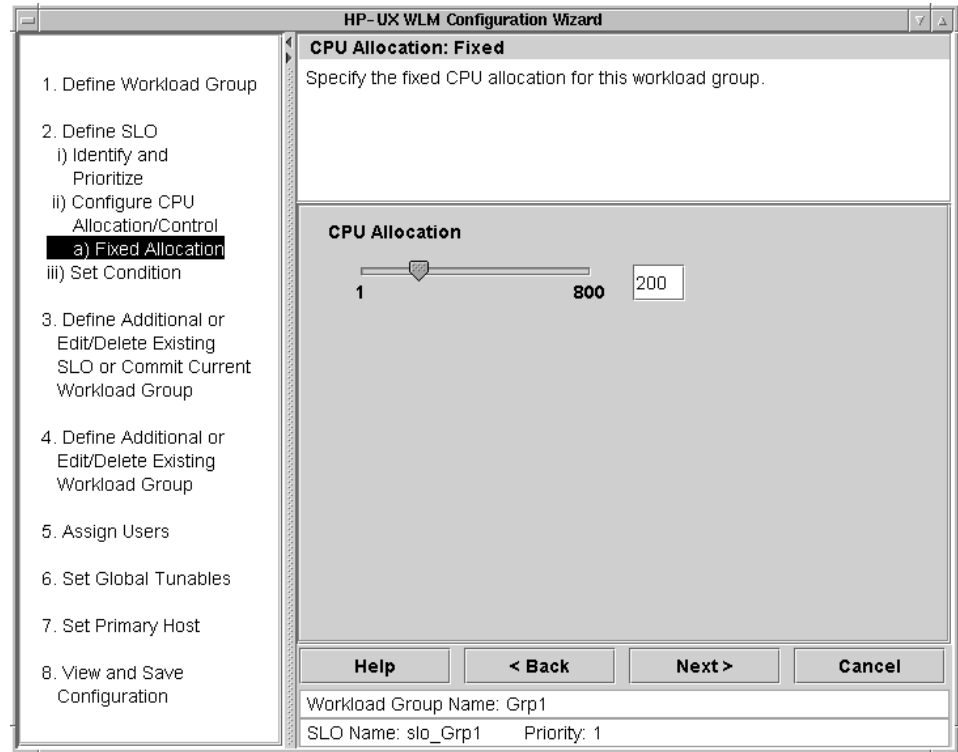
- Now enter data about the service-level objective (SLO), or goal, for the workload. Specify an **SLO Name** and assign an **SLO Priority** by moving the slider or entering a value (between 1 and 100) in the text field. Each SLO needs a priority so that WLM can choose which SLOs get CPU when there is not enough for all the workloads. The highest priority is 1; the lowest is 100. Select the **Next** button to continue.



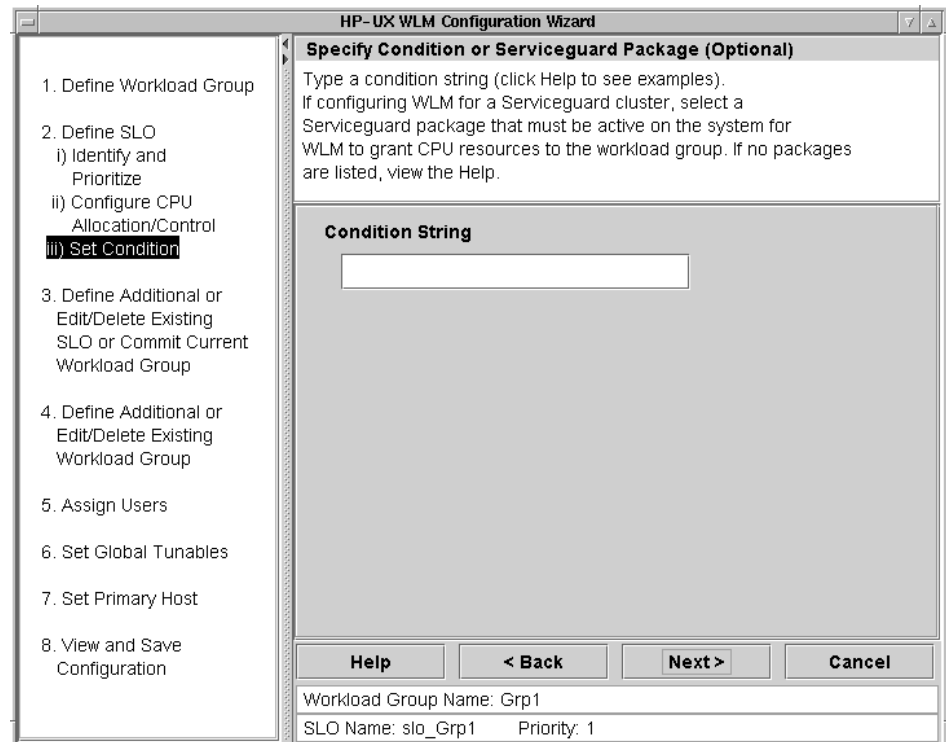
- Choose a CPU allocation that is fixed by selecting the **Fixed Allocation** option. Then select the **Next** button to continue.



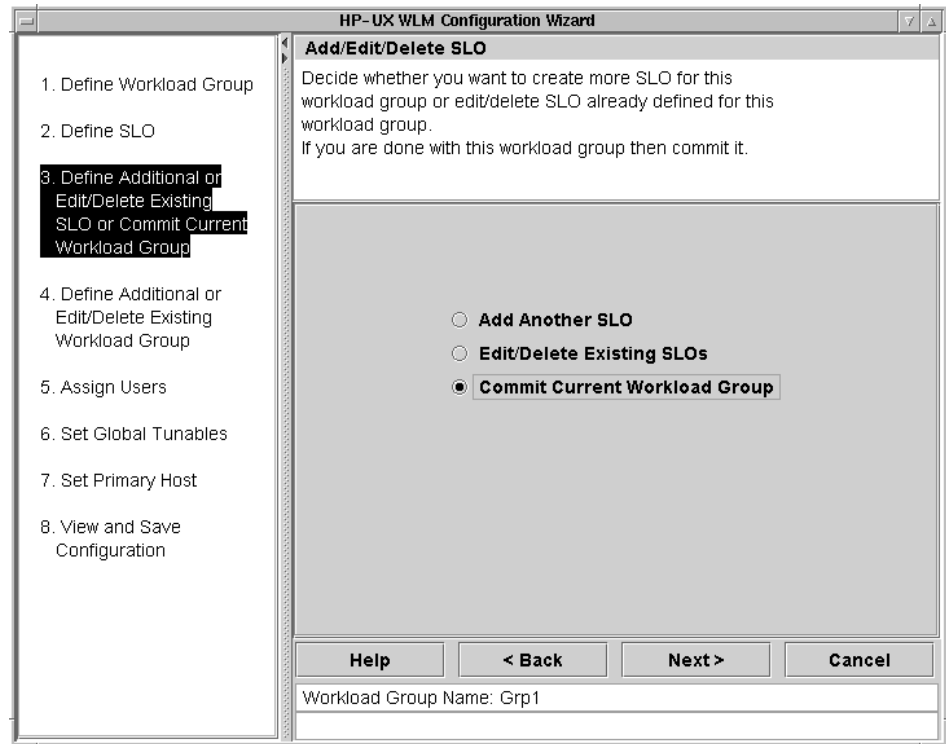
9. Specify the fixed amount of CPU the workload will receive using the slider or by typing in the text field. In this example, the workload gets 200 CPU shares or two CPUs. Select the **Next** button to continue.



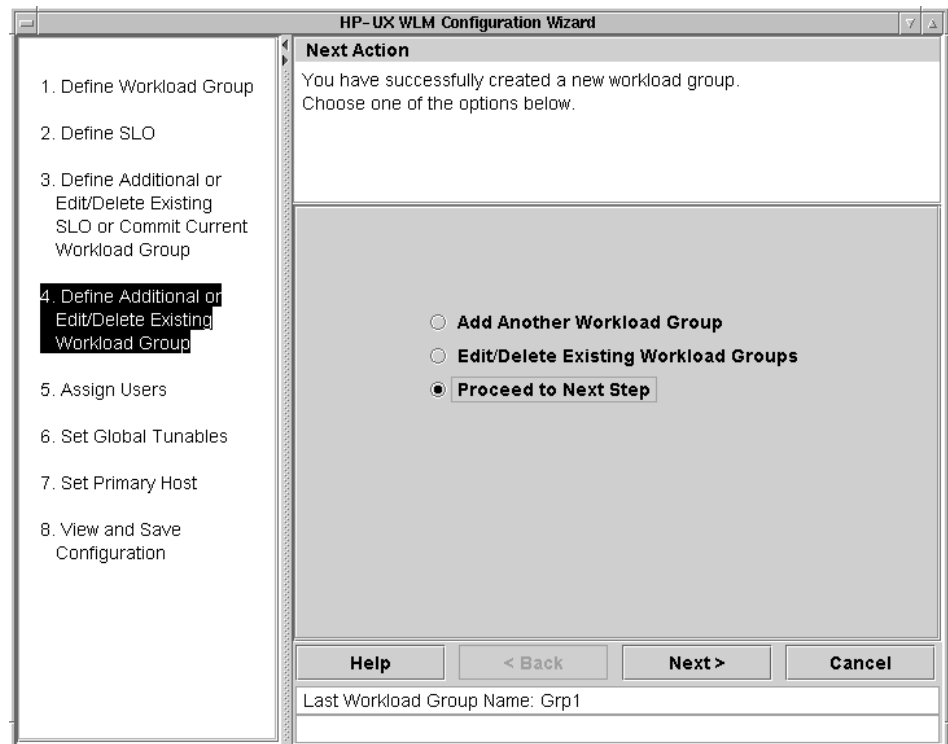
10. This screen allows us to enter a **Condition String** that indicates a condition that must be met for the SLO to be active. We want the SLO to be active all the time, so we enter nothing on this screen and simply select the **Next** button.



11. This screen allows us to add more SLOs for the workload or edit the existing SLO. We will use only the one SLO we just defined. Select the **Commit Current Workload Group** option then the **Next** button.

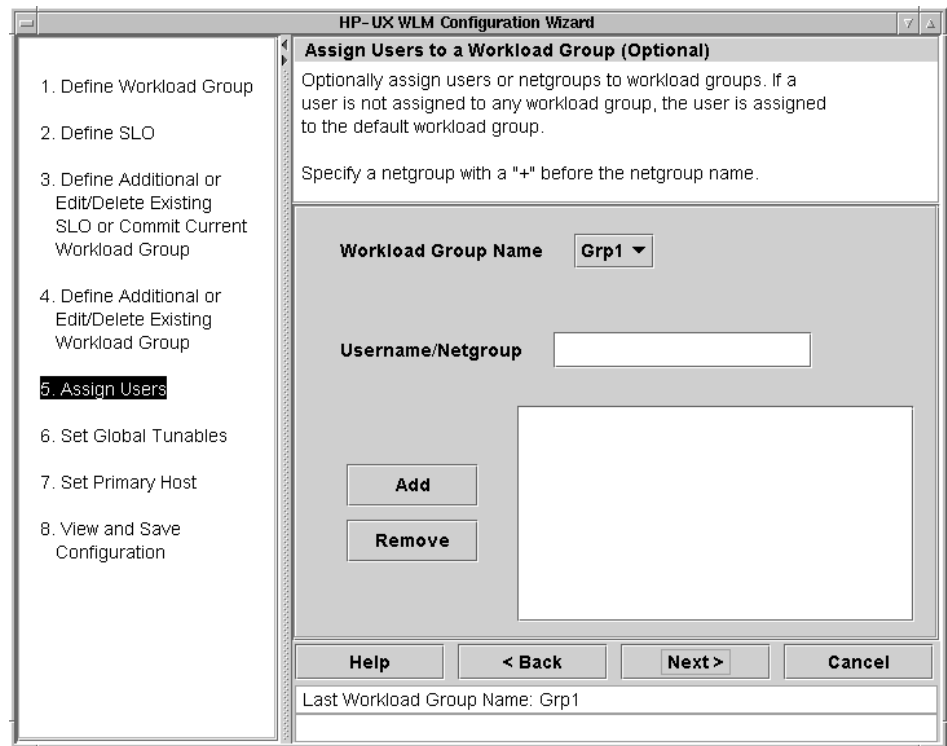


12. You have fixed the CPU allocation for a single workload. You can now continue to add workload groups (using a fixed allocation again or trying other types of allocations), edit the workload group we just defined, or continue with the WLM configuration. For brevity, we will select **Proceed to Next Step**, followed by the **Next** button.

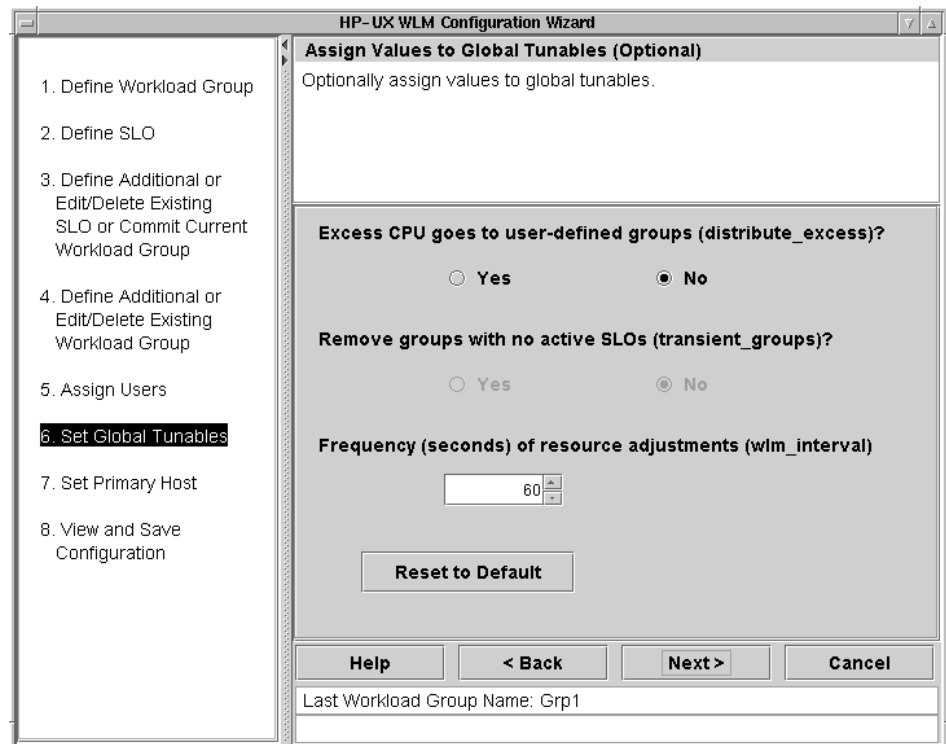




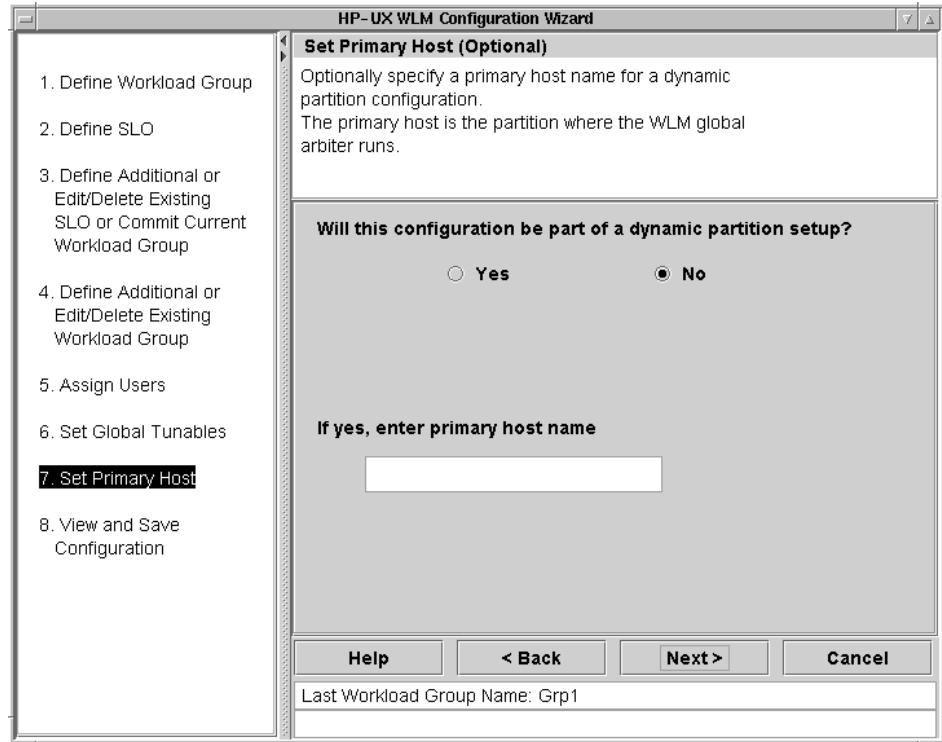
13. Assign users to the workloads if desired. User assignments ensure all the processes of a given user start in the specified workload. We are not concerned with user assignments, so skip these assignments by simply selecting the **Next** button.



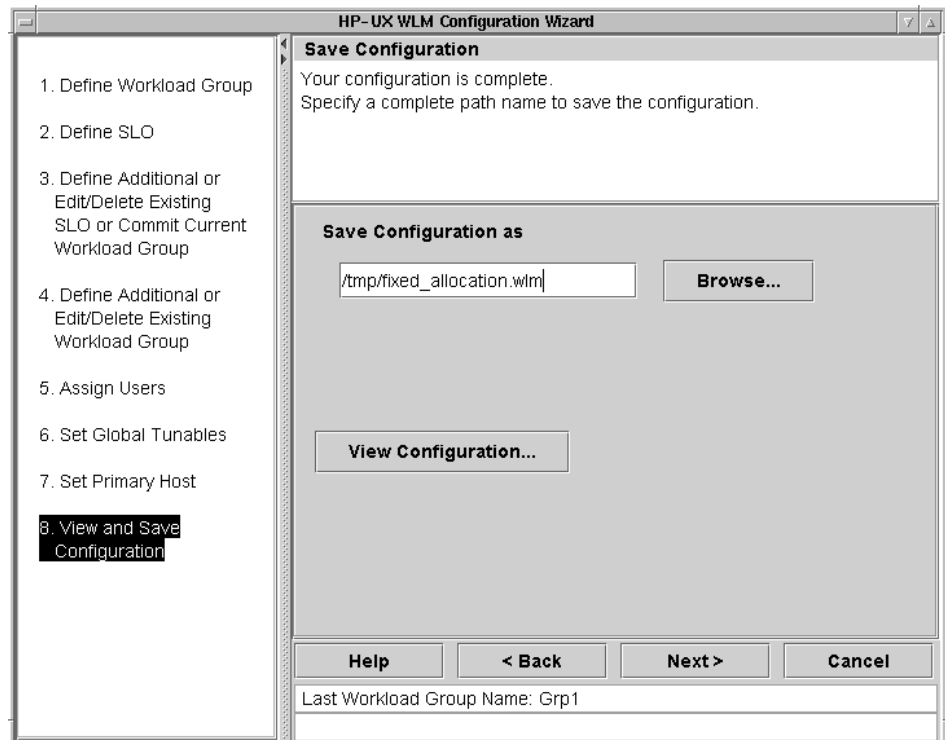
14. This screen allows you to set global tunables. Select the **Next** button to accept the defaults and continue.



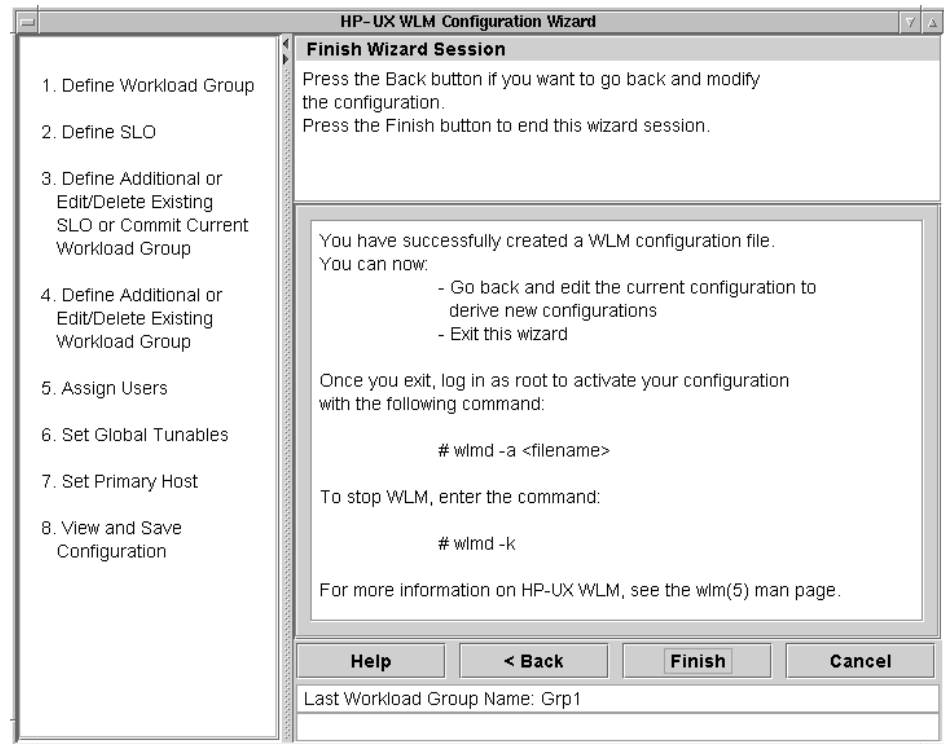
15. This next screen appears only if the system has HP-UX Virtual Partitions (vPars) or Node Partitions (nPars) installed. If you plan on managing vPars or nPars, see the section “Automatically resizing vPars” on page 36 or “Automatically resizing nPars using iCOD software” on page 37 for pointers to white papers that address partition management in more detail. Otherwise, select the **Next** button to continue.



16. Specify the full path and filename indicating where to save the configuration then select the **Next** button. In this example, the configuration will be saved as /tmp/fixed\_allocation.wlm.



