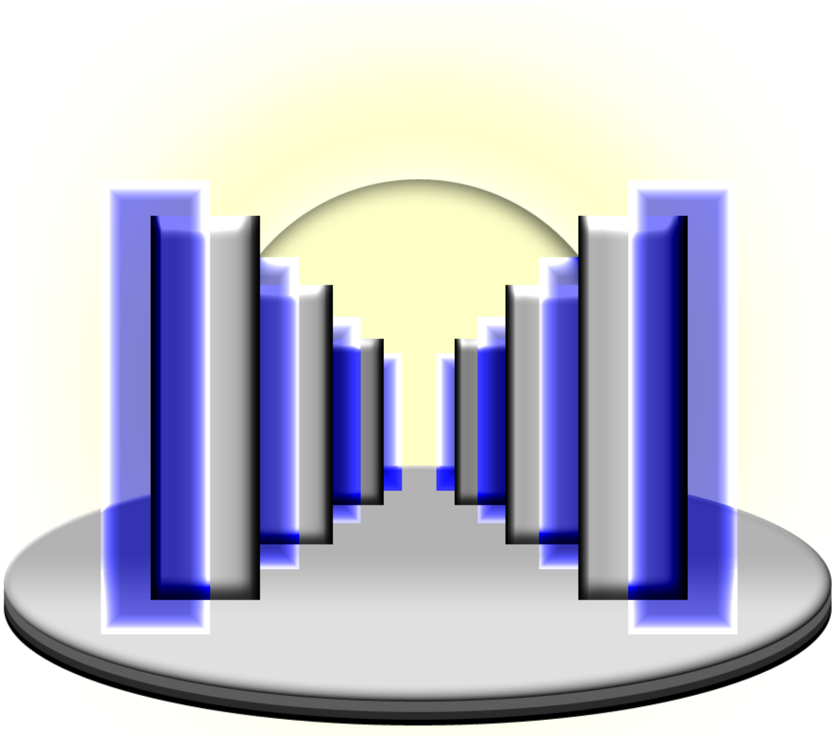
Parallel Computing Manager



Version 1.7

Code and manual written by Dean E. Dauger, Ph. D.

Table of Contents

I. Installing

Make sure your Macintosh running OS X 10.3.9 or later is connected properly to the Internet. Almost any type of network connection (10BaseT, 100BaseT, Gigabit, Airport, or a mix) is sufficient. If this Mac is on an isolated network, manually configure the TCP/IP control panel to a unique IP address from 192.168.1.1 to 192.168.1.254. To install PCM:

1. Copy PCM to your hard drive.
2. Open PCM.

So that your Mac will be available for computation if rebooted, we recommend the following:

3. From the File menu, select "Set PCM in Account Startup Items".

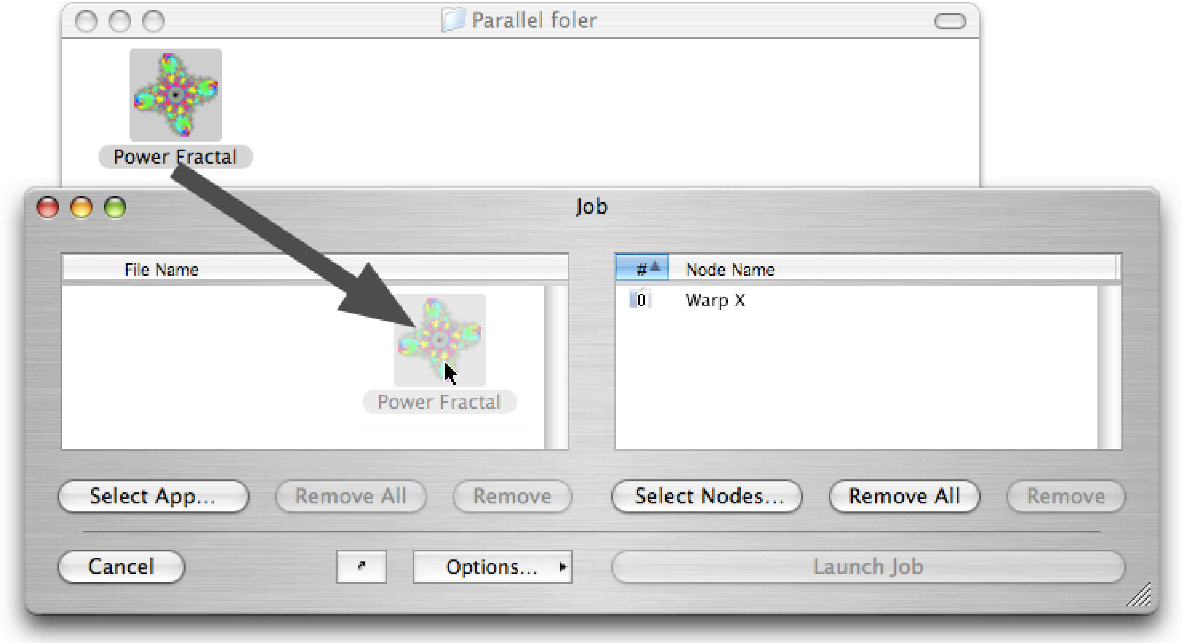
That's it! Your Mac is now ready for parallel computing. Repeat for additional Macs.

II. Your First Parallel Computation

Once PCM is installed and running on each Mac of a local area network, you can use them as a parallel computer. You will need a parallel application to run. For our first test, let’s use the Power Fractal app. Run it on your machine first and note its performance.

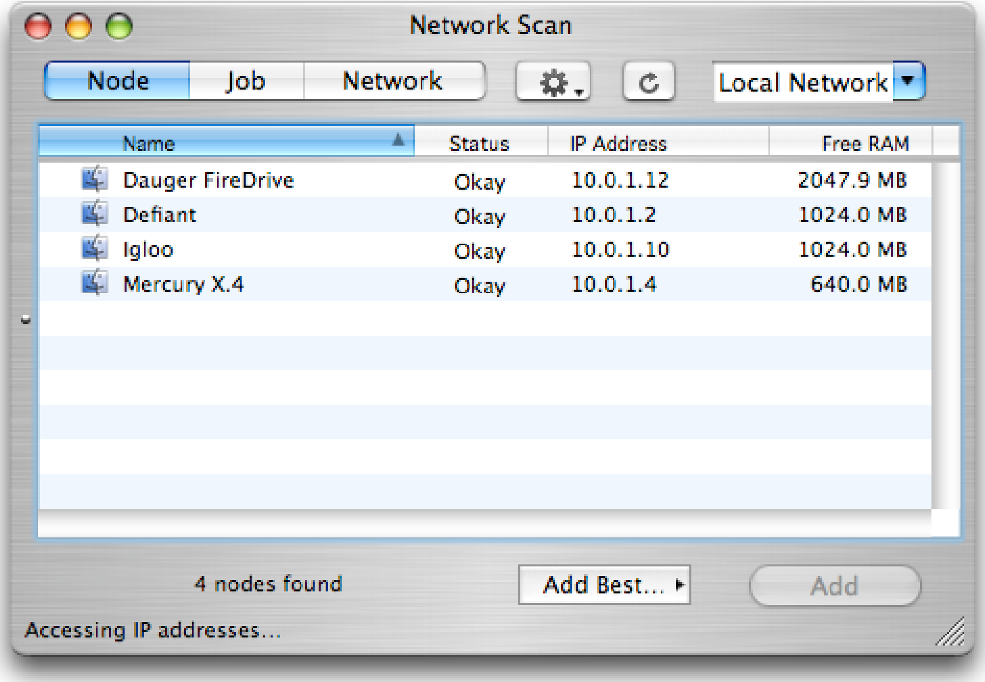
Step One - Selecting a Parallel Application

Switch to PCM and select **New Job...** from the File menu, which results in a new Job Window. Drag the Power Fractal app from the Finder to the Job Window or click **Select App...**.



Step Two - Selecting Nodes

By default, PCM selects your own node as participating in the parallel job. To add other nodes, click on **Select Nodes...** in the Job Window to invoke the Network Scan Window.