17. Close the wizard by selecting the **Finish** button in the resulting screen.



18. Activate the new configuration file, instructing WLM to take control of resource management, using the following `wlmd` command:

```
# wlmd -a /tmp/fixed_allocation.wlm
```

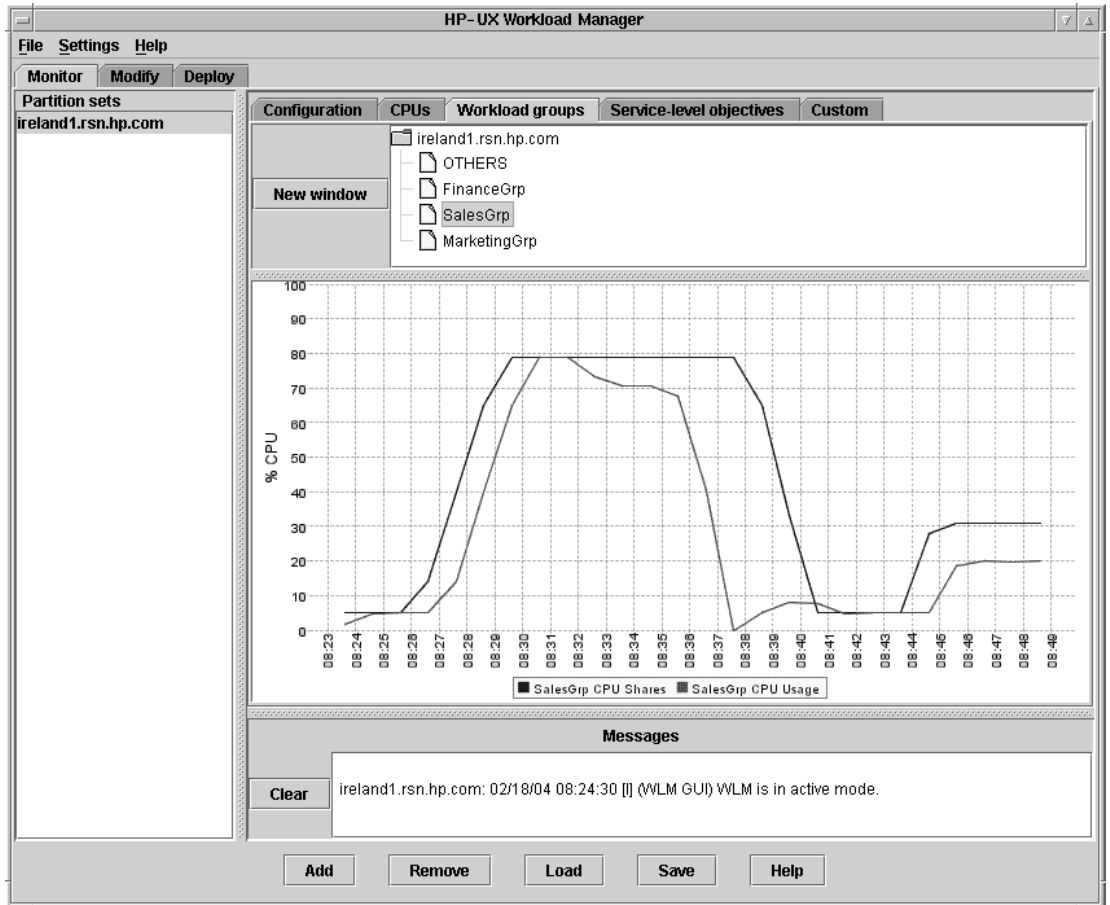
## Providing CPU as it is needed

To ensure a workload gets the CPU it needs—without preventing other workloads access to unused CPU resources—WLM allows you to define usage goals. Usage goals indicate how efficiently a workload should use its CPU allocation. As a result, a workload’s CPU allocation is reduced if it is not consuming enough of its current allocation, allowing other workloads to consume more CPU if needed. Similarly, if the workload is using a high percentage of its allocation, it is granted more CPU.

An advantage that a usage goal has over a fixed CPU allocation is that a workload group with a usage goal frees up CPU it isn’t using. A fixed allocation, however, reserves a constant amount of CPU for a workload regardless of its need for that CPU.

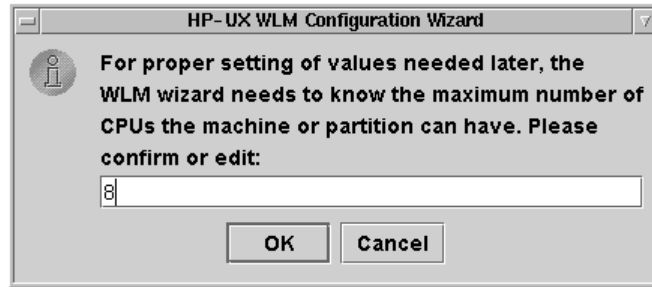
The following figure (produced using `wlminfo -i`) shows a workload, named `SalesGrp`, with an SLO that has a usage goal. WLM determines an allocation for the workload based on its usage. Once the workload is using more than 75% of its allocation, WLM gives it a larger allocation. Similarly, if the workload is not using at least 50% of its allocation, WLM gives it a smaller allocation. The top line in the graph (labeled `CPU Shares`) shows the workload's allocation. The bottom line shows usage. The allocation is capped at about 80 due to allocations to other groups.

Figure 1. Usage goal

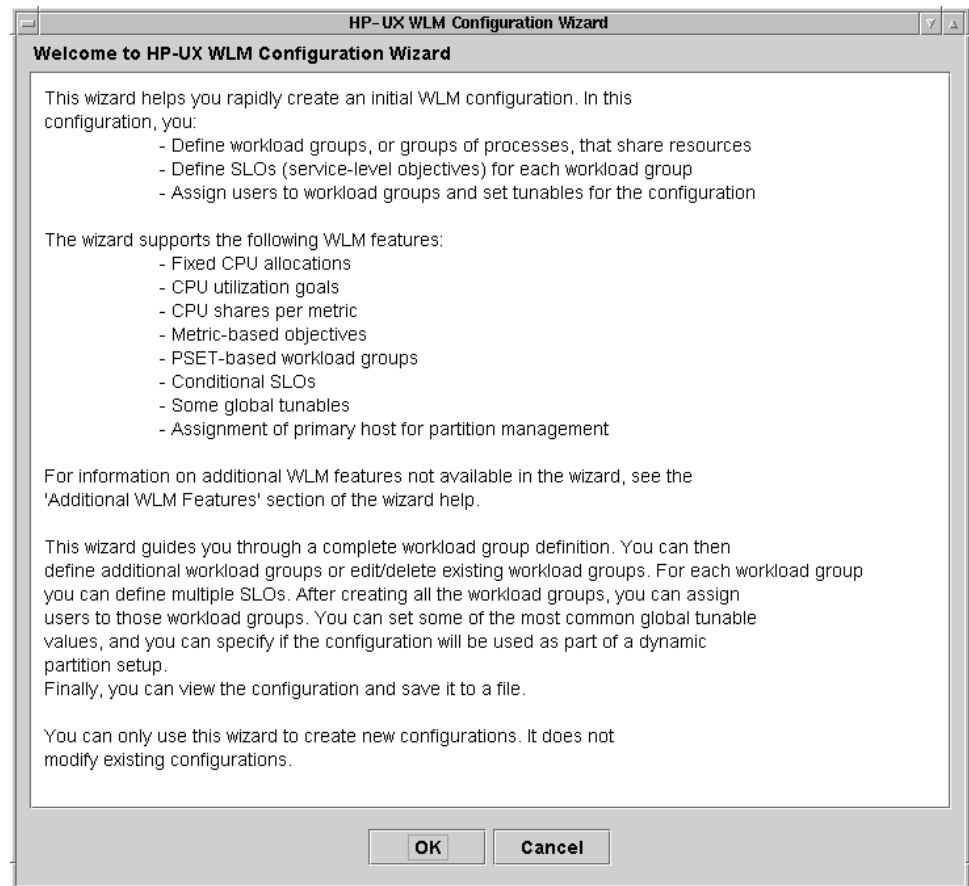


To set up usage-based CPU allocation:

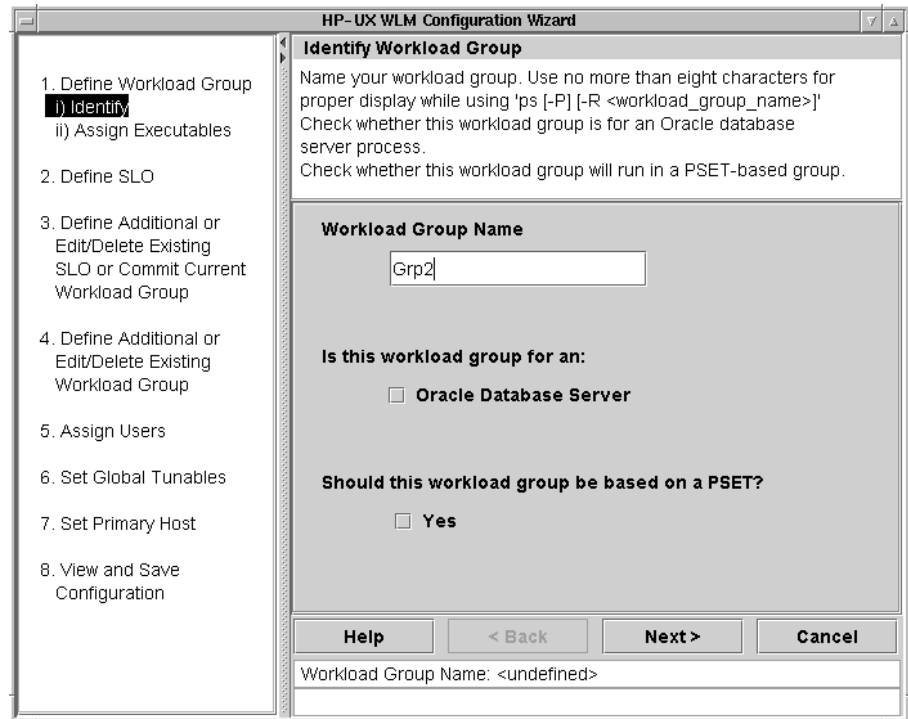
1. Log in as root.
2. Be sure your DISPLAY environment variable is set, then start the wizard:  
# /opt/wlm/bin/wlmcw
3. Enter the maximum number of CPUs the system will have.  
The wizard guesses at the maximum value. Confirm or edit the guess. The value you enter should take into account changes in the CPU count due to iCOD and other such adjustments to the CPU count.



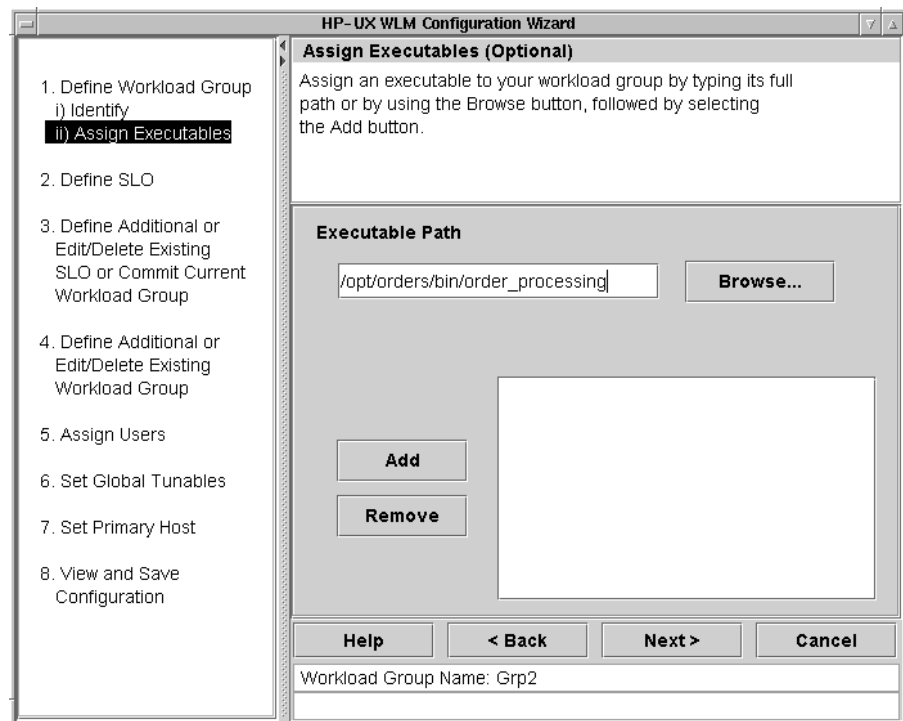
4. Read the introduction screen and press the **OK** button.



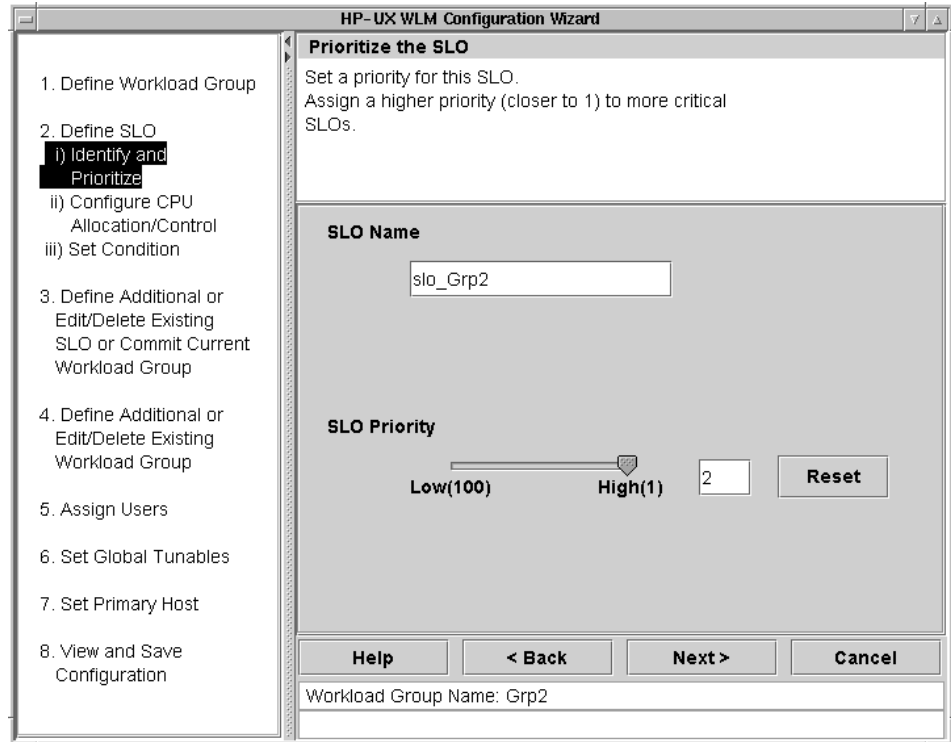
5. Name the workload group and press the **Next** button. In the screen below, the group is named Grp2.



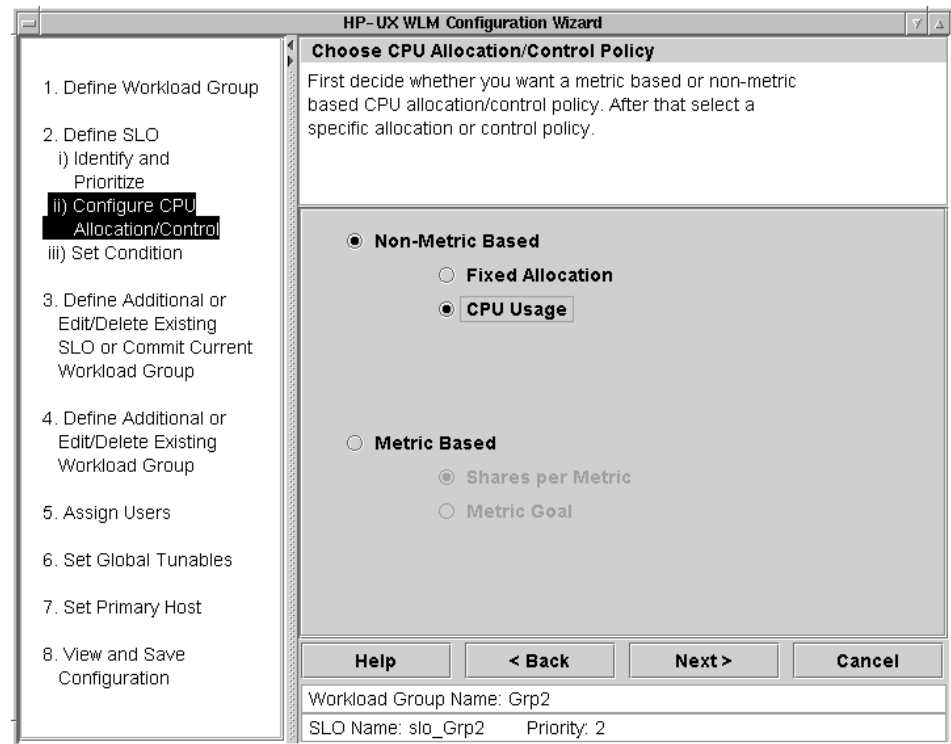
6. Assign a workload to the workload group you defined in the previous step. Specify the **Executable Path** for an application that you want in the workload group and select the **Add** button. Add more applications to the workload group in the same fashion. Select the **Next** button when ready to continue.



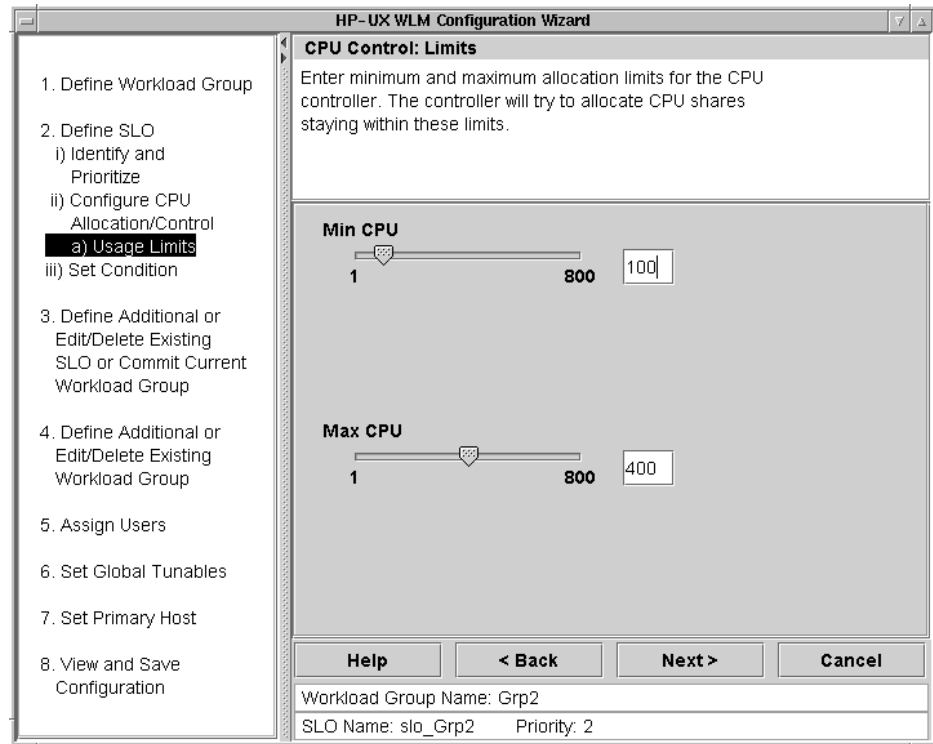
7. Now enter data about the service-level objective (SLO), or goal, for the workload. Specify an **SLO Name** and assign an **SLO Priority** by moving the slider or entering a value (between 1 and 100) in the text field. Each SLO needs a priority so that WLM can choose which SLOs get CPU when there is not enough for all the workloads. The highest priority is 1; the lowest is 100. Select the **Next** button to continue.



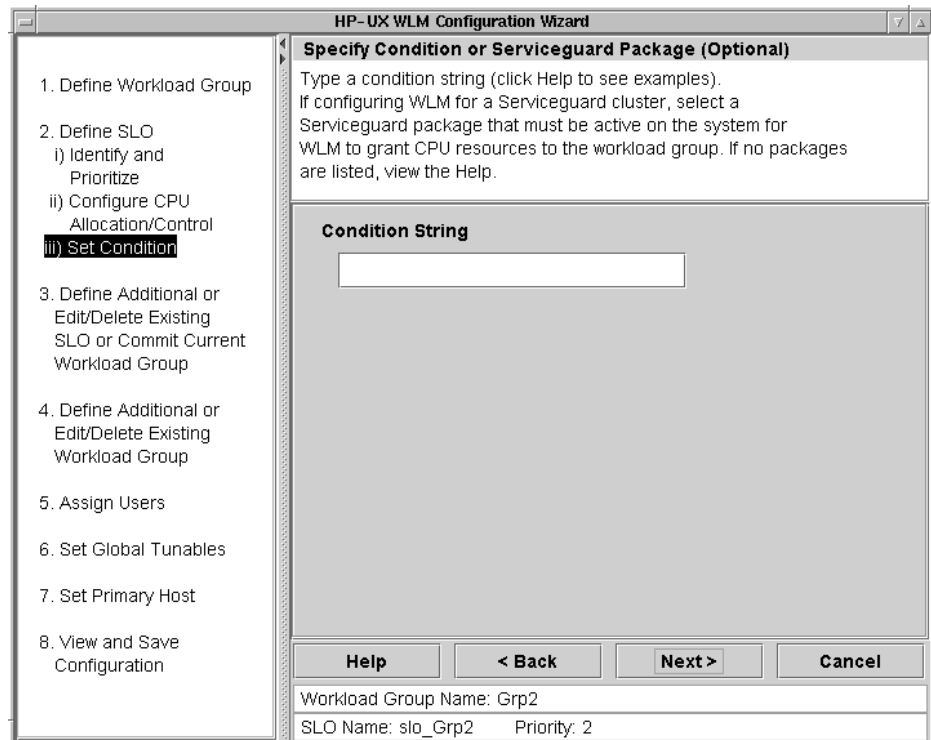
8. Set the CPU allocation to **CPU Usage** and select the **Next** button.



9. WLM automatically adjusts the CPU allocation for a workload when you have a usage-based allocation. You can request that WLM's adjustments fall within a minimum allocation and maximum allocation. Use the slider or input fields to set these bounds. Then, select the **Next** button. NOTE: The **Min CPU** and **Max CPU** values are merely requests; they are not hard limits. The requests below are for 100 and 400 CPU shares, meaning one CPU and four CPUs.

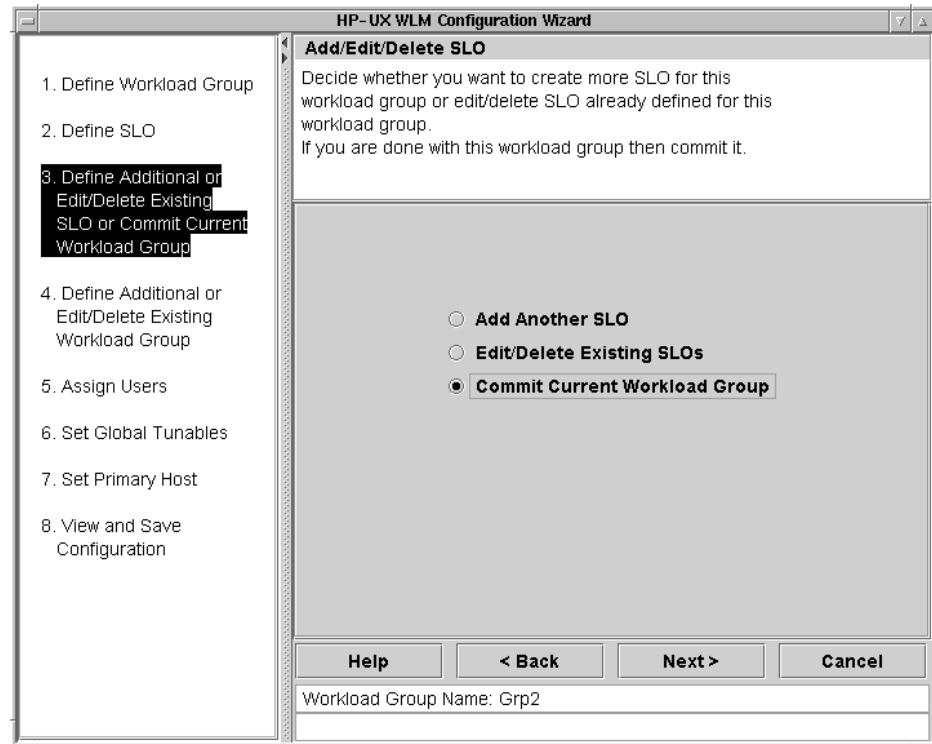


10. This screen allows us to enter a **Condition String** that indicates a condition that must be met for the SLO to be active. We want the SLO to be active all the time, so we enter nothing on this screen and simply select the **Next** button.

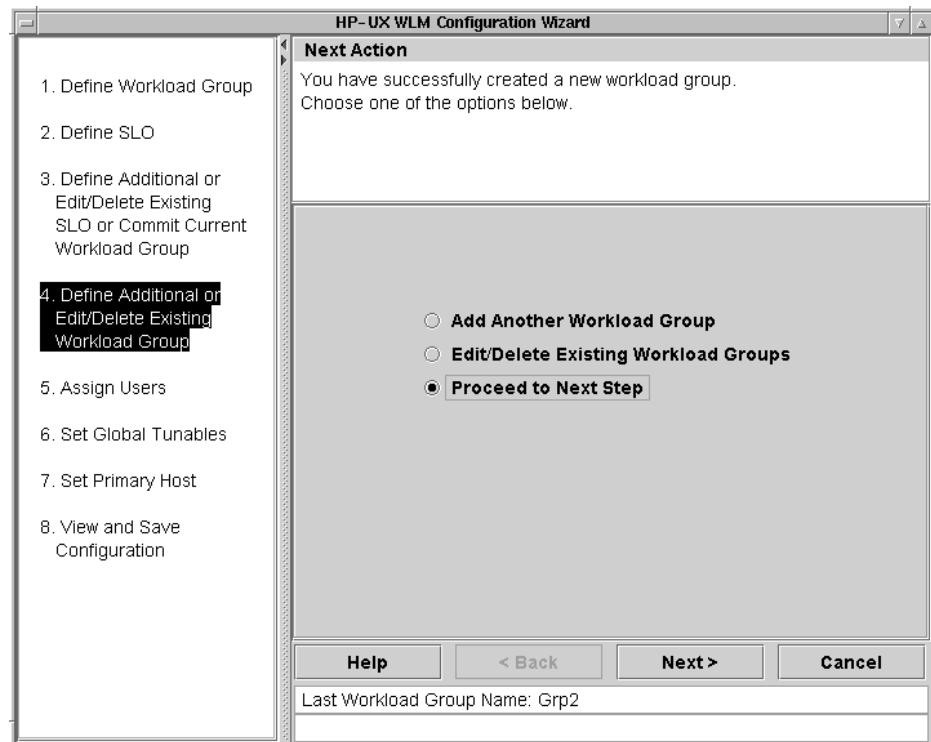




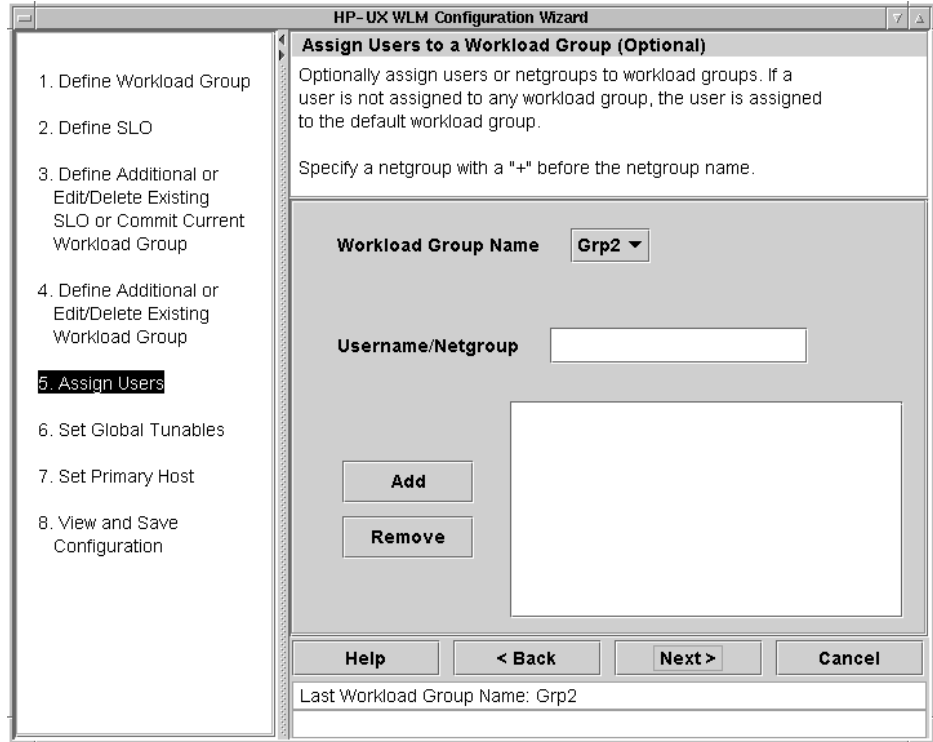
11. This screen allows us to add more SLOs for the workload or edit the existing SLO. We will use only the one SLO we just defined. Select the **Commit Current Workload Group** option then the **Next** button.



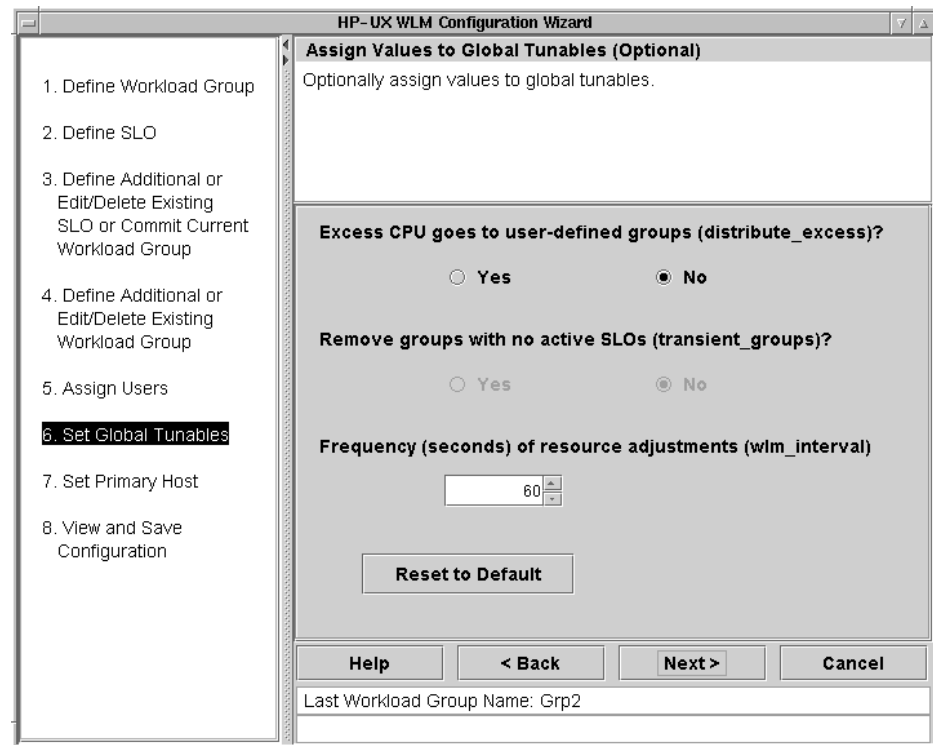
12. You have set up a CPU usage goal for a single workload. You can now continue to add workload groups (using a usage goal again or trying other types of allocations), edit the workload group we just defined, or continue with the WLM configuration. For brevity, we will select **Proceed to Next Step**, followed by the **Next** button.



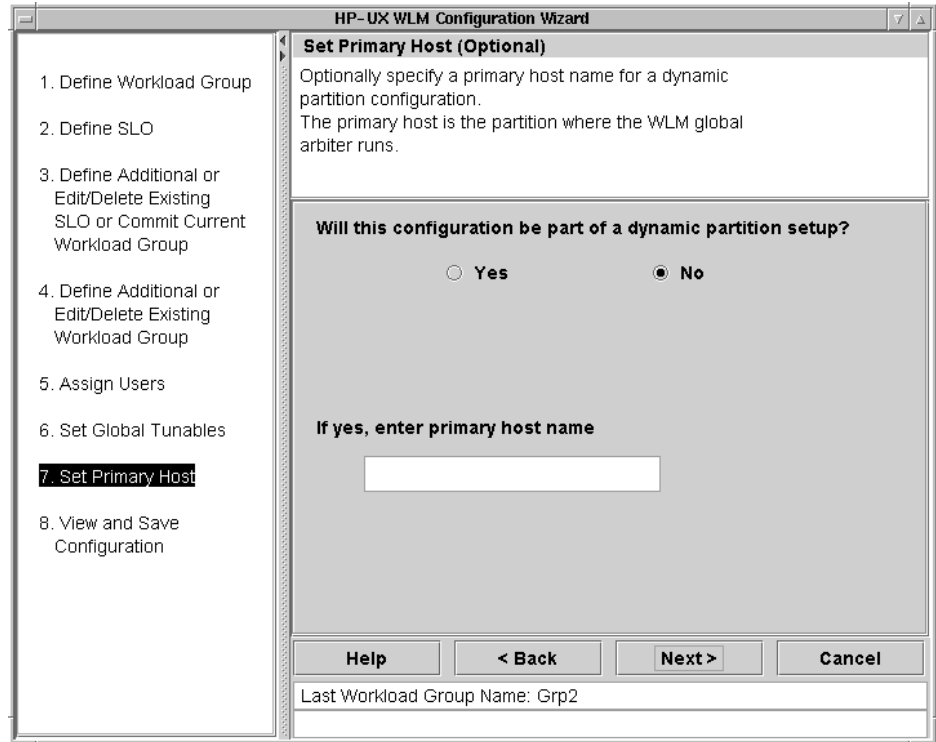
13. Assign users to the workloads if desired. User assignments ensure all the processes of a given user start in the specified workload. We are not concerned with user assignments, so skip these assignments by simply selecting the **Next** button.



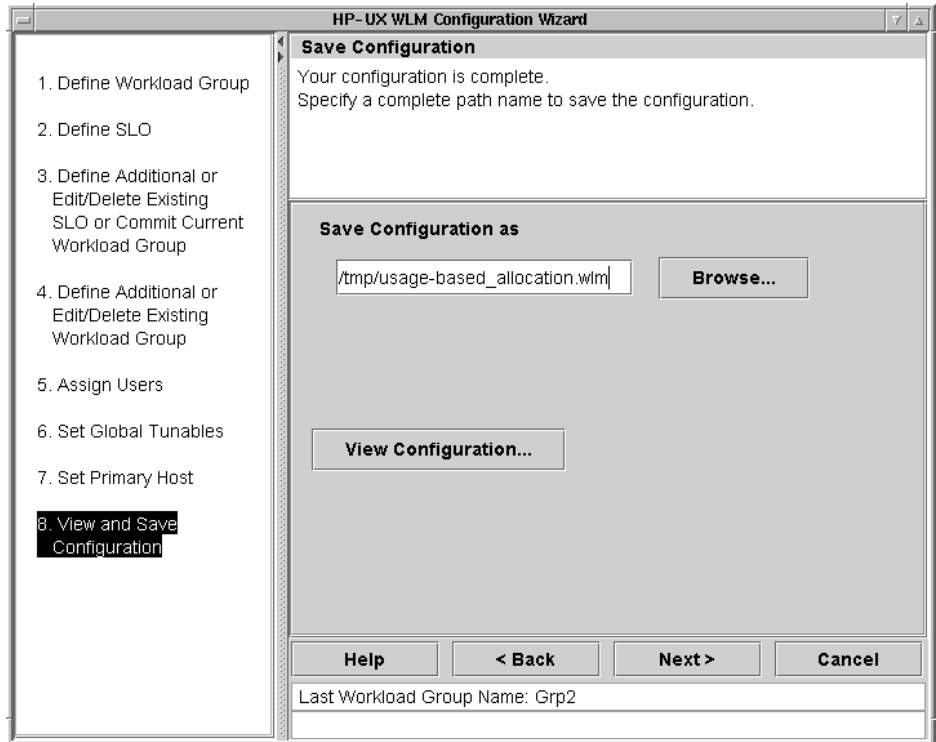
14. This screen allows you to set global tunables. Select the **Next** button to accept the defaults and continue.



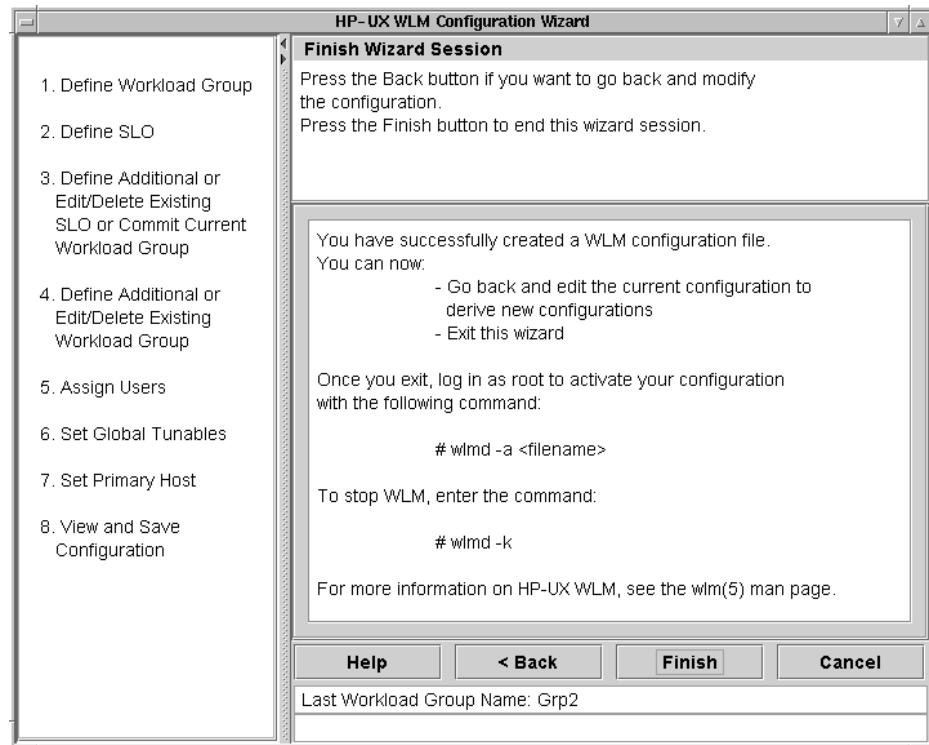
15. This next screen appears only if the system has HP-UX Virtual Partitions (vPars) or Node Partitions (nPars) installed. If you plan on managing vPars or nPars, see the section “Automatically resizing vPars” on page 36 or “Automatically resizing nPars using iCOD software” on page 37 for pointers to white papers that address partition management in more detail. Otherwise, select the **Next** button to continue.



16. Specify the full path and filename indicating where to save the configuration then select the **Next** button. In this example, the configuration will be saved as `/tmp/usage-based_allocation.wlm`.



17. Close the wizard by selecting the **Finish** button in the next screen.



18. Activate the new configuration file, instructing WLM to take control of resource management, using the following `wlmd` command:

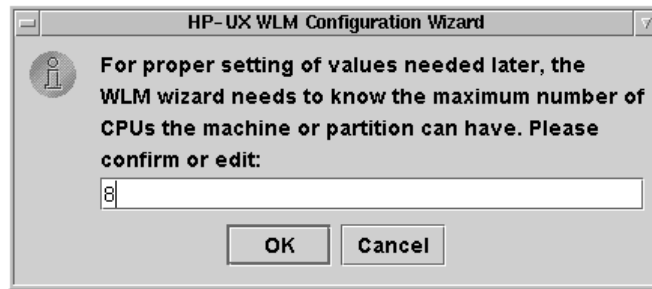
```
# wlmd -a /usage-based_allocation.wlm
```

## Providing CPU for a given time period

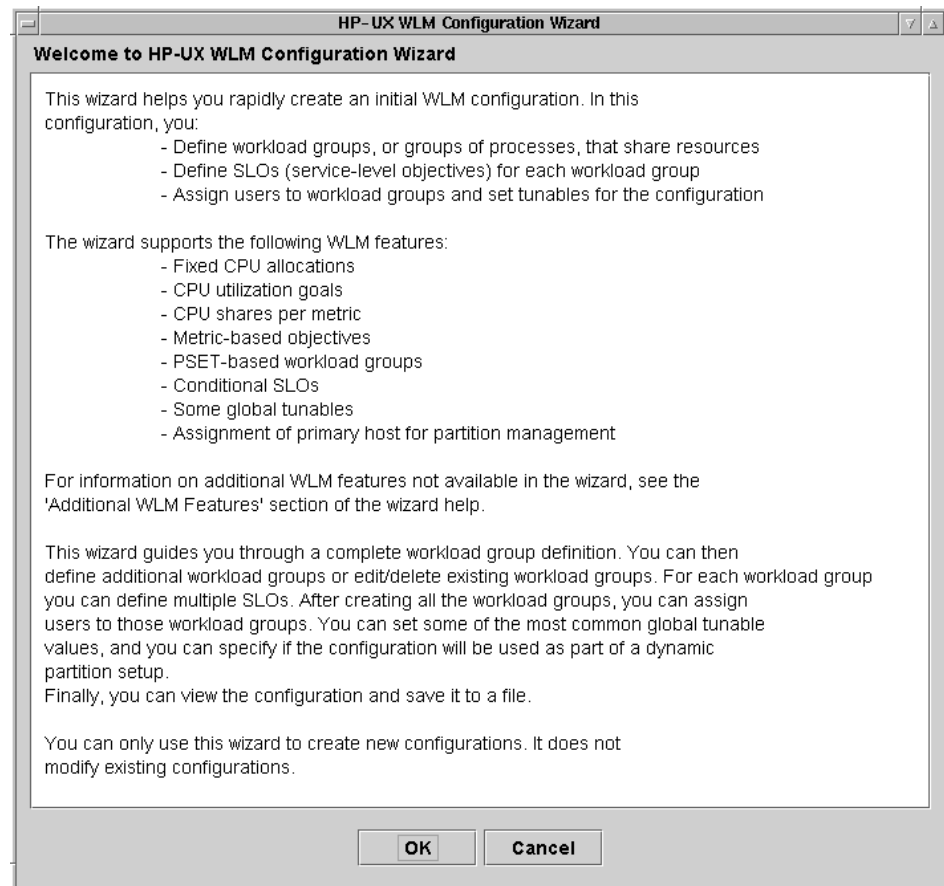
For workloads that are only needed for a certain period of time, whether it is once a day, week, or any other period, WLM provides a way to make SLOs conditional on time and date ranges. With a conditional SLO, the SLO is active and influences the workload's CPU allocation only when the condition, be it a time range or date, is true.

To provide a workload group with CPU for a given time period:

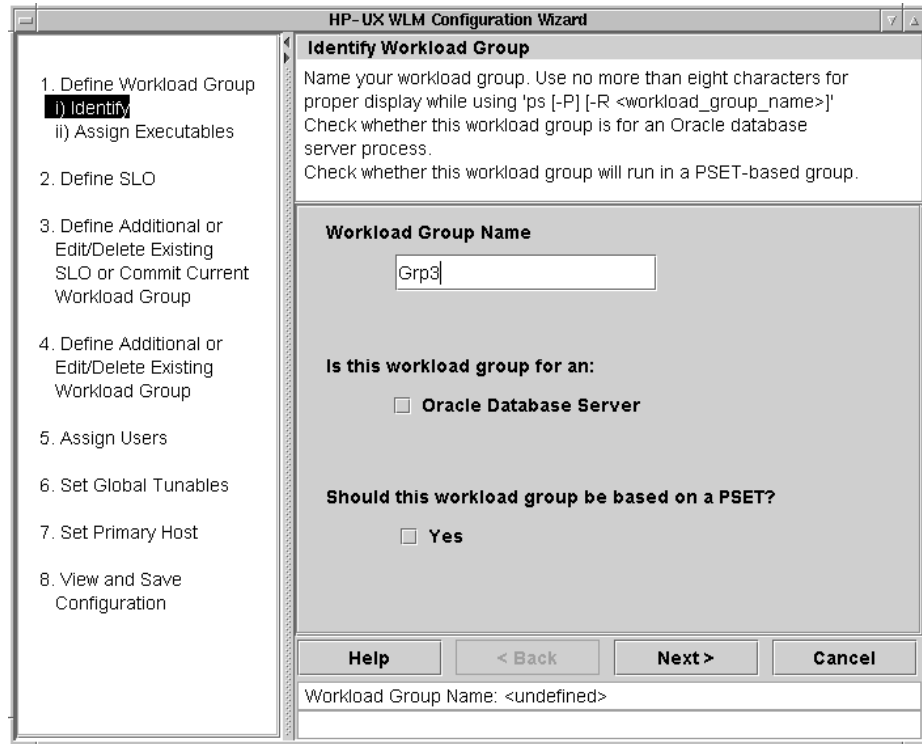
1. Enter the maximum number of CPUs the system will have.  
The wizard guesses at the maximum value. Confirm or edit the guess. The value you enter should take into account changes in the CPU count due to iCOD and other such adjustments to the CPU count.