Name	Status	IP Address	Free RAM
Dauger FireDrive	Okay	10.0.1.12	2047.9 MB
Defiant	Okay	10.0.1.2	1024.0 MB
Igloo	Okay	10.0.1.10	1024.0 MB
Mercury X.4	Okay	10.0.1.4	640.0 MB

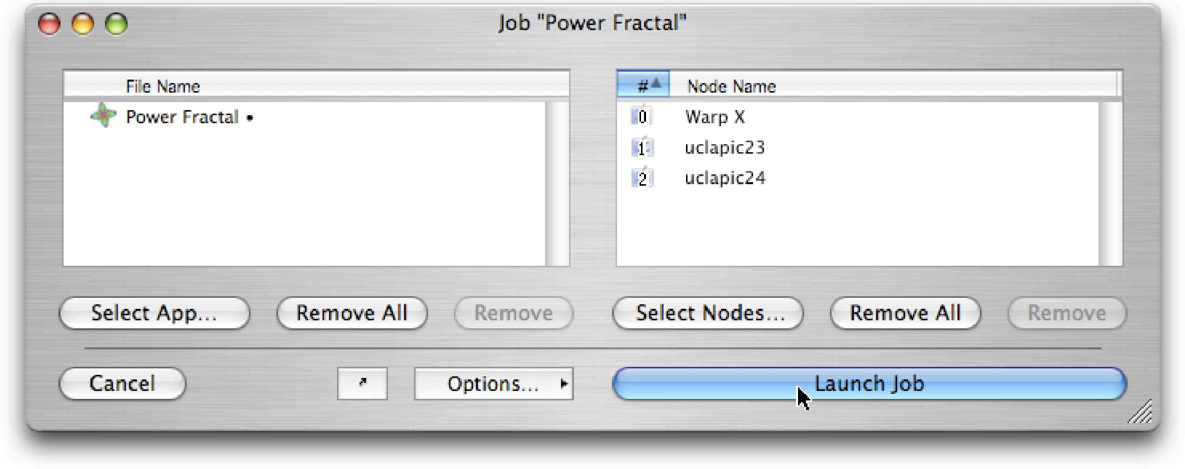
4 nodes found

Accessing IP addresses...

Double-clicking on a node moves it to the node list of the Job Window.

Step Three - Launching the Parallel Job

Finally, click **Launch Job** to start your parallel job.



File Name
Power Fractal •

#	Node Name
0	Warp X
1	uclapic23
2	uclapic24

PCM should now be distributing the code and starting the parallel application. Once the job runs, you should see a measurable performance increase.

Congratulations! You are now operating your first parallel computer.

Optional - Your Second Parallel Computation

1. Open Power Fractal. Note its performance.
2. From the Parallel menu, select "Automatically Launch onto Four Nodes".

The demo will now automatically ask PCM to create a job using the four best nodes on the local network, then submit itself for parallel launch. This is an example of "Computational Grid" computing.

III. Introduction

What is the Parallel Computing Manager?

PCM serves as both the user interface to your parallel computer and the manager of your parallel computer. Cluster and grid computing are both examples of parallel computation. PCM is the parallel computer management and user interface software that can prepare, launch, and monitor these types of parallel computing jobs.

The first role of PCM is to automate the process of starting a parallel application. A parallel application is much like a normal application except that it is designed to run on a collection of intercommunicating computational nodes.[1] Manually starting such an application is a tedious process. PCM automates this process, minimizing the user's effort to operate and control a parallel computation. However, PCM provides much more beyond this fundamental function.

PCM provides access to a great deal of information, customization, and automation regarding the life of jobs run on the cluster and the health of the cluster. Jobs can be launched immediately or queued for a later time or when specific conditions are met. PCM can intelligently and dynamically acquire nodes using its heuristic algorithms or using custom AppleScripts prior to launch. Statistical information about nodes can be accessed, and some node behavior can be customized. Job status is monitored, and statistics about jobs are collected and retained during and after jobs have completed. PCM serves as launcher, scheduler, queuing system, and multisystem

[1] PCM can recognize and launch a wide variety of applications, including Classic apps, CFM-based Carbon apps, bundle-based Carbon apps, Cocoa apps, Mach-O executables, Unix scripts, AppleScript apps, and Universal Applications.

cluster monitor, all presented in a friendly user interface.

The first thing that makes PCM unique from all other software of its kind on any platform is the extension of the revered Macintosh ease-of-use to parallel computing. Since 1998, the AppleSeed project at UCLA's Department of Physics has unquestionably demonstrated the practicality of leveraging the advantages of the Macintosh platform for the goals of numerically-intensive parallel computing. While others used Linux and Windows NT with a hodgepodge of command-line only, unreliable, unstable code fragments, AppleSeed provided the only known solution for parallel computing on the Mac OS. Users of software built by the AppleSeed team are *not* required to know the inner workings of an operating system, nor are they required to hire expert assistance in such matters. Demonstrated at UCLA and around the world time and time again for three years, that advantage has enabled college students, high school and junior high school students, and even ordinary scientists on shoestring budgets to successfully build and operate their very own parallel computer.

Macintosh Parallel Computing

Forming a parallel computer out of a network of computers is called "cluster computing". In this case, the hardware of the parallel computer is a cluster of Macintoshes. AppleSeed provided two major software pieces essential for using such hardware for parallel computation.

The first is a communications library named MacMPI, authored by Dr. Viktor K. Decyk and introduced in 1998. MacMPI provides a means for executables on each Macintosh to communicate with one another. It is a source code wrapper library that translates the calls by the parallel code into calls directly to the Mac OS. MPI (Message-Passing Interface) is a standard application programming interface (API) of the high-performance parallel computing industry. That API is supported on almost all large parallel computers (Cray, IBM SP, Fujitsu, SGI, etc.). Using the Mac OS, MacMPI supports a commonly used subset of those MPI calls.

The second software component satisfies a different essential need of the parallel computation: initiation. To begin, MacMPI requires information about the Macintoshes designated to participate in the computation. In addition, since, when compiled, it is mixed with the object code of the executable itself, MacMPI cannot copy and start the parallel application or start itself on all the Macintoshes that will perform the computation. These needs are filled by parallel computing launching software.

Through 2000, AppleSeed provided software called the Launch Den Mother and the Launch Puppy (LDM & LP) that filled this second need. The combination of MacMPI and LDM & LP are the key ingredients that allowed students and researchers around the world to develop and run their own parallel computing software on the Mac OS.

Meanwhile, Apple was introducing prereleases of Mac OS X in preparation for its final release in 2001. Mac OS X was fundamentally distinct from the original Mac OS while being the future of the Mac OS. Apple required Mac developers to port their code to Carbon, a new API, so that their code can run natively in OS X. Carbonizing MacMPI (renamed MacMPI_X) was straightforward, but the launching mechanism was another story. Most of LDM & LP's fundamental design made it impossible for a simple conversion to OS X. For example, LDM & LP's primary communications mechanisms and LDM's user interface were tied to pieces of the Mac OS not carried forward into Mac OS X.

Since "porting" LDM & LP to OS X would largely be a rewrite of LDM & LP, their author decided instead to simply start from scratch. Rather than a reconstruction of what came before, the new software created the opportunity to create software that vastly superseded LDM & LP's functions, capabilities, and use of the Macintosh user interface. Utilizing three years of experience with parallel computation using Macintosh clusters at UCLA Physics, this project called for a rethinking and redesign of the user interface for a parallel computer.

Using PCM

PCM is meant to be used in an environment where your cluster is a computational resource shared by a cooperative group, much like how a workplace printer is shared. A few scenarios are possible:

- A dedicated Macintosh cluster - Perhaps the simplest configuration, a group of eight, sixteen, or thirty-two identical Macintoshes or Xserves devoted to computation at all hours of the day. PCM enables members of your group to submit jobs to the cluster and retrieve the output at a later time. Also, a subset of the nodes can be used for one job while a second job runs on the others. Many, including UCLA's Department of Statistics and NASA's Jet Propulsion Laboratory, have clusters like this.

- Reusing a network of individual Macs for computations - There are secretaries, scientists, artists, and professors using their own Macs for e-mail, spreadsheets, web browsing, and word processing during the day. After they leave for home, the Macs are usually idle and unused. Merely with the addition of PCM, this network of Macs can become a computational cluster at night or on weekends.

- Combinations of dedicated and individual Macs - A few Macs, perhaps four at a time, can be added to an existing network of Macs as described in the preceding example. The groups of four devoted computational nodes can be used for debugging and diagnosing an experimental parallel code during the day, but then the entire network can be combined in the off-hours for larger (and well-tested) parallel codes. As new hardware is purchased, the compute nodes are mixed with the rest of the population as needed. This more dynamic model is in practice in the Plasma Physics group at UCLA. This setup is well suited for a circumstance where, often, the groups of four are enough for most computations, but occasionally a member of the Plasma Physics group needs to complete some large runs to meet, for example, a conference deadline. So this researcher asks the group and gets time on the entire network. This flexibility has saved the day for more than one researcher's work.

- Be creative: You may find additional scenarios useful. For example, from a PowerBook connected to the Internet via Airport, the author has initiated and controlled, in real-time, parallel jobs computed over 40 miles away. Using a network connection at the Max-Planck-Institut für extraterrestrische Physik in Garching, Germany, the distance record has increased to 6000 miles.