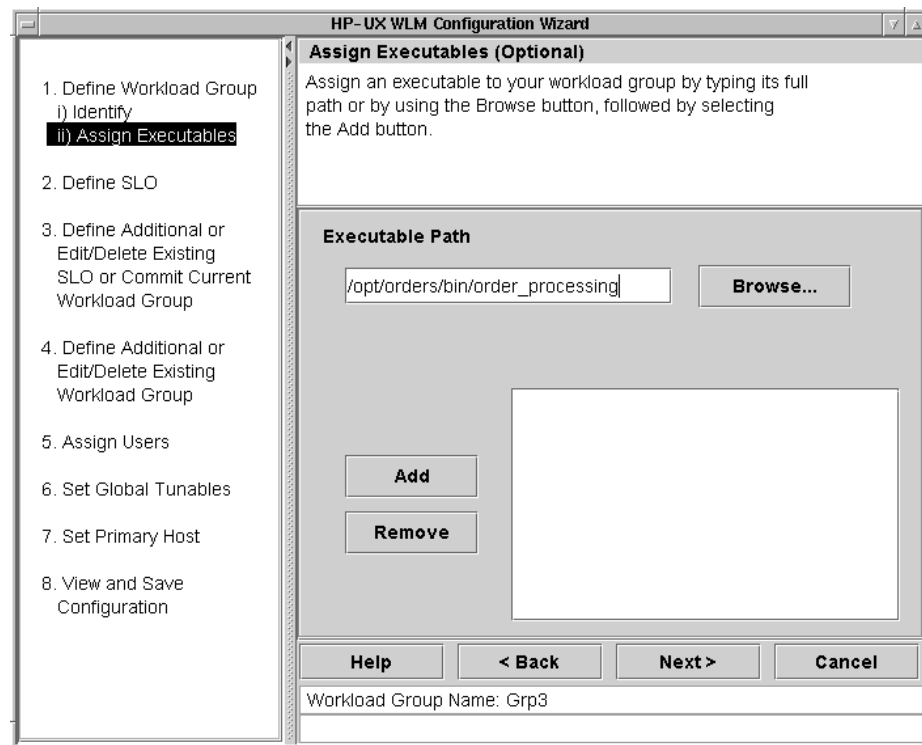
2. Read the introduction screen and press the **OK** button.



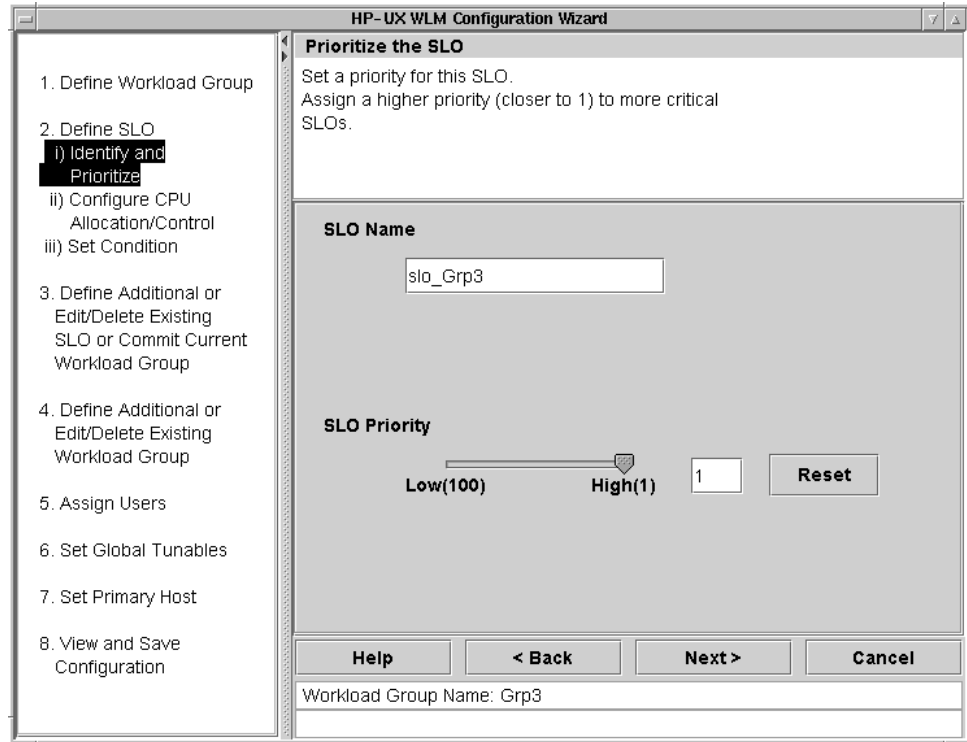
3. Name the workload group and press the **Next** button. In the screen below, the group is named Grp3.



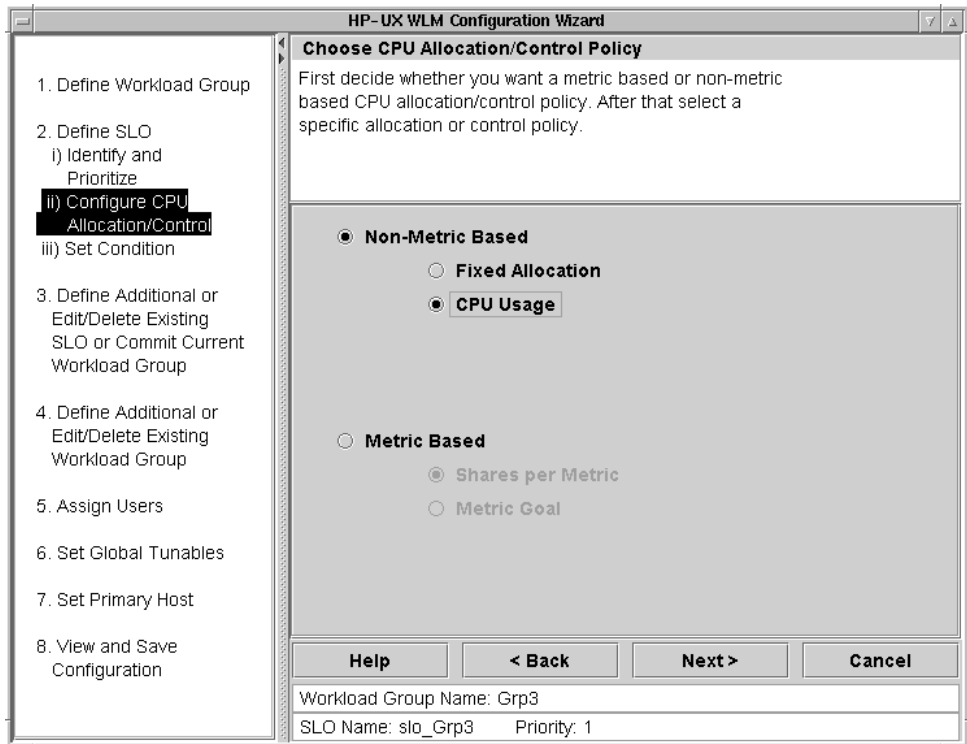
4. Assign a workload to the workload group you defined in the previous step. Specify the **Executable Path** for an application that you want in the workload group and select the **Add** button. Add more applications to the workload group in the same fashion. Select the **Next** button when ready to continue.



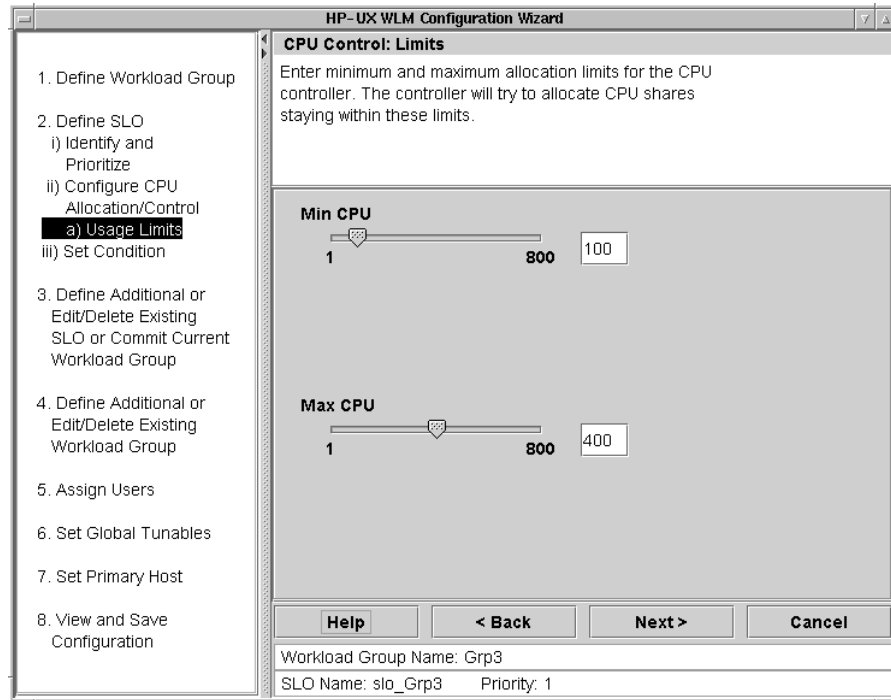
- Now enter data about the service-level objective (SLO), or goal, for the workload. Specify an **SLO Name** and assign an **SLO Priority** by moving the slider or entering a value (between 1 and 100) in the text field. Each SLO needs a priority so that WLM can choose which SLOs get CPU when there is not enough for all the workloads. The highest priority is 1; the lowest is 100. Select the **Next** button to continue.



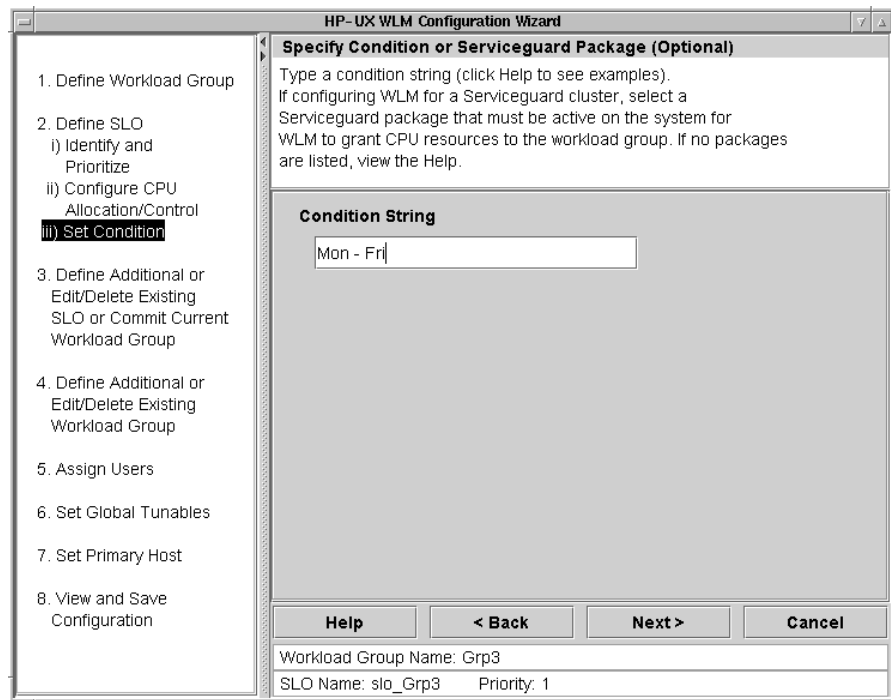
- Set the CPU allocation to **CPU Usage** and select the **Next** button.



7. WLM automatically adjusts the CPU allocation for a workload when you have a usage-based allocation. You can request that WLM's adjustments fall within a minimum allocation and maximum allocation. Use the slider or input fields to set these bounds. Then, select the **Next** button. NOTE: The **Min CPU** and **Max CPU** values are merely requests; they are not hard limits. The requests below are for 100 and 400 CPU shares, meaning one CPU and four CPUs.

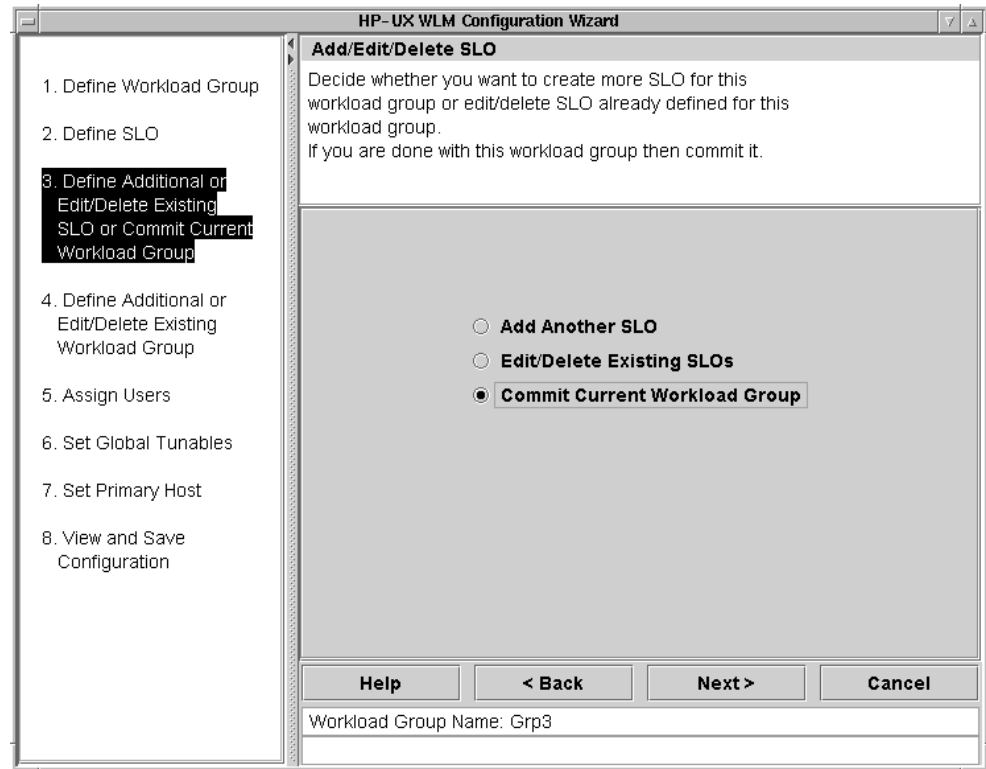


8. This screen allows us to enter a **Condition String** that indicates a condition that must be met for the SLO to be active. We want the SLO to be active Monday through Friday, so we enter the corresponding string and select the **Next** button. (For string syntax, see the online help.)

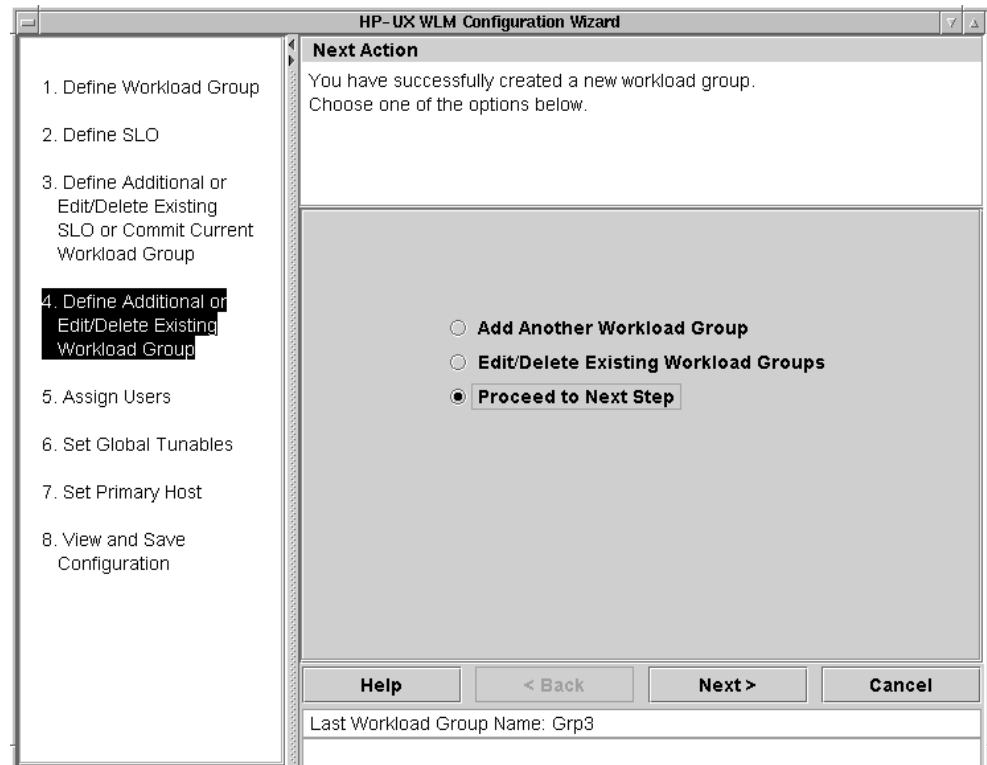




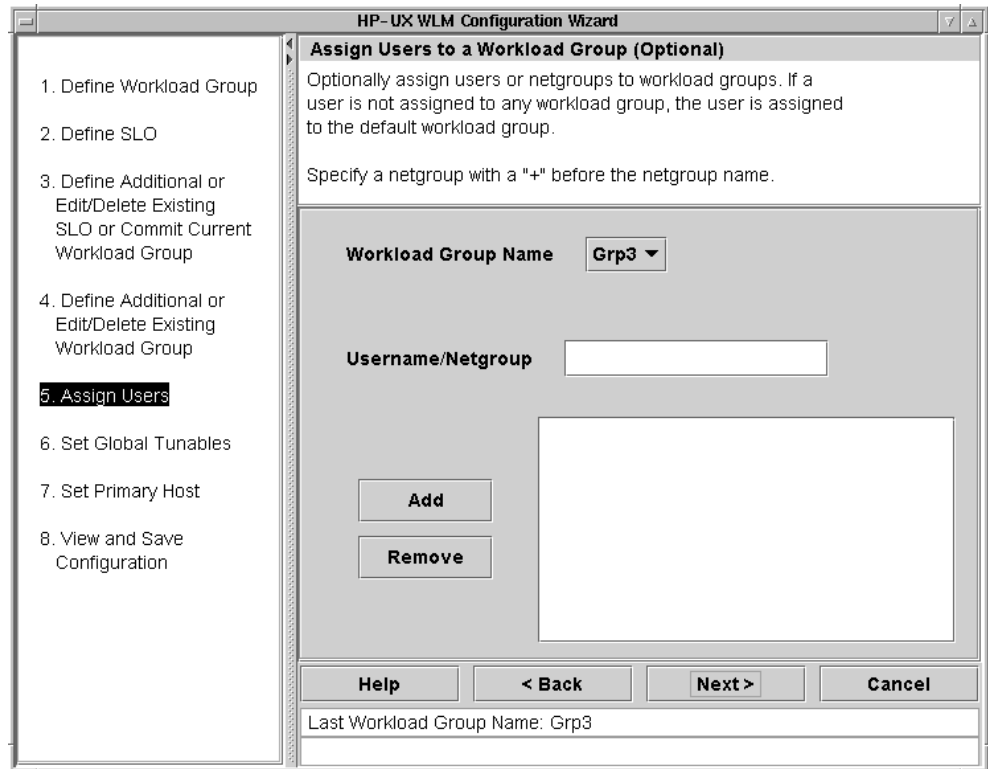
9. This screen allows us to add more SLOs for the workload or edit the existing SLO. We will use only the one SLO we just defined. Select the **Commit Current Workload Group** option then the **Next** button.



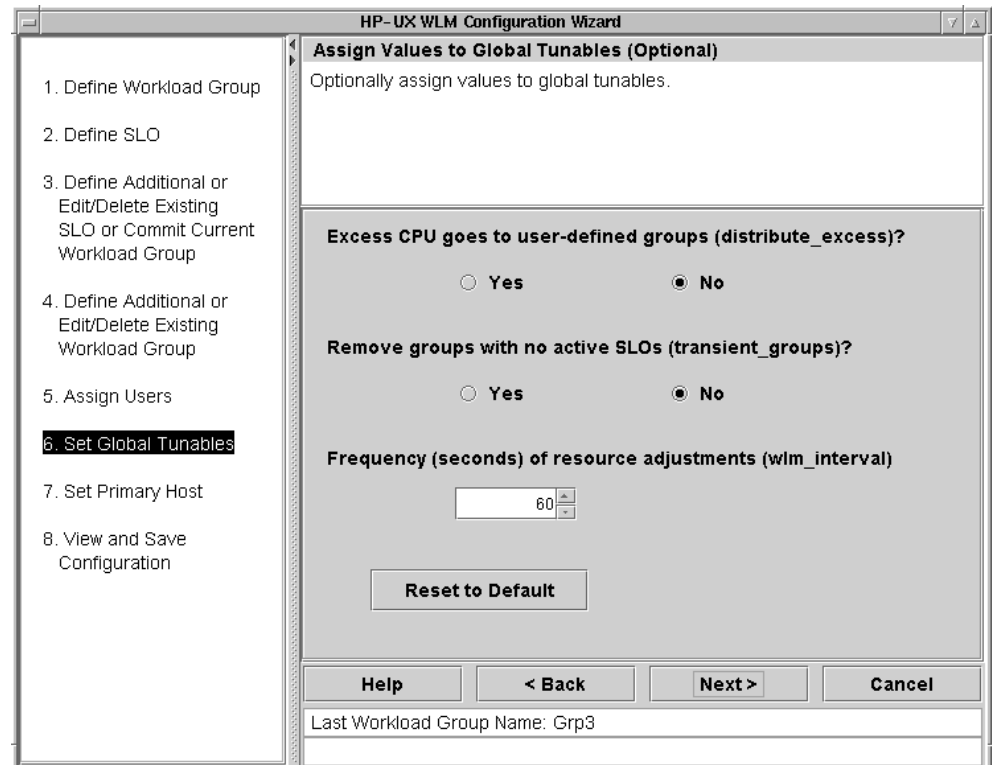
10. You have set up a CPU usage goal for a single workload. You can now continue to add workload groups (using a usage goal again or trying other types of allocations), edit the workload group we just defined, or continue with the WLM configuration. For brevity, we will select **Proceed to Next Step**, followed by the **Next** button.



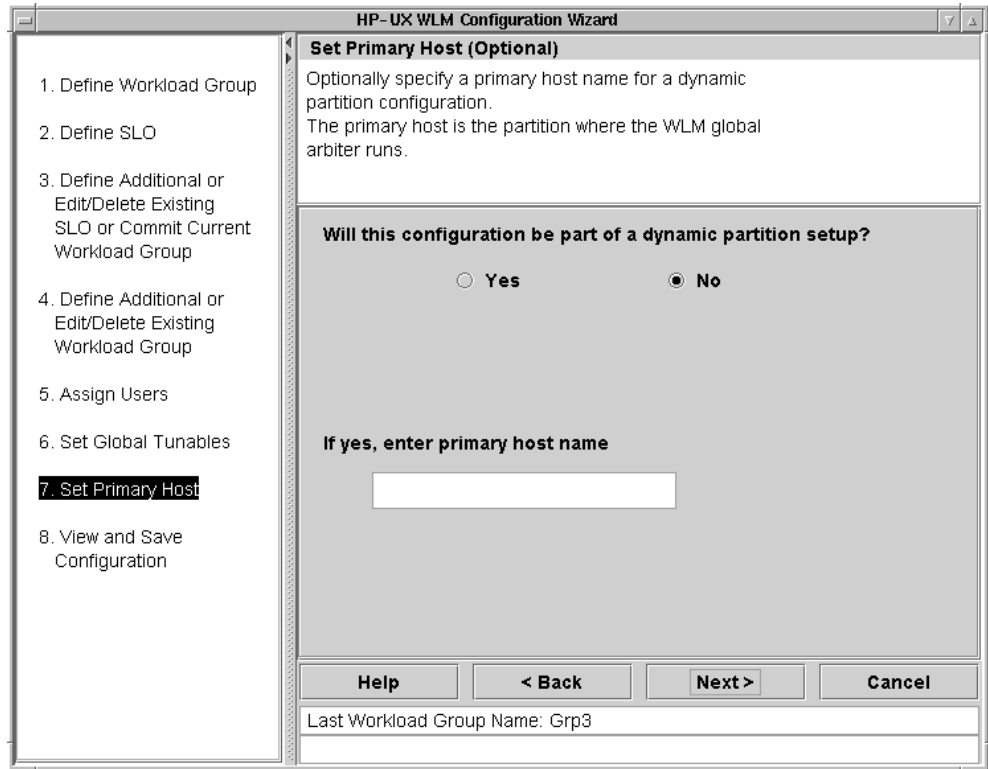
11. Assign users to the workloads if desired. User assignments ensure all the processes of a given user start in the specified workload. We are not concerned with user assignments, so skip these assignments by simply selecting the **Next** button.