System Requirements

In general, PCM needs 4 MB of free memory on a Macintosh with a valid TCP/IP connection. Because PCM is a Universal application, it can run on Intel Macs too. PCM requires OS X 10.3.9 or later.

IV. Menus and Windows

PCM Menus

File menu

□ **New Job...** - Opens a new Job Window (see the Job Window section)

□ **New Queue Job...** - Opens a new Job Window enabling the Queue Job and Automatically Acquire options.

□ **Save Job As...** - Saves the job of a Job Window to a file on disk, which can later be used for another job.

□ **Select App and Files...** - Opens a get files dialog for selecting an application and files for the Job Window. If the Job Window is present, the file dialog will appear as a sheet. If the Job Window is not present, the dialog will be standalone, but if a file is selected it will be added to a new created Job Window.

□ **Reveal Remote Files Folder in Finder** - When PCM receives files from another node, PCM must hold the files in a folder on the hard drive. Invoking this item directs the Finder to open that folder where those files are being held. See

Collect Remote Files. Also, this item is available by control-clicking the PCM icon in the Dock.

□ **Select Nodes...** - Opens the Node View of the Network Scan Window (see the Network Scan Window section)

□ **Show Job View...** - Opens the Job View of the Network Scan Window (see the Network Scan Window section)

□ **Recent Node Lists** - When the Job Window is open, this submenu lists the node lists of the most recently launched jobs. The number before the ":" is the number of nodes. When the node names have similarities, PCM attempts to abbreviate the names. For example, a node list of four computers named node01, node02, node26, and node27, will be abbreviated to "4: node01, …02, …26, …27".

□ **Recent Apps and Files** - When the Job Window is open, this submenu lists the file lists of the most recently launched jobs. The number before the ":" is the number of files. If one of the files is an executable that was launched, it will be listed first.

□ **Set PCM in Account Startup Items** - Makes (or updates) an entry for PCM in the Login Items system preference so that PCM will automatically start up and make this Mac available for parallel computation after a restart. This feature is provided as a convenience.

□ **Collect Remote Files In** - When PCM receives files from another node, it must store those files in a folder on the hard drive. PCM can hold those files either: 1. in the folder where PCM itself resides; 2. inside the Temporary Items folder, as designated by the OS; 3. in the path /tmp/pcm, which is the default case on OS X; or 4. in the path /Users/Shared/pcm/remote. Note that files in /tmp may be deleted when the machine is restarted.

□ **Open PCM Log** - Opens the PCM Log in Console.app. PCM logs events into a text file located at /Users/Shared/pcm/pcmlog.txt. These entries detail steps in the launch process, the queuing system, and network access.

□ **Quit PCM** - Selecting this item, or pressing Command-Option-Q, will quit PCM. PCM is an unusual application because, while its user interface is not in use, it is meant to run at all times in the background. This mechanism will make it less likely to be quit accidentally.

□ **Finder to the Foreground** - Selecting this item sends Finder to the foreground.

Edit menu

□ **Paste** - Invoking this item in the Job Window pastes an item on the clipboard into its file pane. Launching such a job distributes the clipboard to the clipboards on the target nodes. In the Network Scan Window, this pastes into the remote network address field.

Node menu

To use the items on the Node menu, you most select one node from either the node list of the Job Window or the list of nodes in the Node View of the Network Scan Window. Contextual menu-clicking on a single node in the Job Window or the Network Scan Window also invokes this menu.

□ **Get Node Info** - Invokes a Node Info Window that retrieves a variety of information about a node, such as processor type, OS version, load, and free memory. It also retrieves a list of running processes (an application is one type of process) on that node.

□ **Get Job Queue** - Retrieves the parallel jobs queued on that node. Typically, a job is forwarded to the PCM at the job's node zero and held there until certain conditions have developed. For example, a job might start only when its designated start time has passed and the target nodes are no longer busy.

□ **Get Files** - Opens a new window where you can retrieve files from the remote node.

□ **Connect to Server...** - Opens an Apple File Sharing connection to that node, much like the menu item of the same name in the Finder. If that node has File Sharing on, you can enter your password to access files on its volumes. This feature is provided for your convenience.

Network menu

□ **Register this Node** - PCM, by default, registers its node on the network, meaning that it puts up a flag that other nodes can see. This flag has enough information for other nodes to identify and communicate with this particular Macintosh. Selecting "Never" removes this flag, making this node invisible to other nodes. Selecting "On a Schedule…" opens a node registration schedule dialog (see below). PCM will then register and deregister itself according to the specified schedule.

□ **Local Scan Includes this Node** - When scanning for nodes in the local network, selecting **Always** will cause PCM to always add itself to the network results, regardless of the Register this Node settings. Otherwise your node might not appear in the local network scan if your PCM did not register itself. Selecting **Always** does not cause other nodes to see your node.

□ **On Node Deregistration** - When PCM deregisters itself, PCM will kill all jobs it is tracking on this node if **Kill All Jobs** is checked. For example, when this node deregisters itself according to the node registration dialog (see "On a Schedule…" of the "Register this Node" submenu, above), PCM will kill any remaining jobs that it launched. If this menu item is not checked, PCM will allow jobs to continue to their own completion.

□ **Primary Address** - This submenu specifies which active IP address PCM registers as "primary" or the default address. Use System Default has PCM follow the precedence specified by the system, usually set via the port ordering in the Network system preference. Setting this to another IP address, if available, resets PCM's network registration of this node to the new setting. It will encourage, but not guarantee, that other nodes and their jobs will use that address, perhaps because of its more optimal performance. PCM will still listen on the other addresses, as not every IP address will be accessible from other nodes. Selecting the Import Addresses… option selects a whitespace-delimited text file containing additional addresses that identify this node. This is appropriate when, for example, a firewall redirects activity from a public IP address to a private one.

□ **Network Scan Columns** - This submenu allows you to show and hide columns in the Network Scan Window. Check and uncheck the items to toggle them. See the Network Scan Window section for descriptions. Selecting "Show Columns…" opens a window where multiple columns can be toggled at once.

□ **Automatic Rescan** - When the Network Scan Window is open, PCM will automatically repeat its scans on the network at a particular interval. You can set this time interval, or shut off the automatic rescan, by selecting an item on this menu.

□ **Job View Access Time** - This setting determines how far back in time the Network Scan Window, when in Job View, will query when compiling archived job information.

□ **Discover using** - Since its birth, PCM uses both Apple's implementation of the Service Location Protocol (SLP), an IETF standard, and Bonjour (formerly Rendezvous), also known as ZeroConfig or mDNS, to perform discovery of other nodes on a TCP/IP network. For best compatibility, the default setting uses both SLP and Bonjour. The benefit of using only Bonjour is faster discovery.

□ **Access Nodes** - Normally PCM contacts other nodes directly to inquire about their status. A node accessed using the **Remote Node Scan** feature could return the addresses of nodes behind a firewall or other network barriers. Selecting **via Proxy** will use the proxy connection provided by the first remote node to forward data between your local machine and those behind the firewall. Only on OS X 10.4 "Tiger", PCM can access an Xgrid cluster. To access Xgrid, enter the host address of the Xgrid controller and its password using the **via Xgrid…** item of this submenu. If no password is required, leave that field blank. PCM spawns jobs containing a piece of itself through Xgrid, which PCM can then discover and access.

□ **Remote Network Job Access** - Normally PCM on other nodes contacts yours directly to forward jobs and files. For example, a node accessed using the **Remote Node Scan** feature might not be able to return files to your nodes because of firewalls or other network barriers. This feature permits the Network Scan Window to pull these files onto your machine. Typically this is used in combination with the **via Proxy** setting of the previous item. **None** turns off this feature, **Only My Jobs** accesses only jobs that resulted from a job from this node and user, typically a job triggered by **Retrieve Output** job option (see Options of the Job Window), while **All Jobs** will pull all jobs destined for this node. If the name and IP address of your node changes, then this feature might not successfully locate the jobs.

Settings menu

□ **New Job Automatically Reloads** - When the Job Window is created, it will remember information, such as the file list (including the application), the node list, and various job options, from the last parallel job submitted. This submenu toggles what portions of the last job it will retain and what it will discard. Note: some of this information may become stale, such as when you move the app on your hard drive or if a node has a dynamic IP address (which could happen if your network uses DHCP). Selecting **On Job Submit** triggers a new job to invoke as soon as the previous job is submitted to the launch queue.

□ **Remember Recent Node Lists** - This setting determines how many node lists will be displayed in the "Recent Node Lists" submenu of the File menu.

□ **Remember Recent Apps and Files** - This setting determines how many lists of apps and files will be displayed in the "Recent Apps and Files" submenu of the File menu.