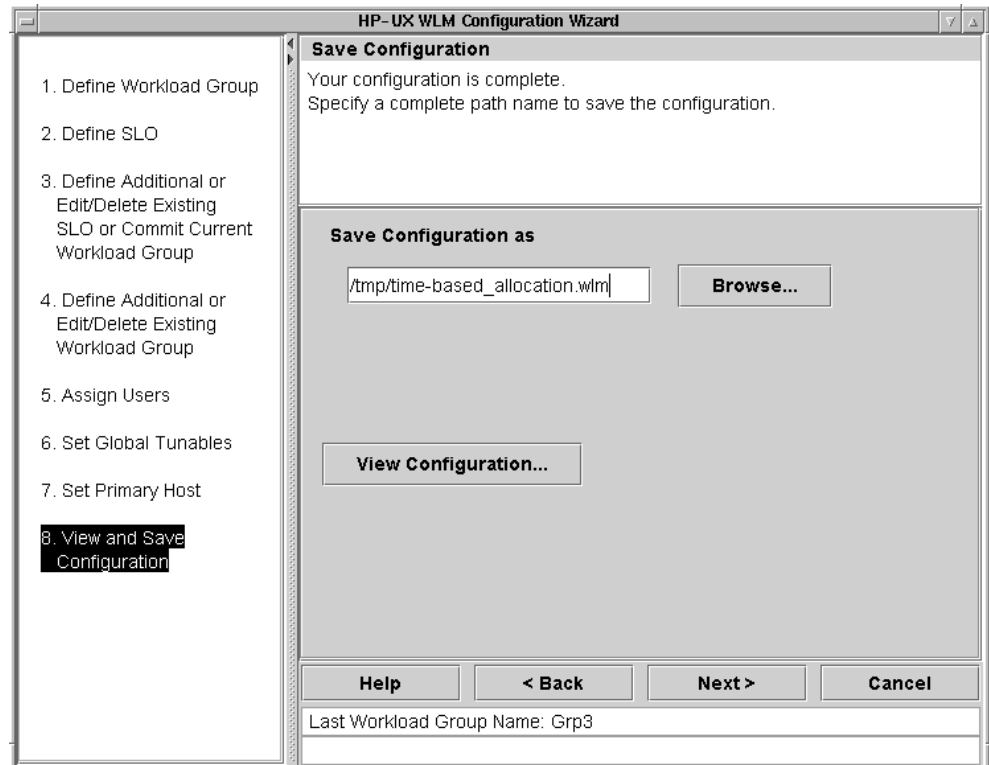
12. This screen allows you to set global tunables. Select the **Next** button to accept the defaults and continue. The second tunable, about transient\_groups, is available because of the condition string we entered earlier.



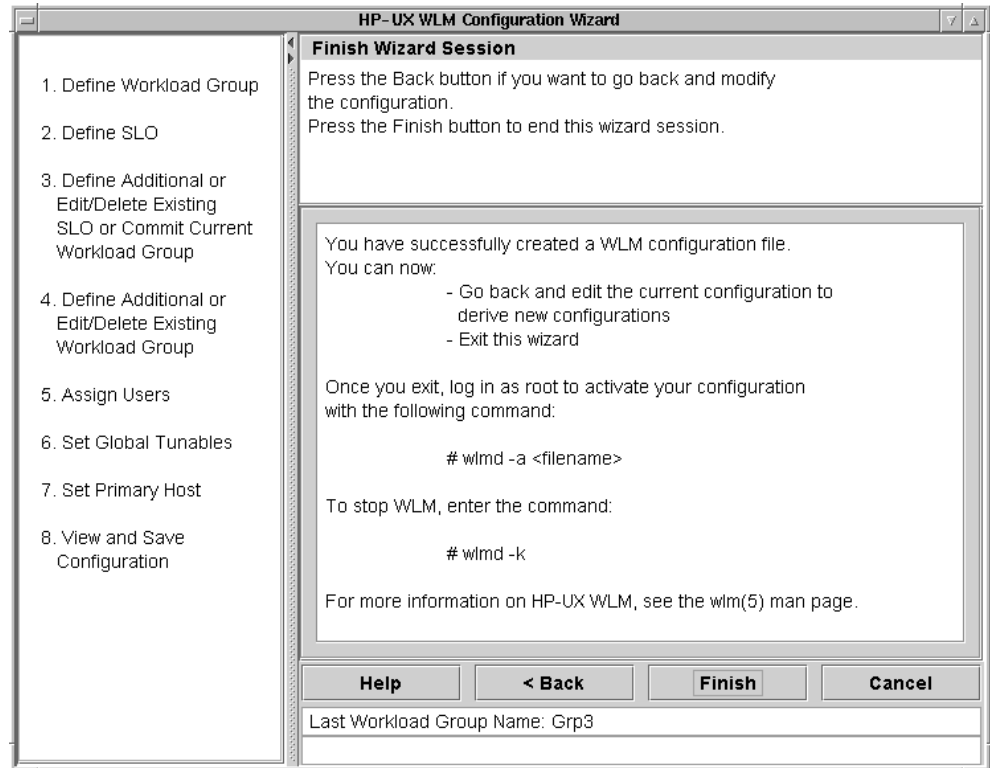
13. This next screen appears only if the system has HP-UX Virtual Partitions (vPars) or Node Partitions (nPars) installed. If you plan on managing vPars or nPars, see the section “Automatically resizing vPars” on page 36 or “Automatically resizing nPars using iCOD software” on page 37 for pointers to white papers that address partition management in more detail. Otherwise, select the **Next** button to continue.



14. Specify the full path and filename indicating where to save the configuration then select the **Next** button. In this example, the configuration will be saved as /tmp/time-based\_allocation.wlm.



15. Close the wizard by selecting the **Finish** button in the next screen.



16. Activate the new configuration file, instructing WLM to take control of resource management, using the following `wlmd` command:

```
# wlmd -a /tmp/time-based_allocation.wlm
```

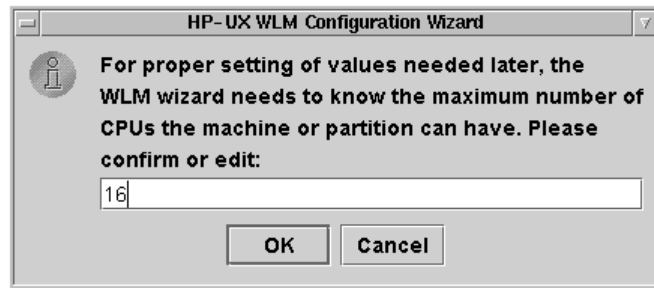
## Automatically resizing PSETs

With multi-processor systems running HP-UX 11i V1.0 (B.11.11) or later, you can group processors together to form processor sets, also known as PSETs. By creating PSETs, you isolate CPU resources for users or applications.

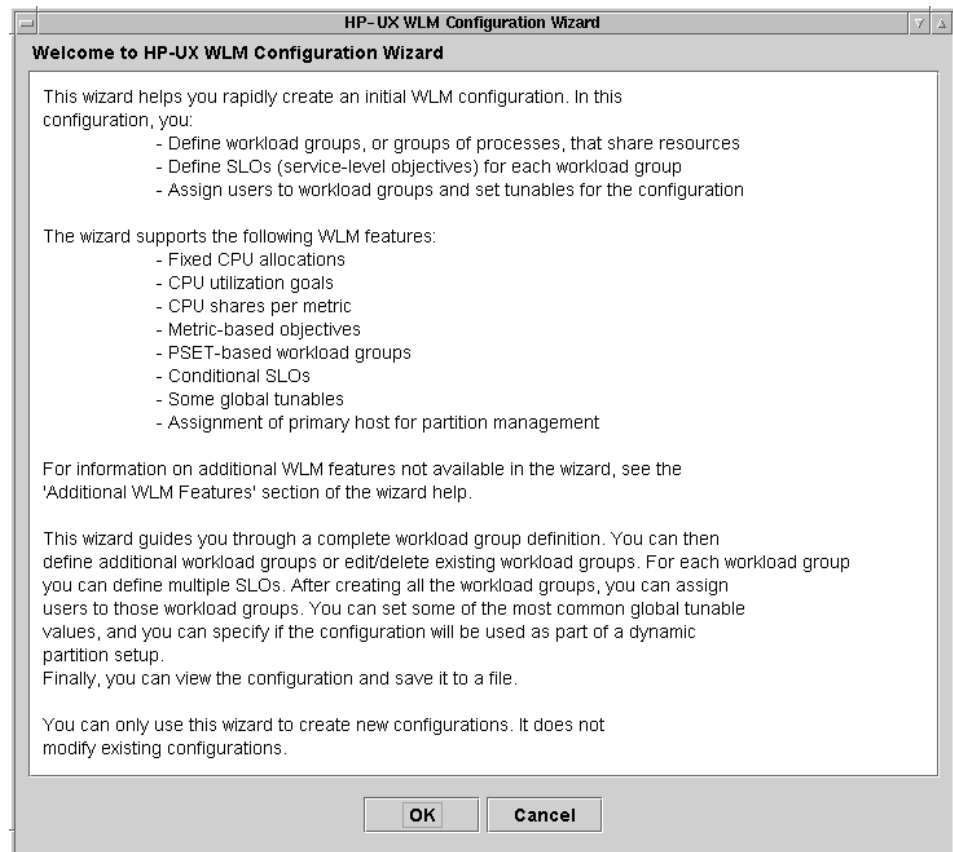
WLM allows you to define workload groups based on PSETs. If you then specify SLOs for these workload groups, WLM automatically adjusts the number of processors in the PSETs based on progress toward the SLOs.

To implement automatically resizing PSETs:

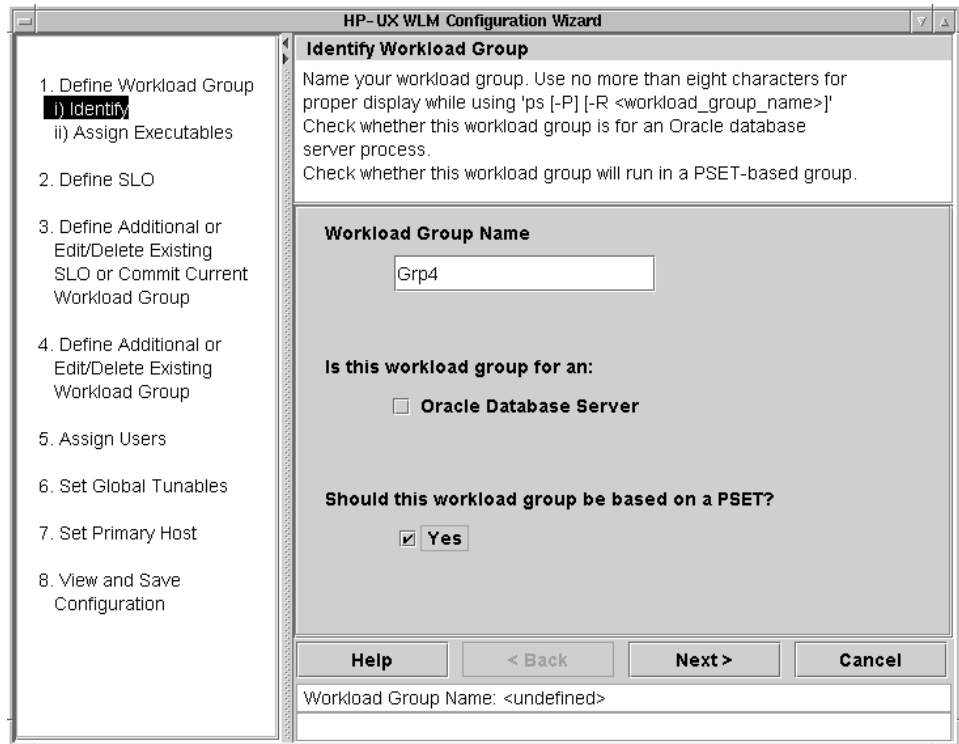
1. Enter the maximum number of CPUs the system will have.  
The wizard guesses at the maximum value. Confirm or edit the guess. The value you enter should take into account changes in the CPU count due to iCOD and other such adjustments to the CPU count.



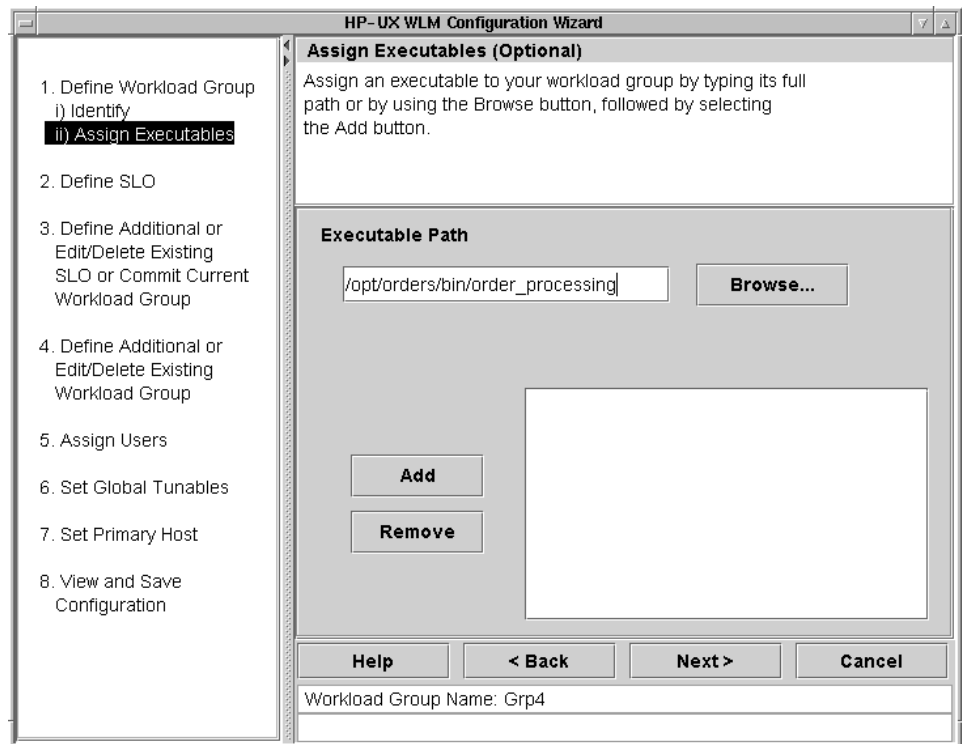
2. Read the introduction screen and press the **OK** button.



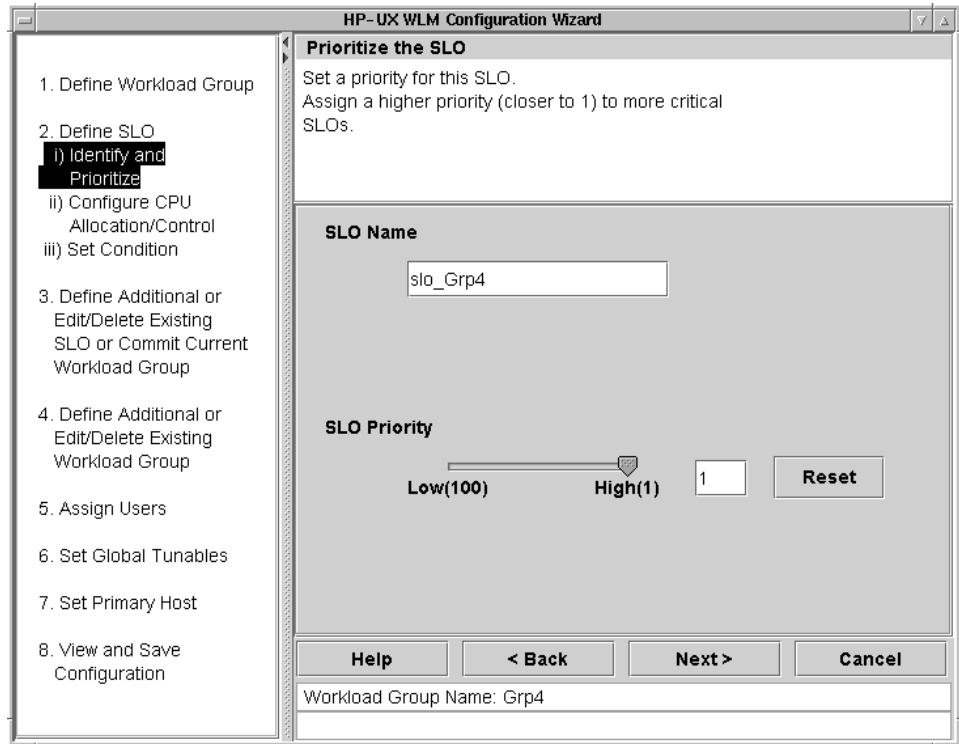
3. Name the workload group and press the **Next** button. In the screen below, the group is named Grp4.



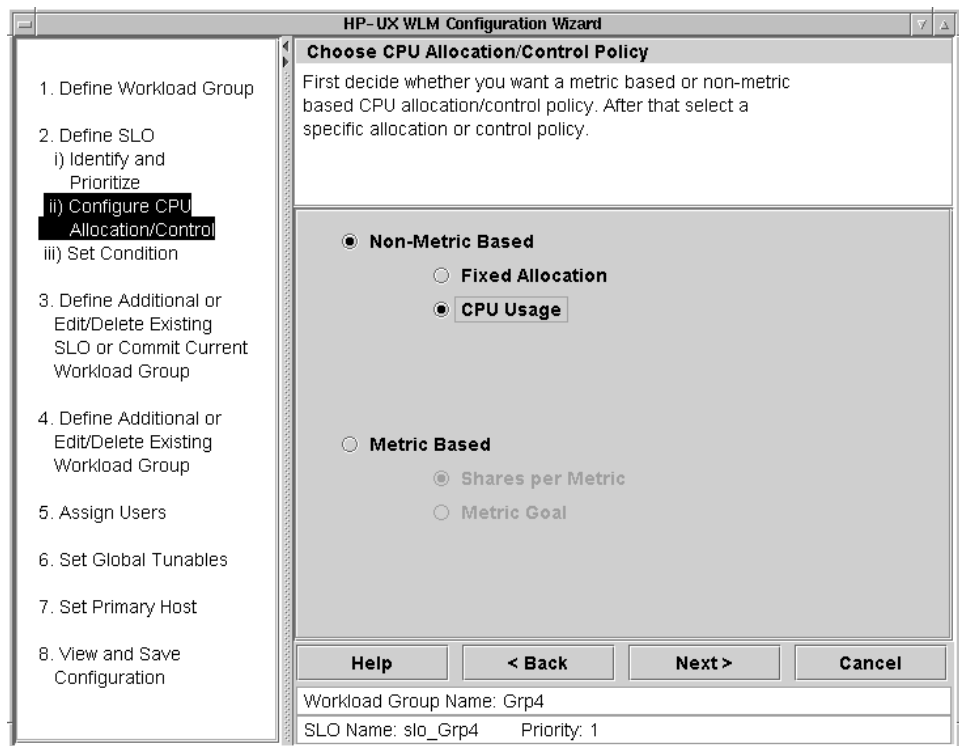
4. Assign a workload to the workload group you defined in the previous step. Specify the **Executable Path** for an application that you want in the workload group and select the **Add** button. Add more applications to the workload group in the same fashion. Select the **Next** button when ready to continue.



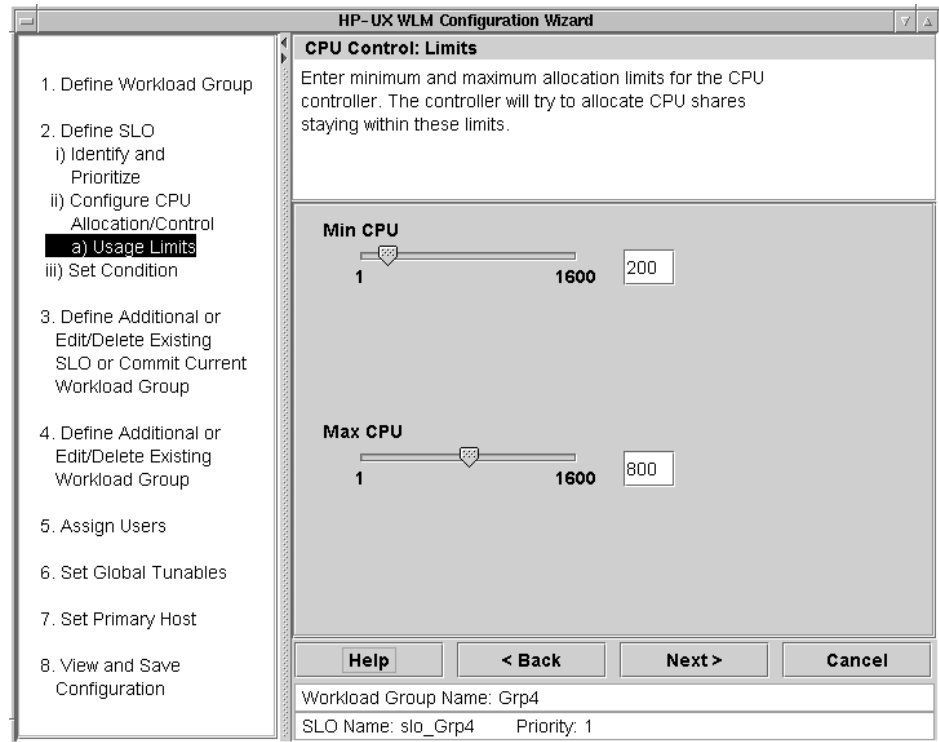
- Now enter data about the service-level objective (SLO), or goal, for the workload. Specify an **SLO Name** and assign an **SLO Priority** by moving the slider or entering a value (between 1 and 100) in the text field. Each SLO needs a priority so that WLM can choose which SLOs get CPU when there is not enough for all the workloads. The highest priority is 1; the lowest is 100. Select the **Next** button to continue.



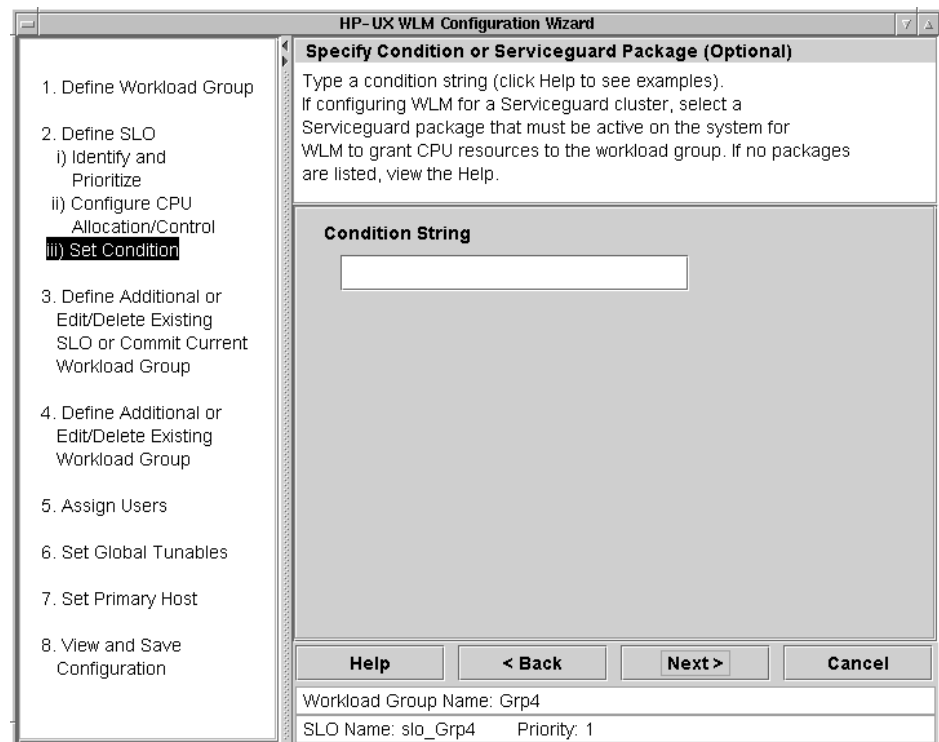
- Set the CPU allocation to **CPU Usage** then select the **Next** button.



7. WLM automatically adjusts the CPU allocation for a workload when you have a usage-based allocation. You can request that WLM's adjustments fall within a minimum allocation and maximum allocation. Use the slider or input fields to set these bounds. Then, select the **Next** button. NOTE: The **Min CPU** and **Max CPU** values are merely requests; they are not hard limits. The requests below are for 200 and 800 CPU shares, meaning two CPU and eight CPUs.

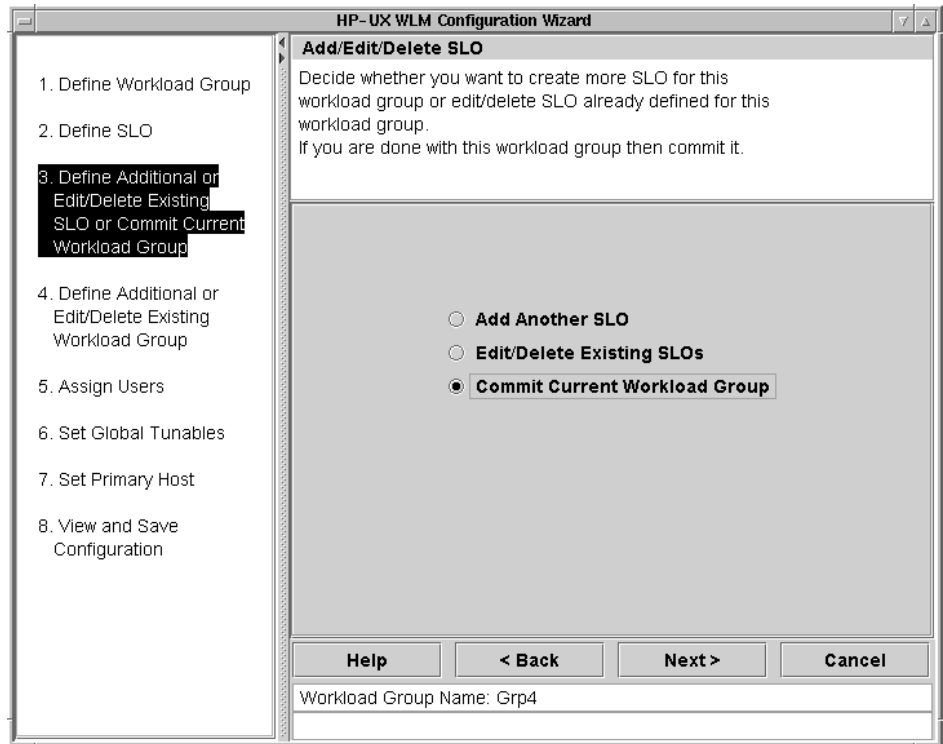


8. This screen allows us to enter a **Condition String** that indicates a condition that must be met for the SLO to be active. We want the SLO to be active all the time, so we enter nothing on this screen and simply select the **Next** button.

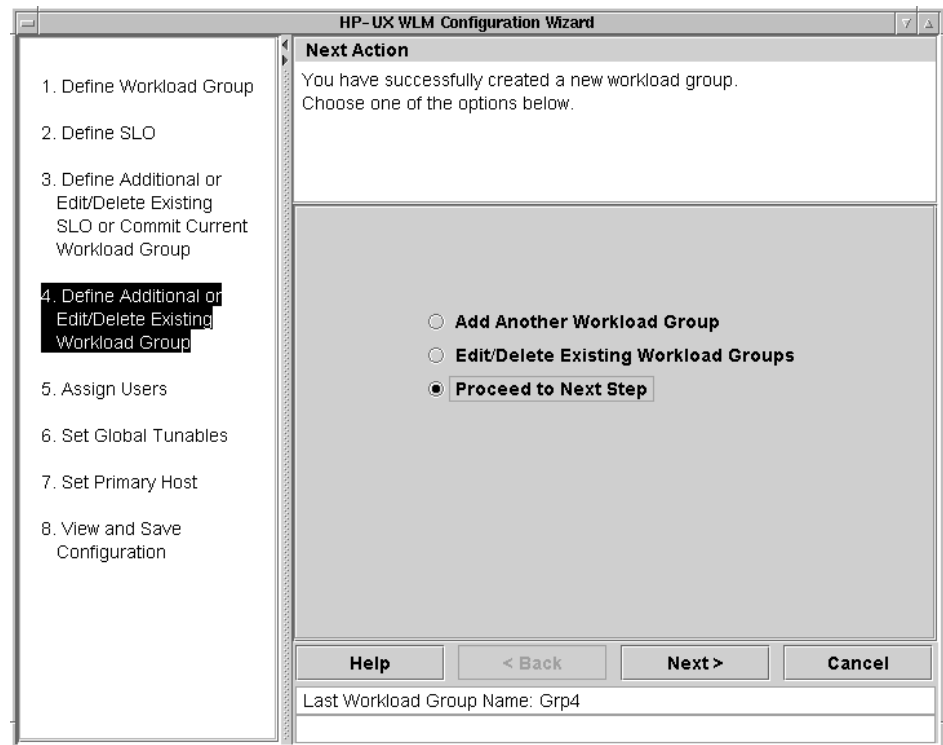




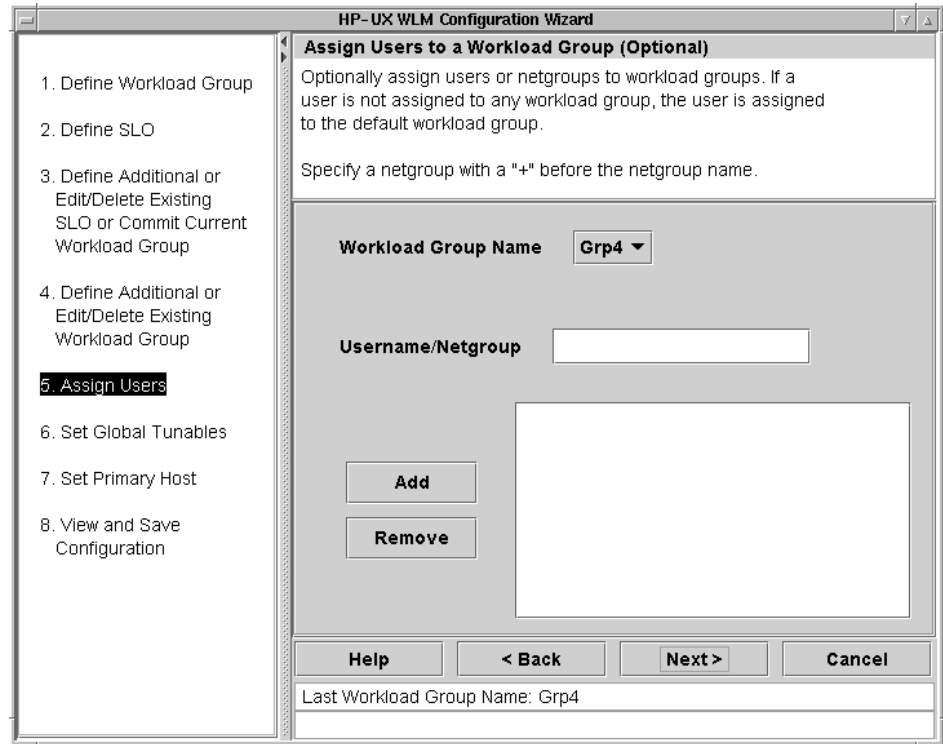
9. This screen allows us to add more SLOs for the workload or edit the existing SLO. We will use only the one SLO we just defined. Select the **Commit Current Workload Group** option then the **Next** button.



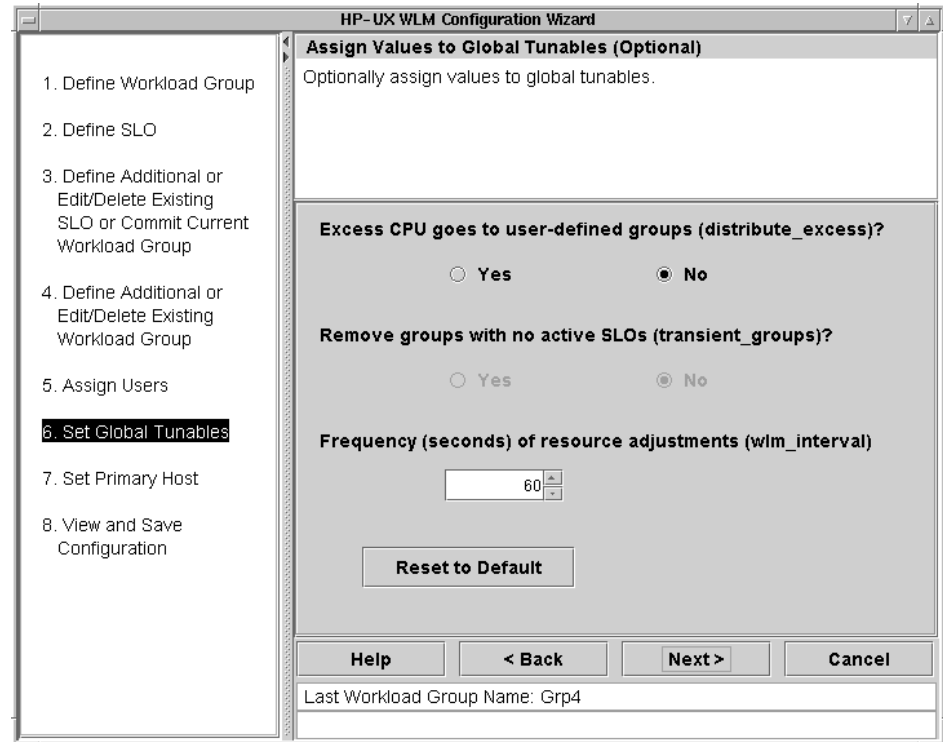
10. You have set up a CPU usage goal for a single workload. You can now continue to add workload groups (using a usage goal again or trying other types of allocations), edit the workload group we just defined, or continue with the WLM configuration. For brevity, we will select **Proceed to Next Step**, followed by the **Next** button.



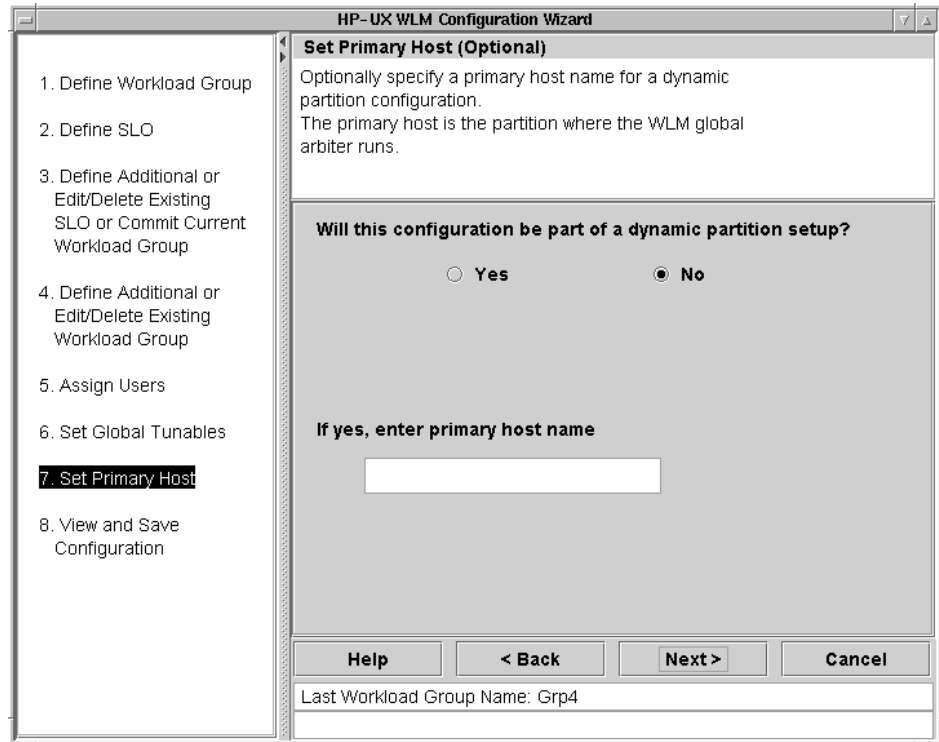
11. Assign users to the workloads if desired. User assignments ensure all the processes of a given user start in the specified workload. We are not concerned with user assignments, so skip these assignments by simply selecting the **Next** button.



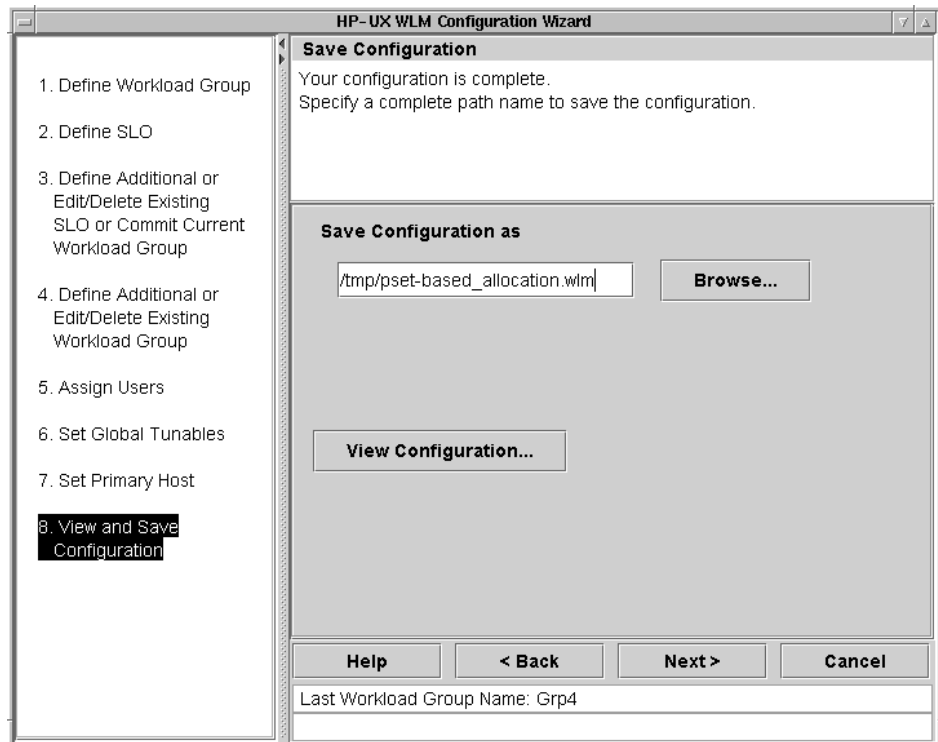
12. This screen allows you to set global tunables. Select the **Next** button to accept the defaults and continue.



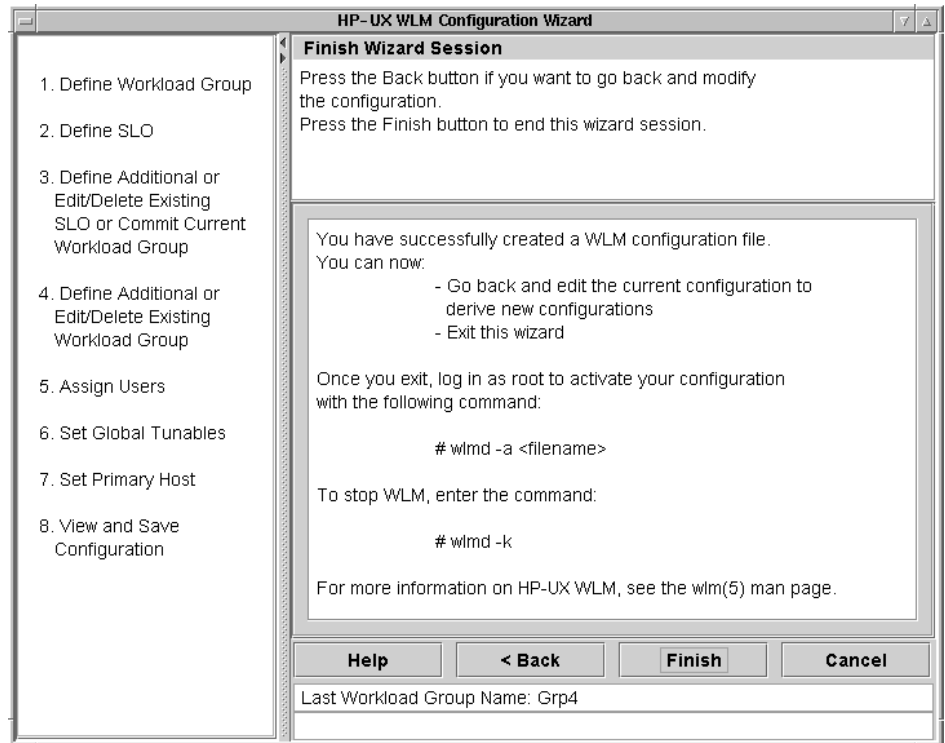
13. This next screen appears only if the system has HP-UX Virtual Partitions (vPars) or Node Partitions (nPars) installed. If you plan on managing vPars or nPars, see the section “Automatically resizing vPars” on page 36 or “Automatically resizing nPars using iCOD software” on page 37 for pointers to white papers that address partition management in more detail. Otherwise, select the **Next** button to continue.



14. Specify the full path and filename indicating where to save the configuration then select the **Next** button. In this example, the configuration will be saved as /tmp/pset-based\_allocation.wlm.



15. Close the wizard by selecting the **Finish** button in the next screen.



16. Activate the new configuration file, instructing WLM to take control of resource management, using the following `wlmd` command:

```
# wlmd -a /tmp/pset-based_allocation.wlm
```

## Automatically resizing vPars

HP-UX Virtual Partitions (vPars), which are implemented in software, provide complete software isolation. Each vPar runs its own instance of the HP-UX operating system. If you require such isolation, but want to optimize use of resources, you can use WLM to migrate CPUs between vPars based on the needs of the applications in those vPars.

WLM provides example configurations to simplify this CPU migration. In addition, there is a white paper dedicated to the topic. The paper, titled "Resizing vPars automatically with HP-UX Workload Manager," is available from <http://www.hp.com/go/wlm/>, on the "information library" page.

## Automatically resizing nPars using iCOD software

HP's node partitions (nPars) are implemented in hardware. As with vPars, each nPar runs its own copy of HP-UX. If each nPar on a system has iCOD software installed, WLM can simulate CPU migration between the nPars. The simulation is achieved by deactivating a CPU on one nPar then activating a CPU on another nPar where the resource is needed more. The total number of active CPUs remains the same, avoiding a charge for additional CPUs.

WLM provides example configurations to simplify this simulated CPU migration. In addition, there is a white paper dedicated to the topic. The paper, titled "Resizing nPars automatically with HP-UX Workload Manager," is available from <http://www.hp.com/go/wlm/>, on the "information library" page.

## Ensuring CPU allocations after a Serviceguard failover

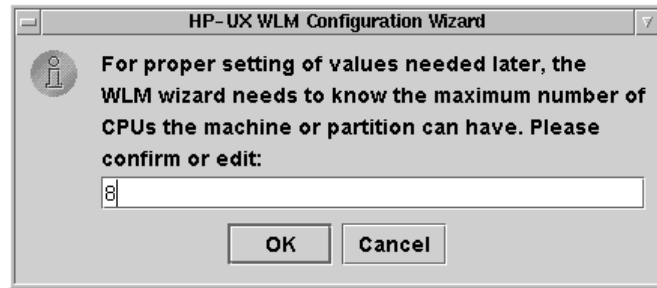
One approach for using WLM with Serviceguard is to create a single WLM configuration that you use on each node in the Serviceguard cluster. This one WLM configuration creates WLM workload groups for each Serviceguard package in the cluster. Then, whenever any package fails over to any node in the cluster, it has a workload group waiting for it. However, that group has active SLOs only when its package is active on that node. WLM then allocates resources on the node to groups with active SLOs, based on the priorities you assigned.

There is a white paper dedicated to the topic. The paper, titled "More efficient high availability through manageability (integrating Serviceguard and HP-UX Workload Manager)," is available from <http://www.hp.com/go/wlm/>, on the "information library" page.