□ **Job Port Number Range** - When PCM launches a job it pseudorandomly assigns a primary TCP port number to the job for its communications. Normally PCM chooses from 16384 up to 49152. You may specify a different range of port numbers here, say, for compatibility with a firewall. The **Use Port Number...** option in the Job Window overrides this setting.

□ **Launch Unix Jobs** - By selecting **in the Background**, Unix-based executables (such as Mach-O executables and Unix scripts) are launched in the background, with output directed to a stdout.txt file. Selecting **using Terminal** directs PCM to launch Unix-based executables in a window of the Terminal app, allowing for user-input. This setting is local to this node: the launch on the remote nodes will use Terminal instead of background launching if **using Terminal** is checked on those nodes as well.

□ **Restrict/Allow Access to Local Processes** - When this item is set to restrict, PCM will block other nodes from seeing process IDs of processes not launched by PCM. This prevents someone else from using the Node Info window to kill, say, your word processor. Use this item only if you wish to restrict control of processes on this node by other nodes.

□ **Accept Commands from** - When set to **Local Subnet**, PCM will check the IP address of incoming connections against its local subnet. If they do not match, the connection will be rejected and this PCM's listener port will reset. Selecting **Import Addresses...** allows you to select a whitespace-delimited text file containing a list of IP addresses. PCM will compare the addresses of incoming connections against this list, including the resolution of DNS data. Setting to **Any Address** removes this restriction. Use this item only if you do not want to control this node from other subnets of the Internet.

□ **Reject/Accept Remote Preferences Command** - When set to accept, other nodes will be allowed to overwrite this PCM's preferences and settings. For example, the node registration schedule may be modified remotely. When set to reject, other nodes will not be allowed to modify settings of this PCM.

□ **Disable/Enable PCM at Logout...** - Selecting this item will direct PCM to run while nobody is logged in. This feature makes it possible to use unutilized, logged-out machines for parallel applications. PCM automatically quits when someone logs in and will return when the user logs out or if they run PCM. Toggling this item may require administrative authorization. The words "(To Be Enabled On Restart)" will be appended if it is necessary to restart the machine to fully enable this feature.

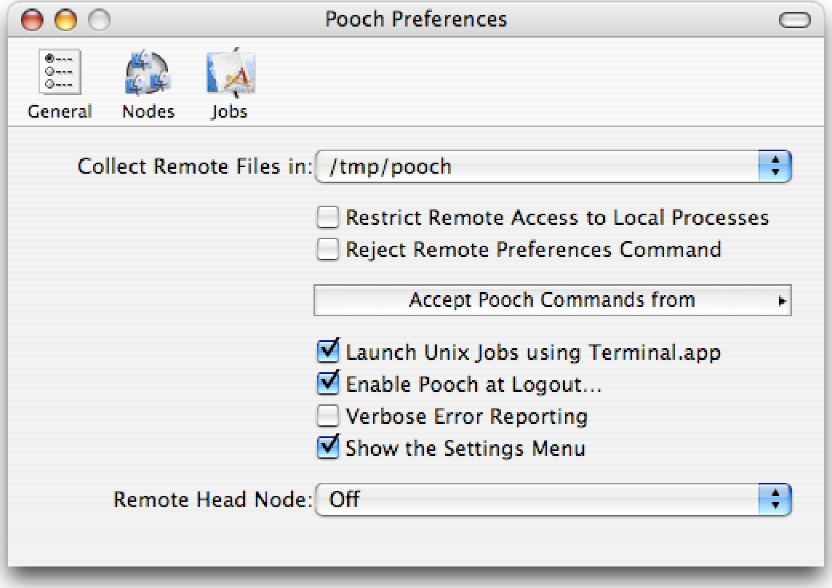
□ **Disable/Enable Verbose Error Reporting** - Reports internal errors, such as network errors, in notification windows. The information reported most useful when diagnosing difficult-to-find problems during PCM operation. It also reports a problem when PCM encounters AppleScript objects or commands that it cannot recognize.

PCM menu

□ **About PCM...** - displays the About Box of PCM, which also can contain status messages. This window normally opens briefly, then closes on its own, at startup. Clicking on the window closes it.

□ **Preferences...** - displays the PCM Preferences window. This window displays all of the preferences available in PCM. These settings are divided into three categories: General, Nodes, and Jobs.

General Preference Pane



Collect Remote Files in: /tmp/pooch

□ Restrict Remote Access to Local Processes
□ Reject Remote Preferences Command

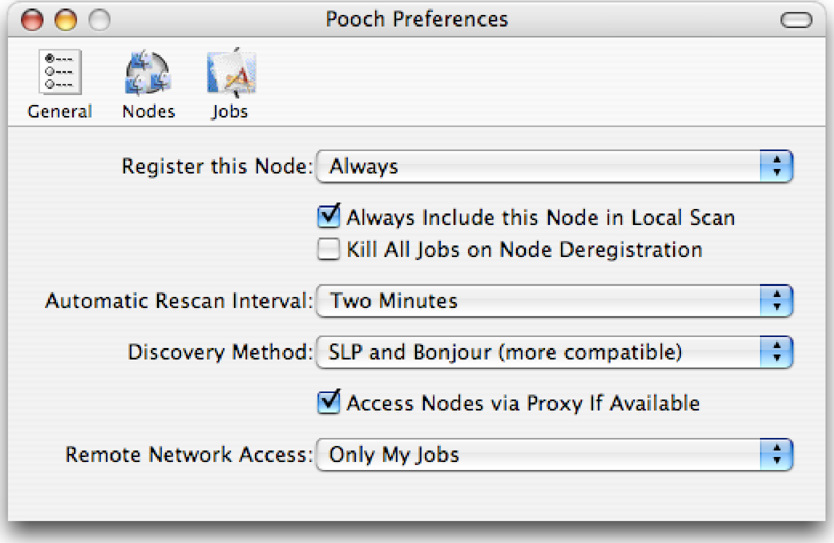
☑ Launch Unix Jobs using Terminal.app
☑ Enable Pooch at Logout...
□ Verbose Error Reporting
☑ Show the Settings Menu

Remote Head Node: Off

Clicking on the icons in the toolbar at the top of the window switches the Preference Window to the other panes. It is normal for this window to disappear when PCM is in the background. For backwards compatibility, the Settings Menu is shown by default.

Remote Head Node - This preference modifies and coordinates four elements of PCM's behavior to support launching jobs into and retrieving files from a cluster with a designated head node. If this feature is enabled, the default Network Scan window scans the head node's network instead of the Local network, the Forward via job option chooses the head node, the Retrieve Output and Queue Job options are enabled, the **Access Nodes via Proxy** setting is enabled, and the **Automatically Acquire Nodes** job option automatically selects the head node as the starting node. New head node IP addresses can be entered via the address field of the Network Scan window.

Nodes Preference Pane



Register this Node: Always

☑ Always Include this Node in Local Scan

☐ Kill All Jobs on Node Deregistration

Automatic Rescan Interval: Two Minutes

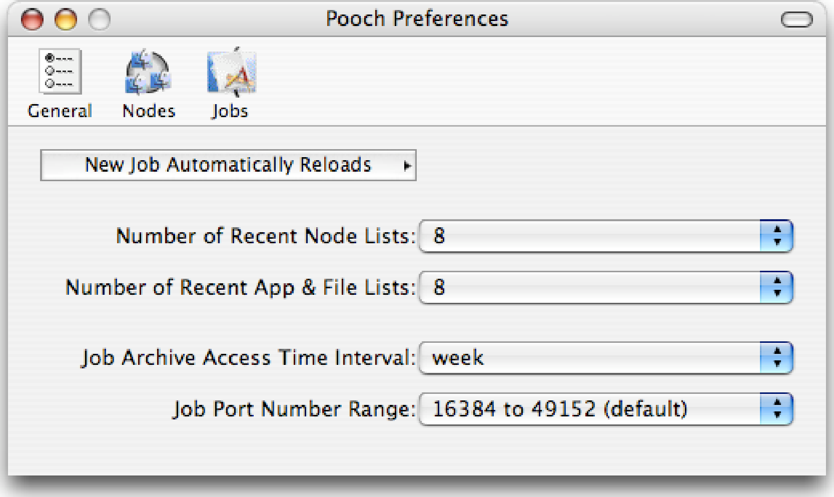
Discovery Method: SLP and Bonjour (more compatible)

☑ Access Nodes via Proxy If Available

Remote Network Access: Only My Jobs

Remote Network Job Access - Normally PCM on other nodes contacts yours directly to forward jobs and files. For example, a node accessed using the **Remote Node Scan** feature might not be able to return files to your nodes because of firewalls or other network barriers. This feature permits the Network Scan Window to pull these files onto your machine. Typically this is used in combination with the **via Proxy** setting of the previous item. **None** turns off this feature, **Only My Jobs** accesses only jobs that resulted from a job from this node and user, typically a job triggered by **Retrieve Output** job option (see Options of the Job Window), while **All Jobs** will pull all jobs destined for this node. If the name and IP address of your node changes, then this feature might not successfully locate the jobs.