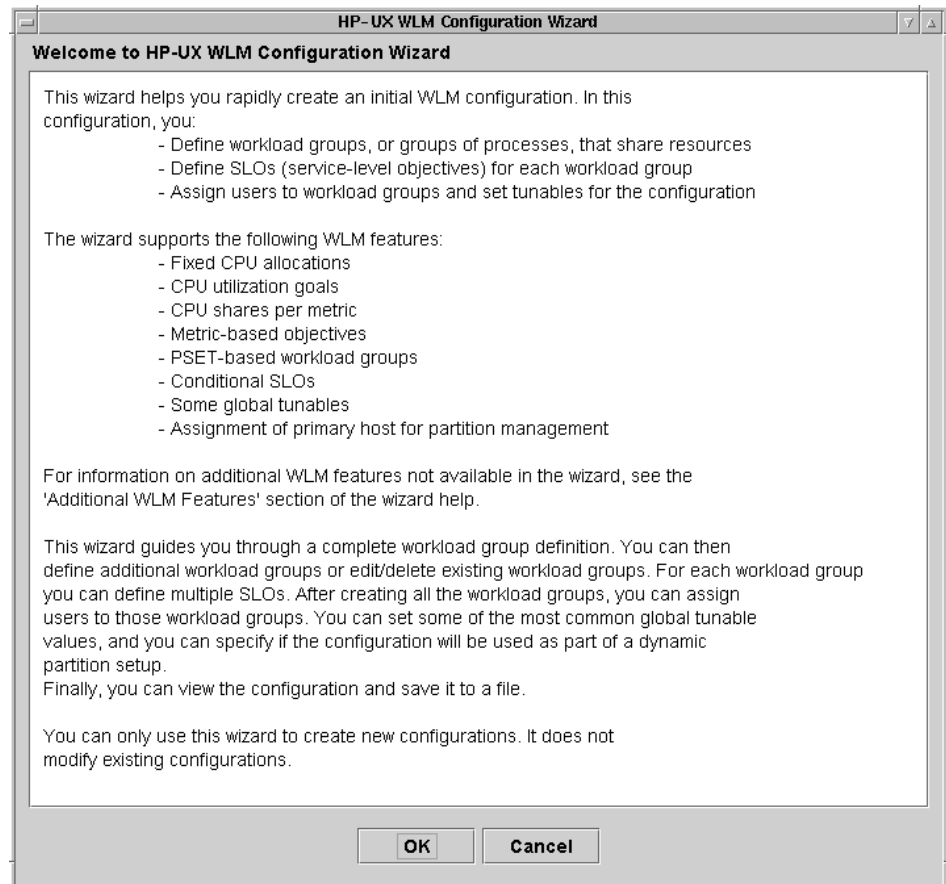
## Isolating Oracle database instances

By creating a WLM workload group for an Oracle database instance, you can separate it from other databases instances and applications on the system. Once separated, the instance can be allocated CPU to ensure its performance meets service-level objectives.

1. Enter the maximum number of CPUs the system will have.  
The wizard guesses at the maximum value. Confirm or edit the guess. The value you enter should take into account changes in the CPU count due to iCOD and other such adjustments to the CPU count.



2. Read the introduction screen and press the **OK** button.



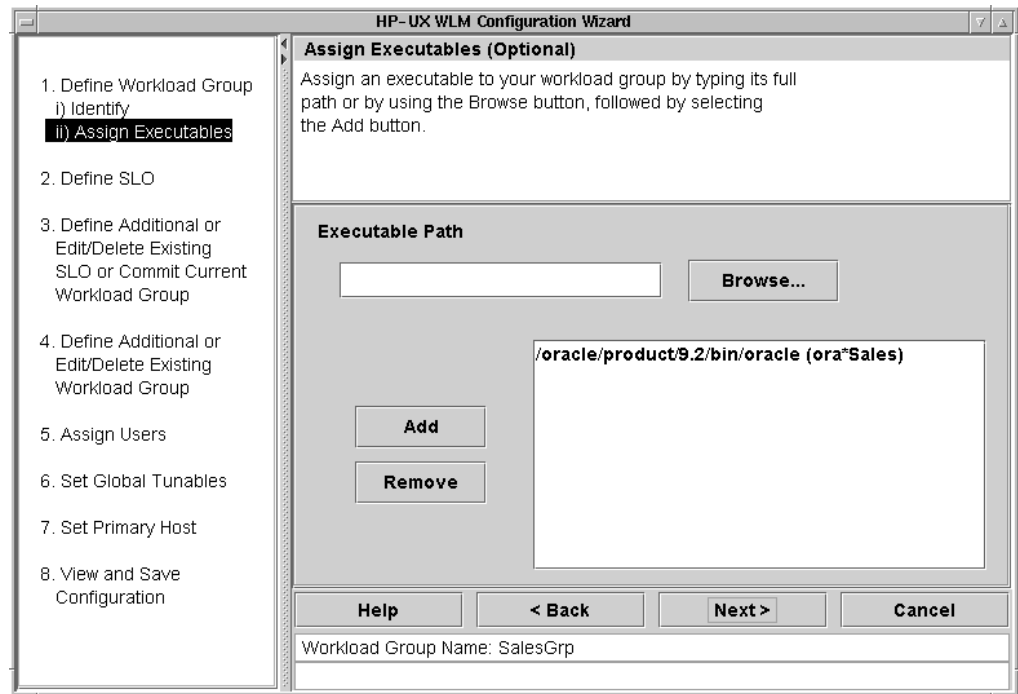
3. Name the workload group and press the **Next** button. In the screen below, the group is named SalesGrp.

The screenshot shows the 'HP-UX WLM Configuration Wizard' window. On the left, a list of steps is shown, with '1. Define Workload Group' expanded to 'i) Identify'. The main area is titled 'Identify Workload Group' and contains the following text: 'Name your workload group. Use no more than eight characters for proper display while using 'ps [-P] [-R <workload\_group\_name>]' Check whether this workload group is for an Oracle database server process. Check whether this workload group will run in a PSET-based group.' Below this, there is a 'Workload Group Name' field containing 'SalesGrp'. There are two checkboxes: 'Is this workload group for an:' with 'Oracle Database Server' checked, and 'Should this workload group be based on a PSET?' with 'Yes' unchecked. At the bottom, there are buttons for 'Help', '< Back', 'Next >', and 'Cancel'. A status bar at the very bottom shows 'Workload Group Name: <undefined>'.

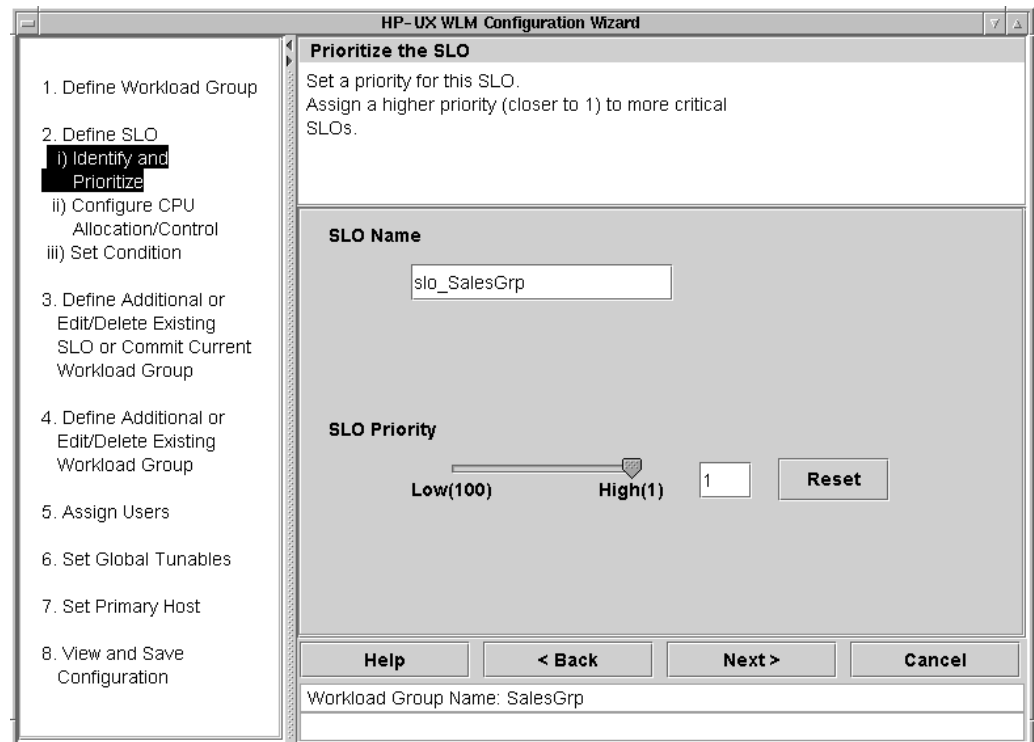
4. Specify values for the \$ORACLE\_HOME and \$ORACLE\_SID environment variables in the **Oracle Home** and **Oracle SID** fields, respectively, then select the **Next** button.

The screenshot shows the 'HP-UX WLM Configuration Wizard' window. On the left, the list of steps is expanded to 'ii) Assign Executables'. The main area is titled 'Specify Oracle Information' and contains the text: 'Assign an Oracle instance to your workload group by specifying its ORACLE\_HOME and ORACLE\_SID values.' Below this, there are two fields: 'Oracle Home' with the value '/oracle/product/9.2/' and a 'Browse...' button, and 'Oracle SID' with the value 'Sales'. At the bottom, there are buttons for 'Help', '< Back', 'Next >', and 'Cancel'. A status bar at the very bottom shows 'Workload Group Name: SalesGrp'.

- The following screen shows that the wizard has made sure that all the Oracle processes running under a name that matches the pattern `ora*Sales` will run in the SalesGrp workload we defined earlier. Select the **Next** button to continue.

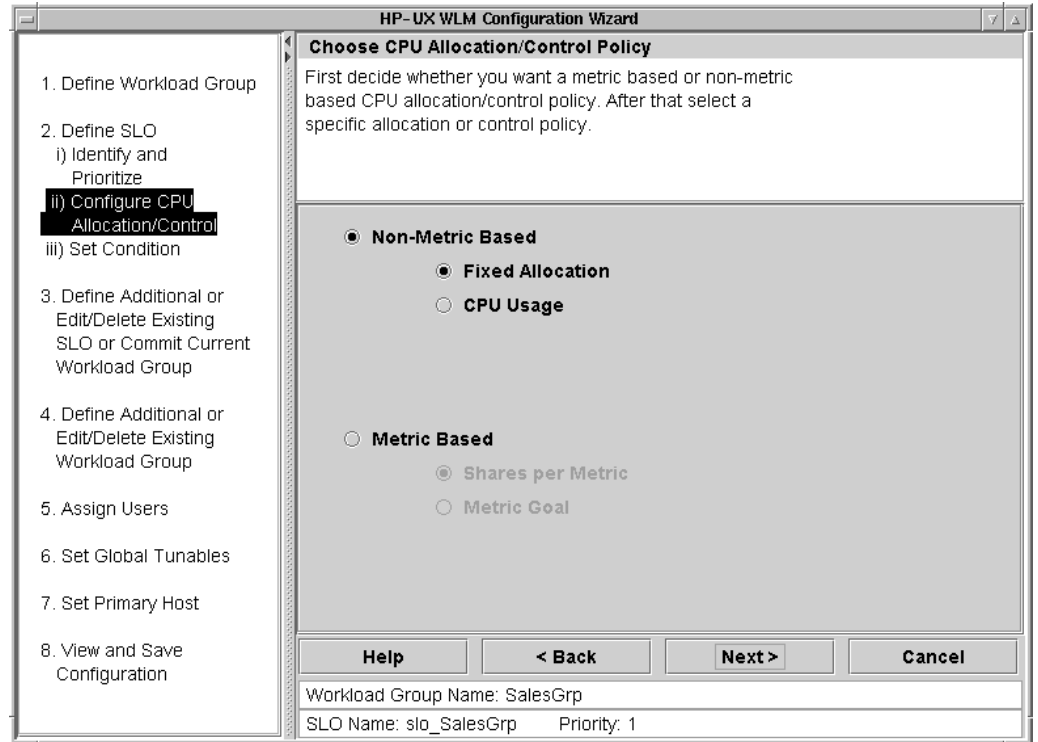


- Now enter data about the service-level objective (SLO), or goal, for the workload. Specify an **SLO Name** and assign an **SLO Priority** by moving the slider or entering a value (between 1 and 100) in the text field. Each SLO needs a priority so that WLM can choose which SLOs get CPU when there is not enough for all the workloads. The highest priority is 1; the lowest is 100. Select the **Next** button to continue.

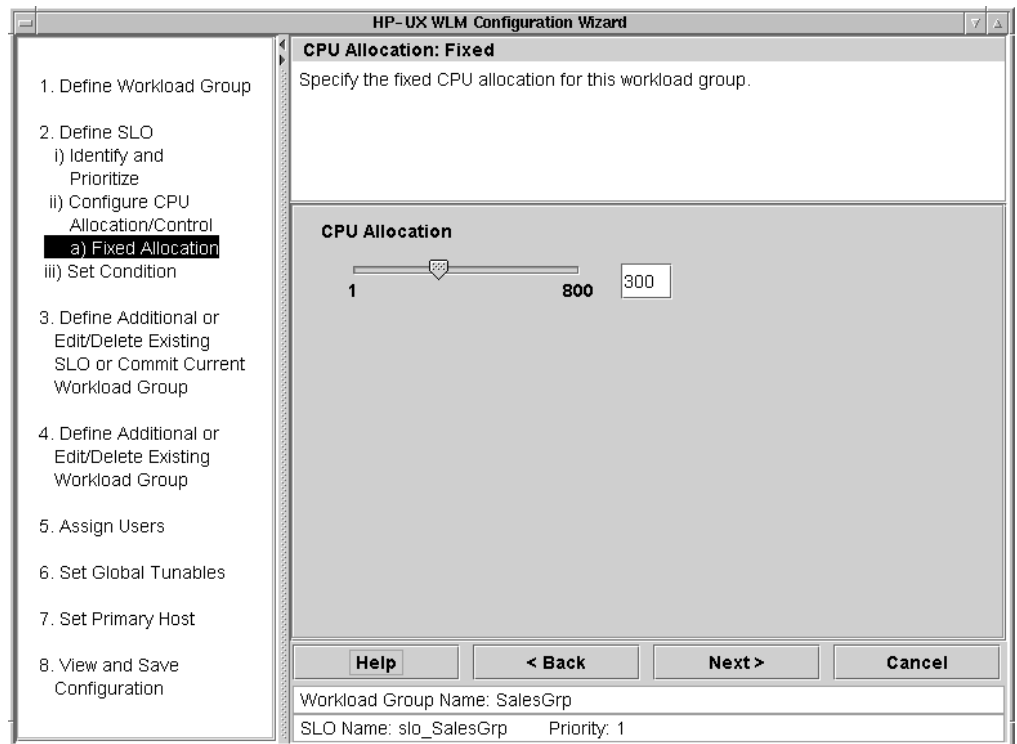




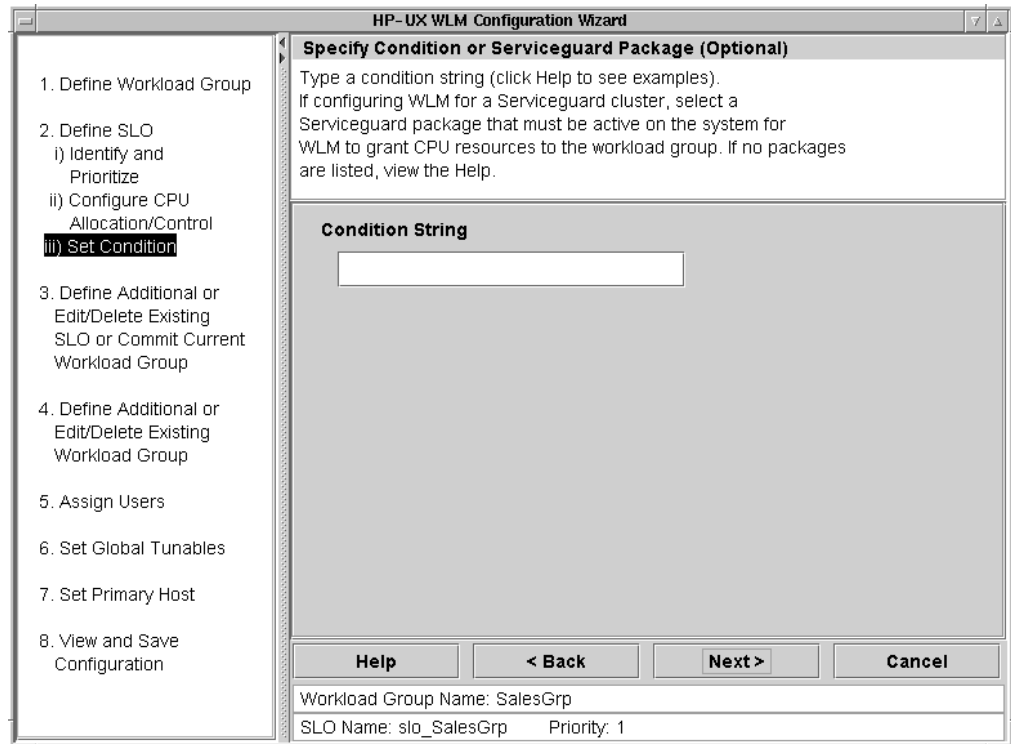
7. Choose a CPU allocation that is fixed by selecting the **Fixed Allocation** option. Then select the **Next** button to continue.



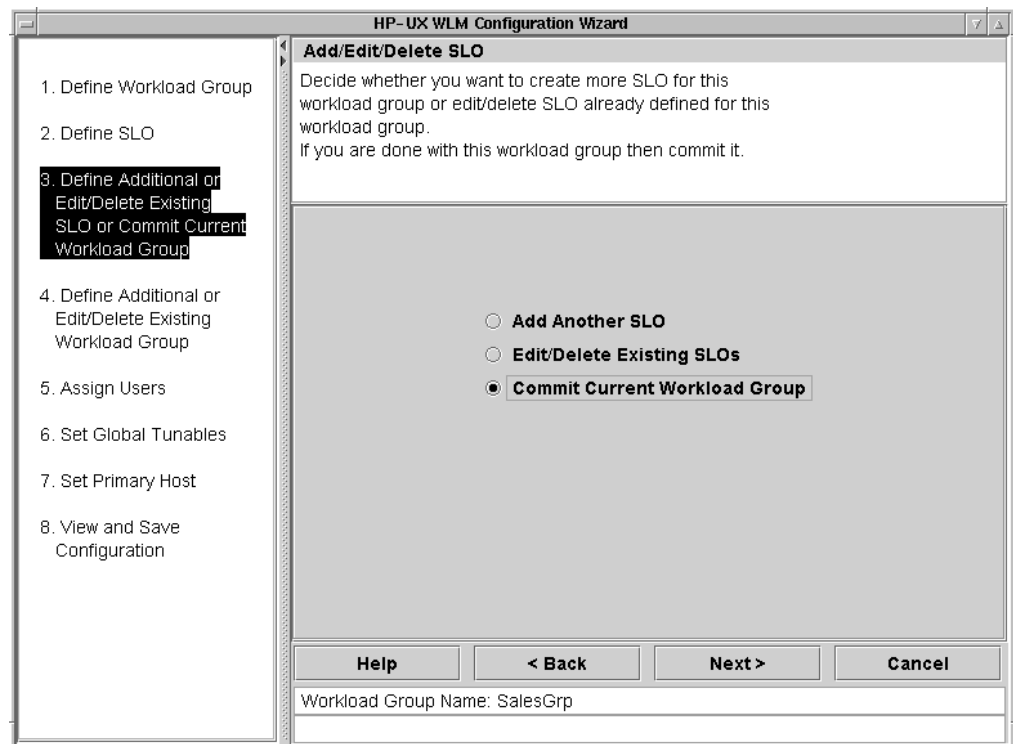
8. Specify the fixed amount of CPU the workload will receive using the slider or by typing in the text field. In this example, the workload gets 300 CPU shares or three CPUs. Select the **Next** button to continue.



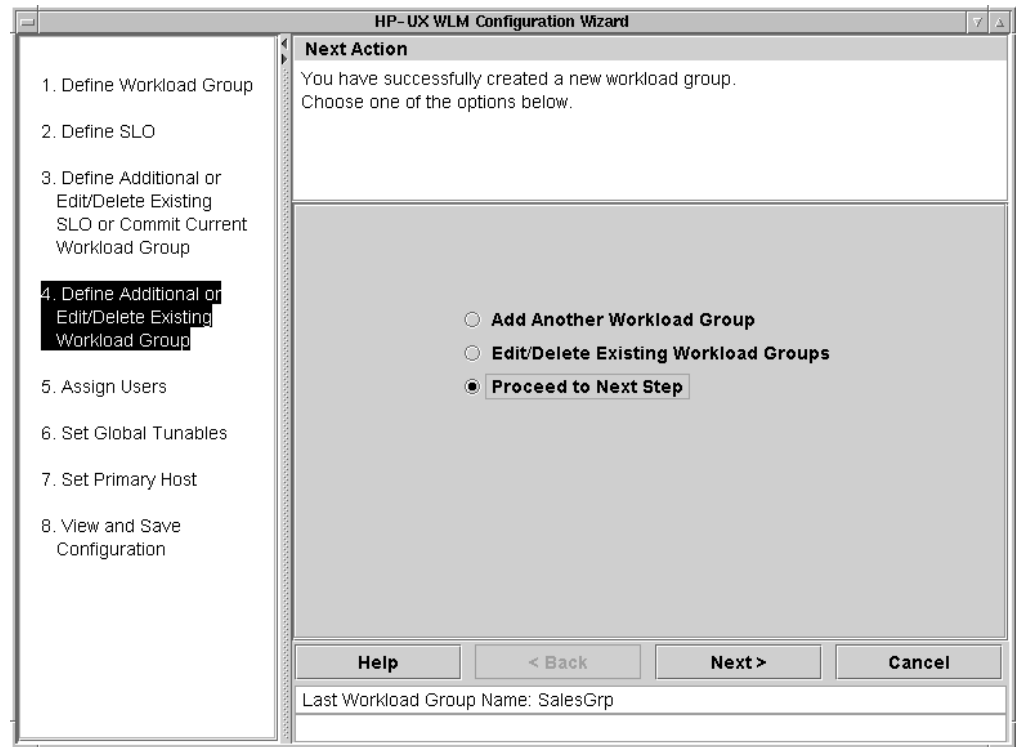
9. This screen allows us to enter a **Condition String** that indicates a condition that must be met for the SLO to be active. We want the SLO to be active all the time, so we enter nothing on this screen and simply select the **Next** button.