Jobs Preference Pane



Number of Recent Node Lists: 8

Number of Recent App & File Lists: 8

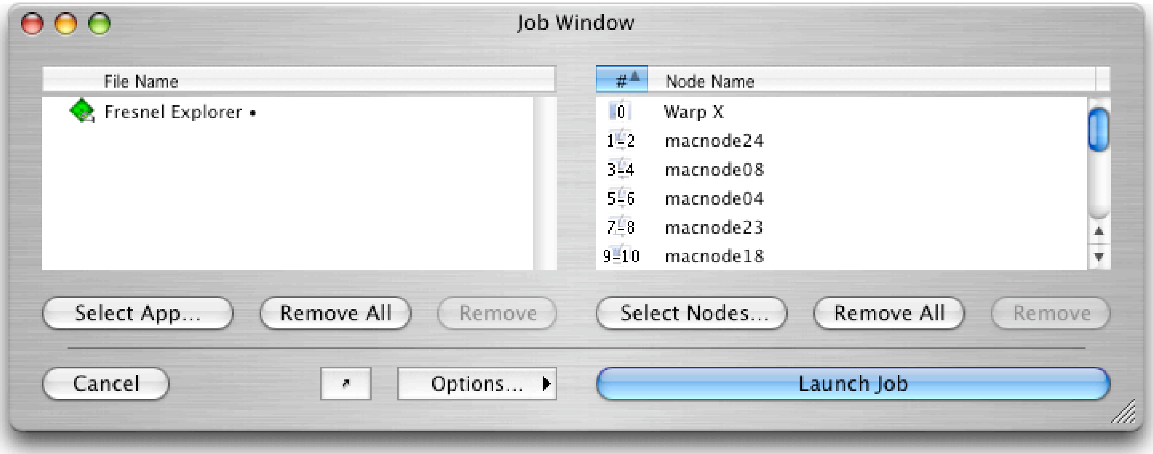
Job Archive Access Time Interval: week

Job Port Number Range: 16384 to 49152 (default)

Selecting **On Job Submit** of the **New Job Automatically Reloads** menu triggers a new job to invoke as soon as the previous job is submitted to the launch queue.

Windows

Job Window



File Name
Fresnel Explorer •

#	Node Name
0	Warp X
1-2	macnode24
3-4	macnode08
5-6	macnode04
7-8	macnode23
9-10	macnode18

The Job Window is where all parallel jobs are prepared for launching, scheduling, or queuing. A job requires information about the parallel application you want executed and what nodes will run this application. This window is resizable so it can display large amounts of information.

- □ File List - The left pane of this window displays the list of files in this job. This list can include an application[2] , which, if present, is placed at the top of the list. Additional applications dragged here will be added to the end of the list, but they will not be launched by PCM. Removing the first application in this list will cause the next application, if present, to shift to the top of the list. Additional files may be added, which some parallel apps use for input data, here. PCM will distribute these files to the nodes on launch. PCM is also capable of selecting and copying folder structures[3] to other nodes. The folder tree will be scanned recursively.

[4] For more about the different types of parallel computing, see http://daugerresearch.com/vault/parallelzoology.html

[3] Unix permissions will be preserved when copying between OS X machines.

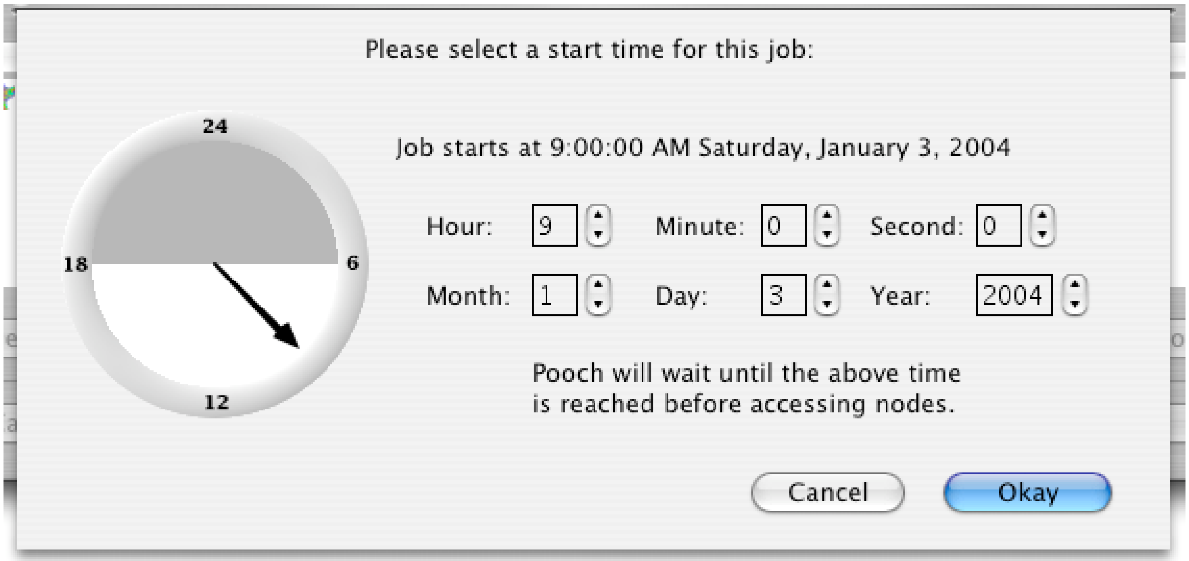
Files inside these folders will be copied as well, but only those files present at the moment of launch. Aliases will not be resolved. You may use the **Select App...** button to invoke a dialog sheet to select an application to execute and files to distribute, or you may drag them from Finder to this pane if you prefer. Use the **Remove All** button to clear the file list, and select a file or files and click the **Remove** button to remove individual files.

□ Node List - The right pane displays the list of nodes for this job. Each node you select is assigned to run an MPI task or tasks. Each task is given a unique identification number. The number(s) on the icon to the left of the node's name indicates which task(s) that node will be assigned. (See the Tasks per Computer job option, below.) This information will be relayed to MacMPI. The parallel code can retrieve this information from MacMPI and use it to partition its problem among the processors. Experimenting with the order of these nodes or the number of tasks can be useful for debugging. Clicking on **Select Nodes...** invokes the Network Scan Window. You can add nodes to the node list by selecting them in the Network Scan Window or dragging them from that window to this one. **Remove All** clears the node list, and selecting a node or nodes and clicking **Remove** removes individual nodes from the list. The limit on the number of nodes is determined when this PCM is registered. Note: PCM mandates that your Mac is node zero if it is participating in the computation.

□ Job Options - In OS X 10.3 and later, clicking the **Options...** button produces a drawer presenting options for this job. Holding down the **Options...** pop-up creates a menu containing the same options. Only the menu is available on previous OS versions.

◇ **Place in Subdirectory...** - When sending files to another Mac, PCM by default places all apps and files within the same folder in which PCM resides. Over the course of many jobs, the mixture of files can become confusing. Specifying a name here asks PCM to create a subdirectory inside that PCM's folder and place all apps and files in that new subdirectory. (PCM does not write outside its folder during the launch process.) For the purposes of organization, you can name these subdirectories based on the job name, or the date, or the name of the user who submitted the job.

◇ **Start Job at Time...** - By default, when you submit a job, PCM will start the job on the specified nodes immediately. This option, however, allows you to have PCM place the job in the job queue and delay its start until a particular future time, any minute, any day. The job will be held in the queue on the node designated zero in the node list. (So, if your Mac is not node zero, it will be forwarded to that node.) The following shows the dialog that appears when selecting a start time:



Please select a start time for this job:

Job starts at 9:00:00 AM Saturday, January 3, 2004

Hour: 9 Minute: 0 Second: 0

Month: 1 Day: 3 Year: 2004

Pooch will wait until the above time
is reached before accessing nodes.

◇ **Tasks per Computer** - Many of the recent Power Macintoshes have more than one processor each, so it would be appropriate to be able to allocate one parallel processing task per processor, enabling parallel computing inside boxes as well as between boxes. By default, this option is set to do just that. You may override this setting by selecting a specific number of tasks per processor. Increasing the number of tasks beyond the number of nodes will probably be less efficient, but this possibility is useful for debugging parallel code. Control-clicking on the node icon in the node list will override this setting, allowing you to select different task counts for each node.

◇ **Job Type** - This option specifies the type of parallel launch needed for this parallel job. MacMPI, mpich, MPI-Pro, mpich-gm, and LAM/MPI need information about the cluster configuration, but they each require significantly different data formats and launch procedures. The MacMPI launch type is the default. It is generally the case that only Unix-based executables can accept information the way that the other MPIs require it, so Carbon, Cocoa, and most other types of apps will usually follow the MacMPI convention. See below for the Grid Job Type.

◇ **Command–Line Arguments...** - This option specifies command-line arguments to add when launching Unix-style Mach-O executables. Only some Unix-based executables require input information using such options (such as "-a -e iou"), and it is usually preferable to supply such input information via input files for portability. This item will be disabled when the file list contains other executable types.

◇ **Use Port Number…** - In order to use TCP/IP, an MPI needs to use a particular TCP port number to initialize its network connections. Under normal circumstances, PCM generates a pseudorandom port number based on the job name and the time of day and supplies this number to the MPI. You can use this job option to override that random value and set a particular port number. This feature can be useful to comply with firewalls that only have specific ports open. However, the TCP convention requires that the same port number not be used again for approximately two minutes. Some TCP implementations follow this convention, and some do not, so your network could appear to be blocked if you launch a second job using the same port number too soon.

◇ **Forward via** - Normally PCM attempts to contact all nodes directly when launching jobs and distributing data. In some configurations, nodes of a cluster could be protected by a firewall or a machine acting as a firewall, such as a “head node”, which prevents this kind of access. This feature allows you to forwarding the job to an “in-between” node, which acts as a proxy. The job will be forwarded to the proxy node, which then forwards it to node 0 in your node list. Usually such proxy nodes should be chosen from the ones used for the Remote Node Scan.

◇ **Retrieve Output** - Turning this setting on directs PCM to automatically retrieve files created and modified by a job after PCM detects that the job has ended. PCM checks for this condition once per minute, and the queuing system also operates on a one-minute cycle. This feature can be set to only send back the files generated at node 0 or on all the nodes. PCM will make a “best effort” to place files in the directory on the node where the executable originated. If that node is not available for some reason, the forwarding process will be queued until it is able to complete the task. As needed, the files will be forwarded through the proxy node specified in the **Forward via** feature, above. The retrieved files are accompanied by a .joblog text file containing useful information about the job's execution.

◇ **Sound** - PCM will sound when launching a job if this item is checked.

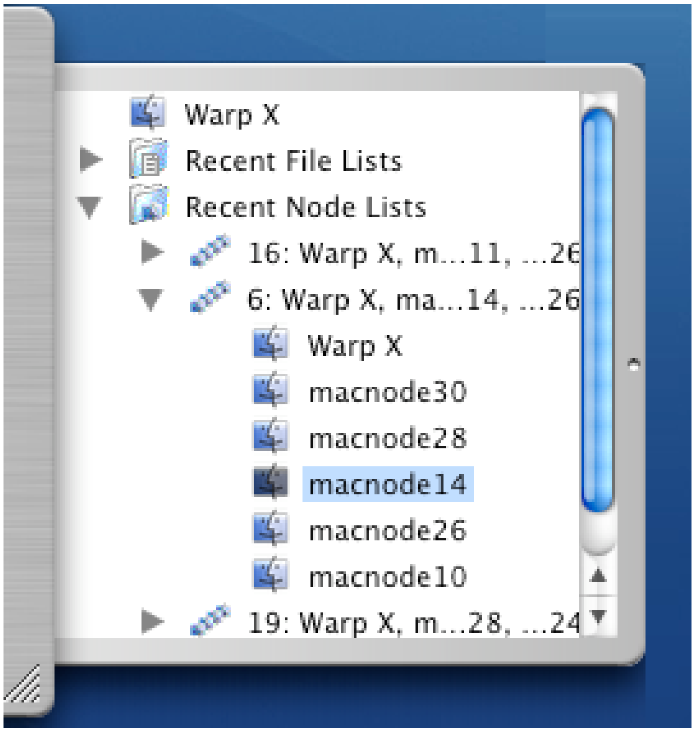
◇ **Don't Use Busy Nodes** - Normally, PCM checks if the nodes on the node list are already running a job submitted using PCM. If any of the nodes are running such a job, PCM will not launch the new job. This feature is useful because a parallel job usually runs best if there are no other jobs running. Unchecking this item disables this feature.

◇ **Queue this Job** - If this option is enabled, PCM will save this job into a job queue and launch it as soon as possible. If the job launch process fails, which can happen if the nodes are busy or inaccessible, the job will be saved back into the job queue of the node from which the launch was attempted. PCM will check the queue once per minute. (See the Node Info Window or the Job View of the Network Scan Window to access this queue.) Otherwise, PCM discards the job after the first attempt. If you are using the **Start Job at Time...** feature, you may want to consider checking this flag in case a different job is already running on the same nodes.

◇ **Automatically Acquire Nodes** - If this option is enabled, the Job Window will replace the node list with options to automatically acquire the nodes needed to run this job. PCM will attempt to fulfill the request specified in the acquire controls just prior to launch. Used in conjunction with the **Start Job at Time...** option, the acquisition will be scheduled for that specified time. Used in conjunction with the **Queue this Job** option, the acquisition will be attempted again if the request was not satisfied.

□ Launching, Scheduling, or Queuing a parallel job - Clicking on **Launch/Schedule/Queue Job** will ask PCM to launch, schedule, or queue, depending on the **Start Job at Time...** and **Queue this Job** options, a parallel application onto the nodes according to the information given in this window. If your node is participating in the parallel computation, your machine's PCM will direct the launch process or hold the job in its queue, depending on the job and the circumstances. If your machine is not participating in the computation, the job will be forwarded to the PCM at node zero of the computation, and that PCM becomes responsible for the job. If the job has not yet launched, you may inquire about this job's status by accessing the job queue of the job's node zero or using Job View in the Network Scan Window. After the job successfully launches, the nodes running the parallel job will report BUSY and the job will report Running in the Status column of the Network Scan Window until the job ends. Getting Info about a node will return additional details about the computation's use of that node.

□ Recent Items Drawer - Clicking on the recent button at the bottom of the Job Window will open a new drawer containing recent items. You can use this feature to recall recently used executables, files, or nodes.

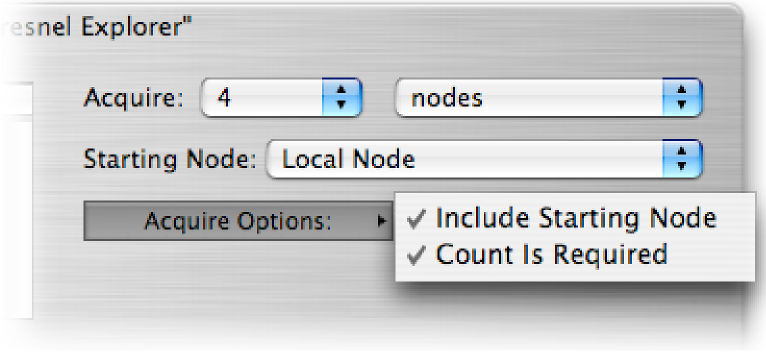


The Recent File Lists folder contains the file lists of previously submitted jobs, each of which can be opened to access specific files. The Recent Node Lists folder contains the node lists of those previous submitted jobs, which can also be opened to recall specific nodes. The Remote Scan List displays the cached IP addresses used in the Remote Node Scan feature of the Network Scan Window. For your convenience, an icon at the top of this drawer contains the name and IP address information of your node. This drawer is available only in OS X 10.2 or later.

Variations of the Job window

The Job window offers two variations from the node list. The first is the ability to acquire nodes automatically from the cluster at launch time. This feature allows the executable to be launched on nodes as they become available. The second is the ability to distribute a single-processor executable on a cluster, giving them a different integer for each instance of the executable. This feature gives you the ability to explore a parameter space using an executable not otherwise designed for parallel execution.

Automatically Acquire Nodes



Acquire: 4 nodes

Starting Node: Local Node

Acquire Options:

✓ Include Starting Node

✓ Count Is Required

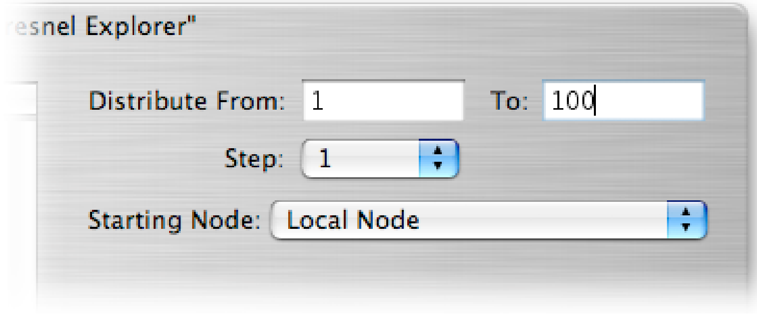
□ Acquire nodes or processors - When the **Automatically Acquire Nodes** option is enabled, selecting from these pop-up menus specifies how many nodes or processors PCM will attempt to acquire just before launching the job. If available, nodes of different processor counts could be combined into a parallel job. Options such as **Tasks per Computer**, do apply. PCM will add as many available local nodes as it needs to satisfy this request.