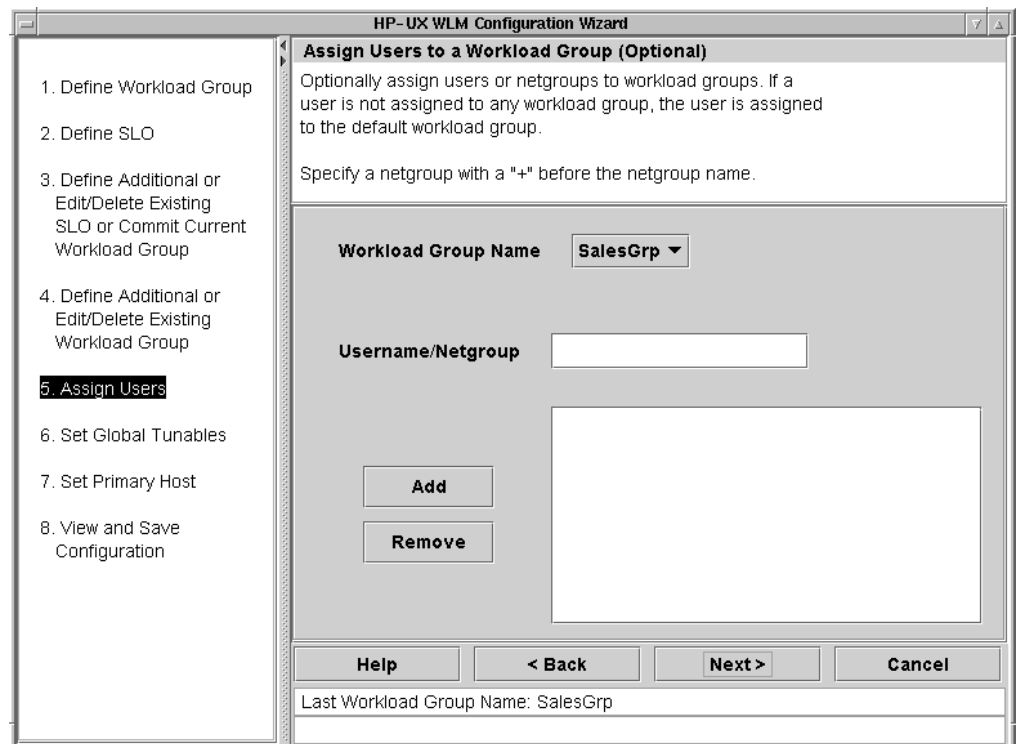
10. This screen allows us to add more SLOs for the workload or edit the existing SLO. We will use only the one SLO we just defined. Select the **Commit Current Workload Group** option then the **Next** button.



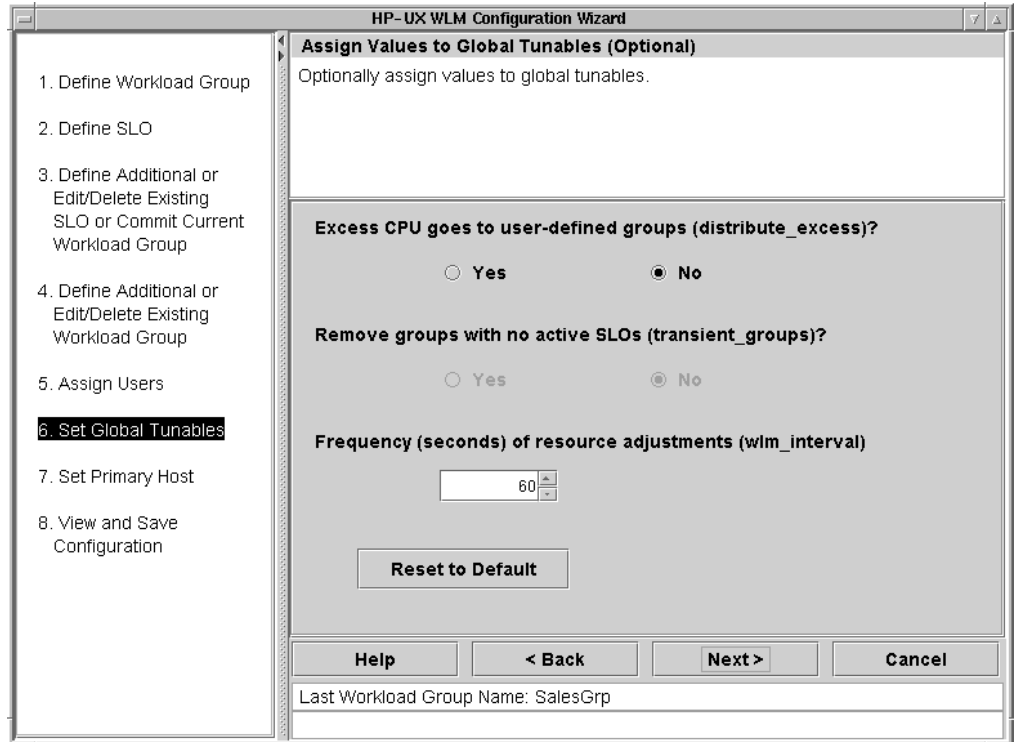
11. You have fixed the CPU allocation for a single database instance's workload group. You can now add workload groups for other database instances, edit the workload group we just defined, or continue with the WLM configuration. For brevity, we will select **Proceed to Next Step**, followed by the **Next** button.



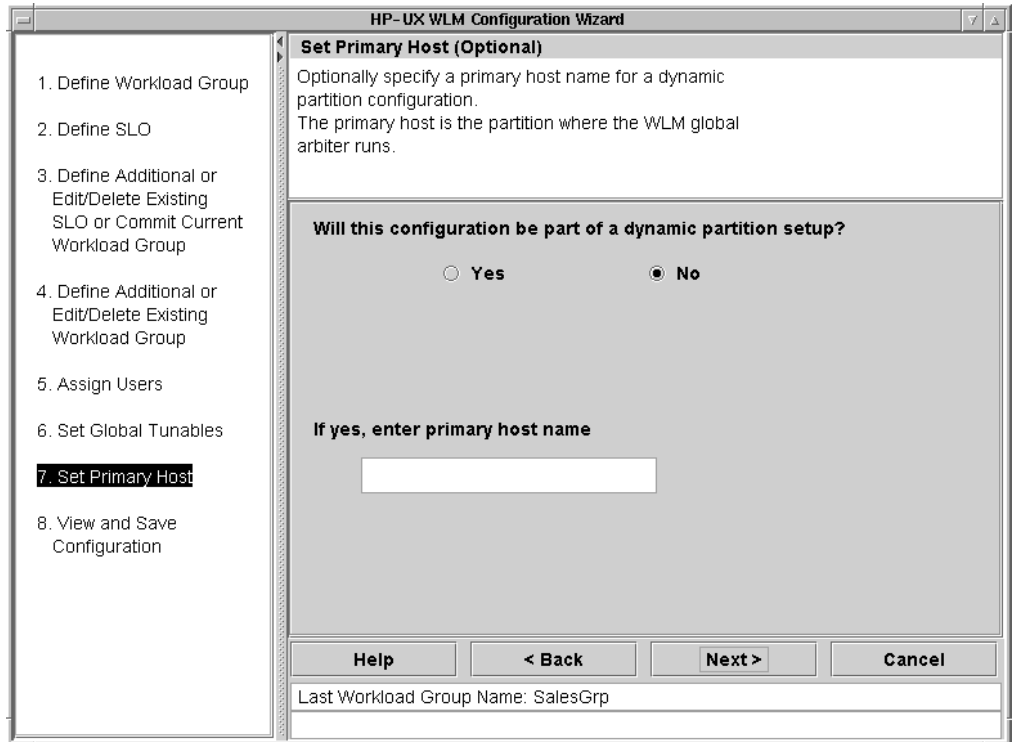
12. Assign users to the workloads if desired. User assignments ensure all the processes of a given user start in the specified workload. We are not concerned with user assignments, so skip these assignments by simply selecting the **Next** button.



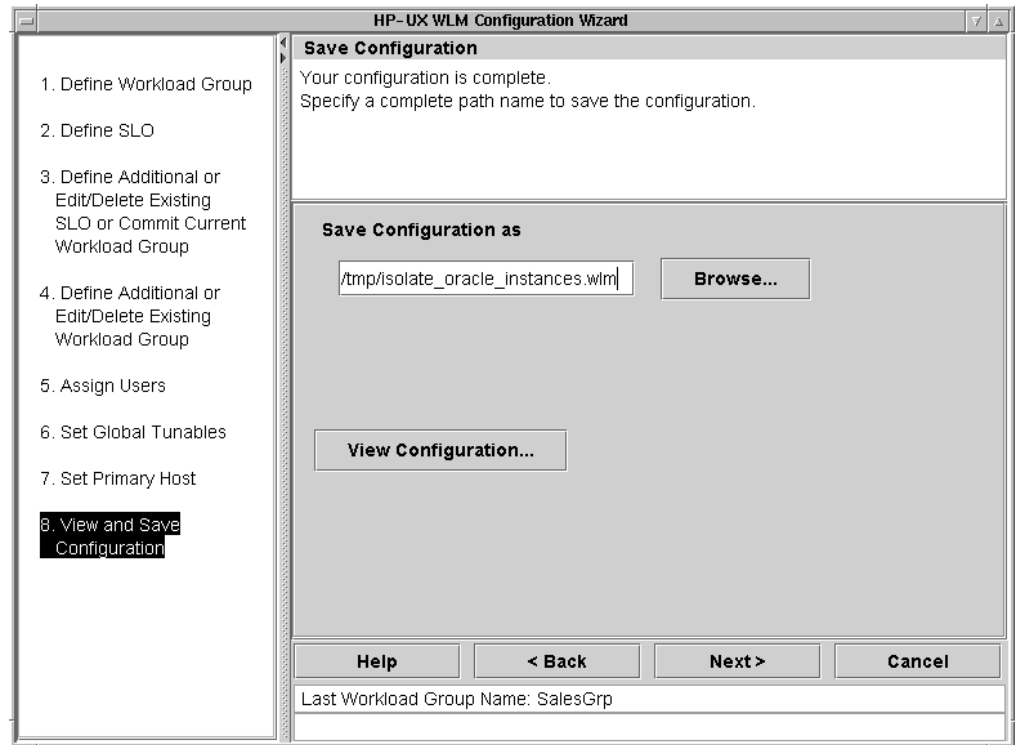
13. This screen allows you to set global tunables. Select the **Next** button to accept the defaults and continue.



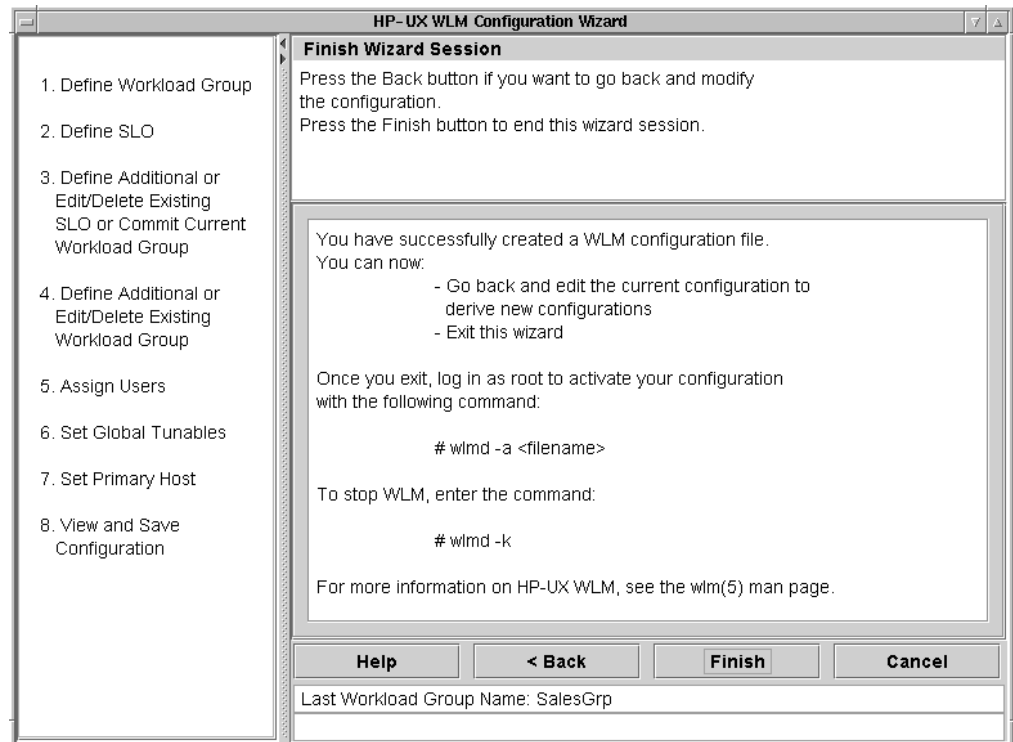
14. This next screen appears only if the system has HP-UX Virtual Partitions (vPars) or Node Partitions (nPars) installed. If you plan on managing vPars or nPars, see the section “Automatically resizing vPars” on page 36 or “Automatically resizing nPars using iCOD software” on page 37 for pointers to white papers that address partition management in more detail. Otherwise, select the **Next** button to continue.



15. Specify the full path and filename indicating where to save the configuration then select the **Next** button. In this example, the configuration will be saved as `/tmp/isolate_oracle_instances.wlm`.



16. Close the wizard by selecting the **Finish** button in the next screen.



17. Activate the new configuration file, instructing WLM to take control of resource management, using the following `wlmd` command:

```
# wlmd -a /tmp/isolate_oracle_instances.wlm
```

## Example WLM configurations

If you would like to try other configurations, WLM offers a number of example configuration files. Each file includes comments explaining the components used. The syntax used in these files is documented in the `wlmconf(4)` man page and in the HP-UX Workload Manager User's Guide.

The example configurations, some of which are described below, are available from <http://www.hp.com/go/wlm> and in `/opt/wlm/examples/wlmconf/`.

- `entitlement_per_process.wlm`  
A configuration file that demonstrates the use of a shares-per-metric goal. A workload group's allocation, or entitlement, is based directly on the number of currently active processes running in the group.
- `manual_entitlement.wlm`  
A configuration file to help a new WLM user characterize the behavior of a workload. The goal is to determine how a workload responds to a series of allocations (entitlements). For a similar configuration that changes the number of CPUs in the PSET upon which a workload group is based, see `/opt/wlm/toolkits/weblogic/config/manual_cpucount.wlm`.
- `metric_condition.wlm`  
Configuration file to illustrate that an SLO can be enabled based upon the value provided by a metric (in this case, the metric is provided by a glance data collector provided with the WLM product). Metrics can be used in both the `goal` statement and the `condition` statement of a single SLO.
- `stretch_goal.wlm`  
Example configuration file to demonstrate how to use multiple SLOs for the same workload (but at different priority levels) to specify a stretch goal for a workload. A stretch goal is one that we'd like to have met if all other higher-priority SLOs are being satisfied and there are additional CPU cycles available.

In addition, WLM has a number of free toolkits that simplify integrating WLM with various major applications. Each toolkit comes with example WLM configurations as well. Some of these toolkits help you use WLM with:

- Apache webserver
- iCOD (Instant Capacity on Demand)
- Oracle databases
- Pay Per Use
- SAS® Software
- HP-UX SNMP Agent
- BEA WebLogic Server™

For information on these toolkits and their configuration files, see the `wlmtk(5)` man page or <http://www.hp.com/go/wlm>.

# Alternative methods for placing processes in workload groups

The WLM configuration file allows you to place processes in workload groups based on the application name or the user running the application. Outside the configuration file, you can use the `prmrn` and `prmmove` utilities discussed below to manage process placement.

## `prmrn`: starting a process in a workload group

You can explicitly start a process in a particular workload group using the `prmrn` command. For a particular user, the groups the user can access are identified in the WLM configuration file's `prm` structure: the first group named after the user is the default group for processes the user starts; however, if any additional groups are named, the user can also run processes in those groups. For example, given the following `prm` structure:

```
prm {
    groups = OTHERS : 1,
           testers : 2,
           coders : 3,
           surfers : 4;

    users = moe : coders surfers,
           curly : coders surfers,
           larry : testers surfers;
}
```

user larry running the command

```
# my_really_big_job &
```

would cause his job to be run in group `testers`. However, user larry also has permission to run processes in the group `surfers`. Thus, larry can use the `prmrn` command below

```
# /opt/prm/bin/prmrn -g surfers my_really_big_job &
```

to run his process in the group `surfers`.

## `prmmove`: moving an existing process to a workload group

Use the `prmmove` command to move existing processes to a different workload group. If larry from the above example has a job running with process ID (PID) 4065 in the group `testers`, he could move that process to group `surfers` by running the command:

```
# /opt/prm/bin/prmmove surfers -p 4065
```

# Monitoring wlm

For monitoring WLM, there are several methods, some of which are described below.

## ps [-P] [-R *workload\_group*]

The `ps` command has options that are specific to Process Resource Manager, or PRM, which WLM uses to define workload groups:

- `-P`  
Adds the column PRMID, showing the workload group for each process.

```
# ps -P
      PRMID    PID TTY          TIME COMMAND
      SalesGrp  6462 tty1        4:34 report_maker
```

- `-R workload_group`  
Lists only the processes in the group named by *workload\_group*. Here is output showing processes in the workload group SalesGrp:

```
# ps -R SalesGrp
      PID TTY          TIME COMMAND
      6462 tty1        4:34 report_maker
      6533 tty1        6:41 sales_tracker
```

## wlminfo

The `wlminfo` command, available in `/opt/wlm/bin/`, displays information about SLOs, metrics, workload groups, vPars, nPars, and the current host. Here is example group information:

```
# /opt/wlm/bin/wlminfo group
Workload Group  PRMID  CPU Shares  CPU Util  Mem Shares  State
OTHERS         1      49.00      0.00      0.00      ON
FinanceGrp     2      15.00      0.00      0.00      ON
SalesGrp       3      31.00     19.57      0.00      ON
MarketingGrp   4       5.00      0.00      0.00      ON
```

Based on the output, the only group with any activity is SalesGrp.

The `wlminfo` utility also has a graphical interface. To use this interface, use the `-i` option:

```
# wlminfo -i
```

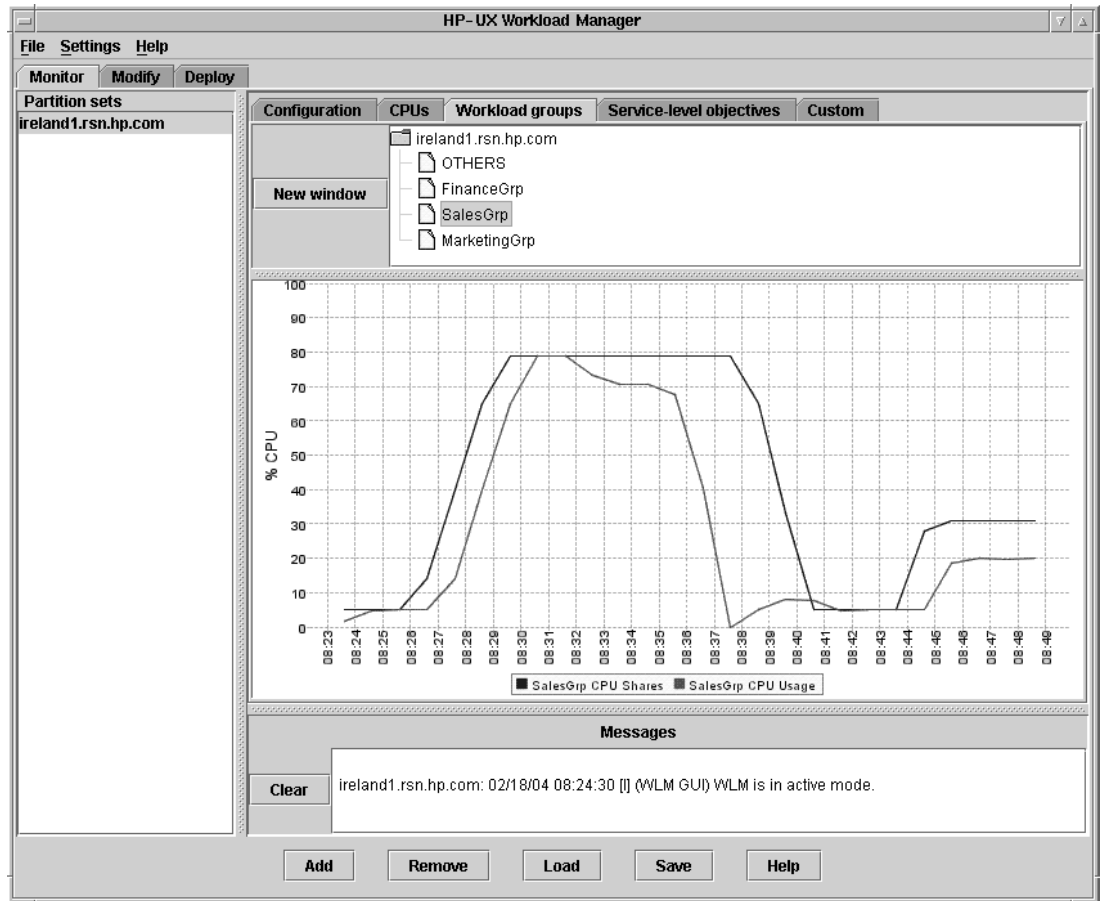
This interface is also accessible from the `wlmgui` command.

Selecting the Monitor tab, we can see graphs on resource allocation for one or more workload groups. By default, "CPU Shares" and "CPU Usage" are graphed for the selected groups.



The following figure shows how much CPU is allocated to SalesGrp (the top line), as well as how much CPU the group is actually using (the bottom line). Because a usage goal is affecting the group's CPU allocation, the group gets additional CPU when it uses more than a certain percentage of its current allocation. Similarly, when the group is using less than a certain percentage of its allocation, WLM removes CPU resources from the group. The allocation is capped at about 80 due to allocations to the other groups.

**Figure 2.** Usage and shares allocation for SalesGrp



## GlancePlus

If you have the optional HP product GlancePlus installed, you can also use it to see data on WLM workload groups. In GlancePlus though, the workload groups are known as PRM groups.

## Status and statistics logs

WLM provides the following logs:

- `/var/opt/wlm/msglog`  
Always created; provides ongoing messages about WLM operations
- `/var/opt/wlm/wlmdstats`  
Created when you use the `wlmd -l` option; provides statistics on resource allocation and usage, SLO compliance, and metrics

When you configure WLM's vPar management or nPar management, you can also enable logging of vPar or nPar statistics in the file `/var/opt/wlm/wlmpardstats`.

## Stopping WLM

With WLM running, stop it by logging in as root and running the command:

```
# /opt/wlm/bin/wlmd -k
```

## For more information

For more information on HP-UX Workload Manager:

- Browse the online help available in the WLM configuration wizard
- See the `wlm(5)` man page, which provides an overview of WLM as well as pointers to all the other WLM man pages
- See the HP-UX Workload Manager User's Guide, available on systems with WLM installed at `/opt/wlm/share/doc/WLMug.pdf` and from the web at <http://www.docs.hp.com/hpux/netsys/>
- Contact any of our worldwide sales offices or visit our web site at: <http://www.hp.com/go/wlm>

To learn more about the adaptive enterprise and virtualization, please visit <http://www.hp.com/go/virtualization>.

© 2004 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Itanium is a trademark or registered trademark of Intel Corporation in the U.S. and other countries and is used under license.

02/2004