□ Starting Node of the acquisition - When the **Automatically Acquire Nodes** option is enabled, the acquisition is performed by the node selected here. By default, the local node is selected. If another node is selected, the job will be forwarded to that node prior to launch. This pop-up menu is populated with the nodes previously used for the remote network scan feature of the Network Scan window. Adding a node from the Network Scan window to the Job window also sets this acquire setting.

□ Acquire Options… - When the **Automatically Acquire Nodes** option is enabled, the acquisition will include the starting node if the **Include Starting Node** item is checked. Otherwise, the starting node may or may not be included. The acquisition depends on the number, status, and rating of the nodes on the network. The starting node can be selected if it helps satisfy the acquisition request. If the **Count Is Required** item is checked, the processor or node acquisition request becomes a minimum requirement, without which the job will not be launched and, if **Queue this Job** is selected, will be queued for a later attempt. In that case, the acquisition attempt will be repeated until these requirements are satisfied.

Grid Job Type

The Grid job type allows you distribute single-processor tasks on a cluster automatically, exercising a subset of parallel computing called "distributed computing",[4] well-known in brute-force key breaking and SETI@Home. This feature implicitly activates the Queue this Job, Automatically Acquire Nodes, and Retrieve Output options because its operation requires these capabilities. Launching this job creates subfolders beginning with "case" where the original executable resides and numbered with the integer assigned to that job. The output of the executables will be returned to these folders. Use the Subdirectory job option to customize these folder names.



esnel Explorer"

Distribute From: 1 To: 100

Step: 1

Starting Node: Local Node

[4] For more about the different types of parallel computing, see http://daugerresearch.com/vault/parallelzoology.html

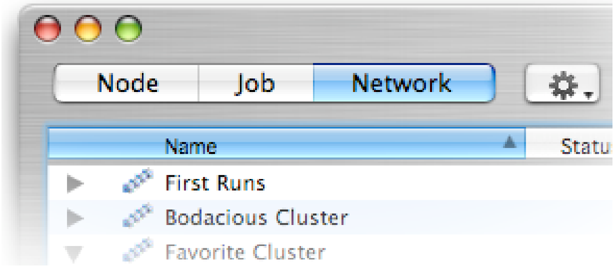
□ Distribute From: To: - When the **Grid** job type is selected, these entries specify the range of integers PCM will use for each instance of execution. The Job in the above figure shows that it will launch cases using every integer from 1 to 100, inclusive. PCM will create a text file named "input" in the same directory as the executable containing that integer. By default, PCM will also append that integer to the command line of the executable. To change that setting, use the Command-Line Arguments option. The %index argument will be replaced with the integer for that case.

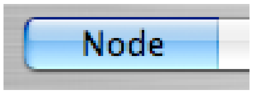
□ Step - With the **Grid** job type is selected, this pop-up menu specifies the step between cases. PCM will start with the From entry and increase the integer by the Step selection. So, if 7 were selected above, PCM will launch the executable with the integers 1, 8, 15, 22, and so on until case 99.

□ Starting Node - Similar to the Starting Node selection of the the **Automatically Acquire Nodes** option, the acquisition of nodes will be performed by the node selected here.

Network Scan Window

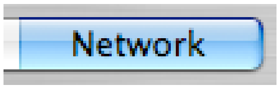
The Network Scan window features three different views, toggled via the segmented view control in its upper left corner.





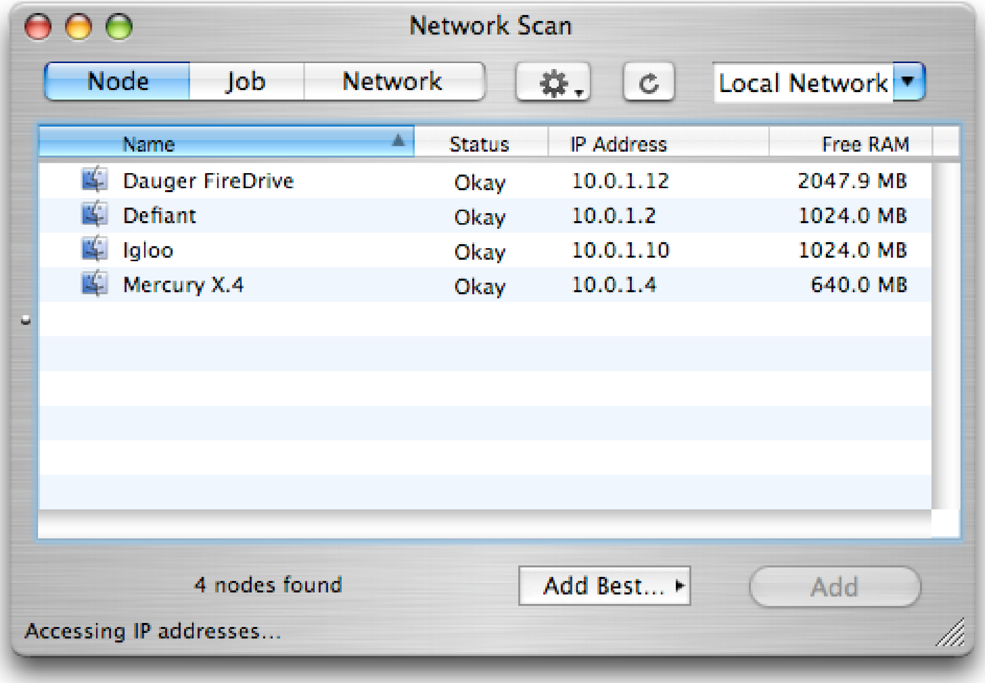
The Node View displays the nodes discovered on the network, along with diagnostics and statistics about these nodes. This view is used to access nodes, query their status and health, and select them for parallel computational jobs.

The Job View retrieves from those nodes data about jobs that are queued, launching, running, or terminated. This view compiles data from these sources to display the condition and other statistics about parallel computing jobs.



The Network view displays nodes in saved node lists, as well as remotely scanned networks in one large view. Each network of nodes can be revealed by clicking on the node list's or network's disclosure triangle.

Node View



Name	Status	IP Address	Free RAM
Dauger FireDrive	Okay	10.0.1.12	2047.9 MB
Defiant	Okay	10.0.1.2	1024.0 MB
Igloo	Okay	10.0.1.10	1024.0 MB
Mercury X.4	Okay	10.0.1.4	640.0 MB

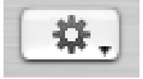
4 nodes found

Accessing IP addresses...

When this window is in Node View, PCM scans the network for available nodes and displays its results here. Once PCM has finished scanning, you can use this window to select nodes for your computation. Double-clicking on a node moves it to the node list of the Job Window, which will open if not already present. Clicking the **Add** button while a node is selected will have the same effect. Nodes also present in the Job Window will appear in gray in the Network Scan Window.

Selecting a number from the pop-up menu of **Add Best** moves that many nodes that PCM considers “best” to the Job Window. The selection excludes nodes that are already on the Job Window’s node list. The rating system PCM uses to rank the nodes is described fully in the Network Scan Columns section below. This mechanism is designed to be more likely to select nodes that perform better than others and be less likely to select nodes that are busy or heavily loaded. Selecting the highest number from this menu selects all nodes that PCM can communicate with and are not busy with a parallel job.

Normally, the Network Scan Window will scan the network again according to the **Automatic Rescan** specified in the Network menu. The **Refresh** button directs PCM to rescan and refresh the list immediately. This window is resizable for large amounts of information.



- After selecting one node, you can use this pop-up button to get information about that node. The **Get Node Info** and **Get Job Queue** items invoke a Node Info Window. The **Get Files** items invokes a Finder-like window from which you can drag files to the Finder.

- The Search box at the top right of the Network Scan window accepts a search string to limit the displayed data. It compares the search string against the node and job data available in the list below and displays only those items that match.   For example, if you type "G5", it will show nodes that use a G5 processor. If you type "dual", only those with two processors will appear. Other text will be compared against the names and other data to reveal appropriate matches. As with any search using text, overly specific search strings will yield little or no data, and overly general strings will produce excess.

- The address box at the top middle of the window provides access to remote clusters of nodes, something like the address box in a web browser. PCM normally can only directly scan within its own local network neighborhood. This is the default scanning process. You may always return PCM to this mode by selecting **Local Network** from this pop-up menu or typing "Local" in the box. Technically, unless your network administrators have set up your routers to support IP multicasting, this means that PCM can only see other nodes in the same IP subnet. (This limitation is inherited from the Internet-standard Service Location Protocol implemented in Apple's Network Services Location Manager and Apple's Bonjour implementation.)

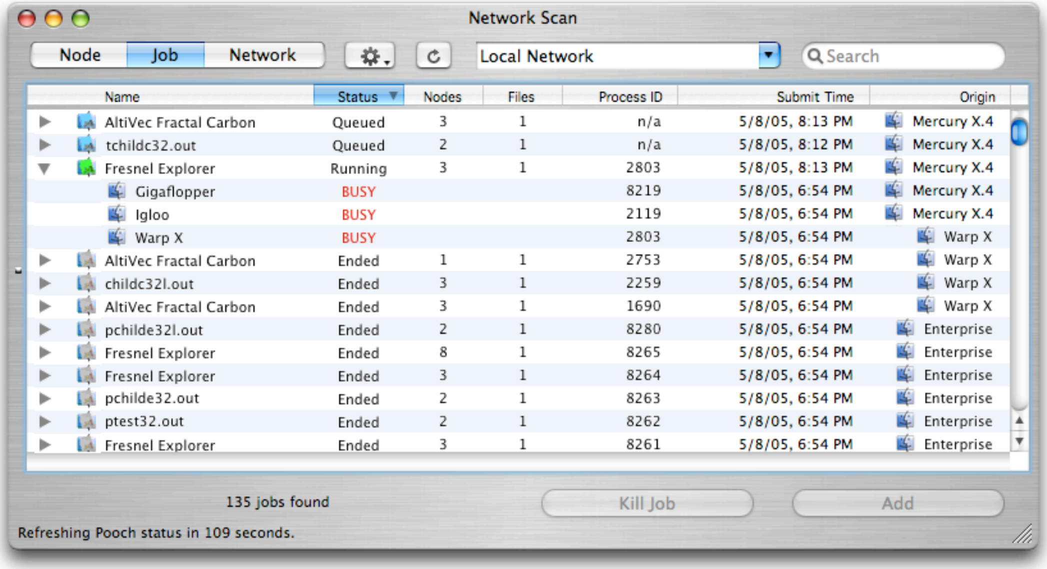
Entering an IP address in this box allows you to circumvent this limitation. Enter the address, in DNS (e.g., mymac.mydomain.com) or "dotted quad" (e.g., 192.1.2.34) form, of a Mac you know is running PCM. Your local PCM will then ask the far away PCM to do its local scan and relay the results. Essentially, the PCM on your Mac "borrows the nose" of the other PCM. This feature allows you to scan for nodes on any other accessible subnet. This is useful for circumstances where, say, you want to access a chem.ucla.edu Mac from a physics.ucla.edu Mac. Selecting **Local Network** returns the network search to your local network.

Once you enter one address, PCM will remember that address and make it available on the menu of the pop-up button for future use. As you accumulate more addresses, PCM will re-sort the list in order of most recently accessed first.

The functionality of this feature extends well beyond just one domain, however. From your Mac, you may use this to access nodes on any subnet *anywhere on the Internet*. This means you can even access your cluster running PCM from your home dial-up connection.

This feature is very powerful. With this power comes responsibility. There are security features in PCM that will make unauthorized access without PCM extraordinarily unlikely. These issues are discussed in the Security section.

Job View



Name	Status	Nodes	Files	Process ID	Submit Time	Origin
▶ AltiVec Fractal Carbon	Queued	3	1	n/a	5/8/05, 8:13 PM	Mercury X.4
▶ tchildc32.out	Queued	2	1	n/a	5/8/05, 8:12 PM	Mercury X.4
▼ Fresnel Explorer	Running	3	1	2803	5/8/05, 8:13 PM	Mercury X.4
Gigaflopper	BUSY			8219	5/8/05, 6:54 PM	Mercury X.4
Igloo	BUSY			2119	5/8/05, 6:54 PM	Mercury X.4
Warp X	BUSY			2803	5/8/05, 6:54 PM	Warp X
▶ AltiVec Fractal Carbon	Ended	1	1	2753	5/8/05, 6:54 PM	Warp X
▶ childc32l.out	Ended	3	1	2259	5/8/05, 6:54 PM	Warp X
▶ AltiVec Fractal Carbon	Ended	3	1	1690	5/8/05, 6:54 PM	Warp X
▶ pchilde32l.out	Ended	2	1	8280	5/8/05, 6:54 PM	Enterprise
▶ Fresnel Explorer	Ended	8	1	8265	5/8/05, 6:54 PM	Enterprise
▶ Fresnel Explorer	Ended	3	1	8264	5/8/05, 6:54 PM	Enterprise
▶ pchilde32.out	Ended	2	1	8263	5/8/05, 6:54 PM	Enterprise
▶ ptest32.out	Ended	2	1	8262	5/8/05, 6:54 PM	Enterprise
▶ Fresnel Explorer	Ended	3	1	8261	5/8/05, 6:54 PM	Enterprise

135 jobs found

Refreshing Pooch status in 109 seconds.

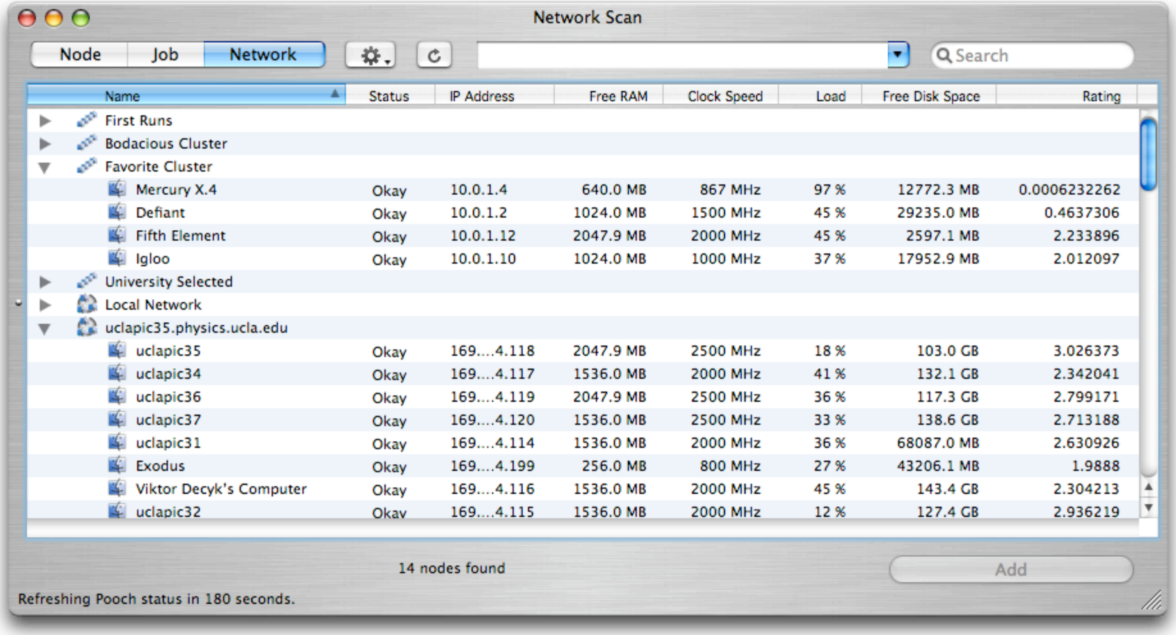
When the Network Scan window is in Job View, it scans the network for nodes and accesses data, if present, about queued, launching, running, terminated, and aborted jobs and compiles that job data for display here. The root level of the list shows all the jobs, named according to their executable, found on the network. The job icon for each job is color coded as follows:

Icon	Color	Meaning
⬜	white	A normal parallel computing job
🟦	blue	A queued job ("on ice")
🟨	yellow	The job is being launched
🟩	green	The job is running
▩	gray	The job has ended
☒	white X on black	The job was aborted

Opening the disclosure triangle displays the nodes that will be included in, are being used for, or were included in that job, depending on the state of the job. If possible, the information about a particular node will be queried directly, otherwise the data is derived from the data associated with the job.

This view correlates and compiles this data from all the nodes on the network. A variety of otherwise independent jobs may be accessed as far back as the **Job View Access Time** preference will permit. Since nodes used in one job can be reused for another one later, nodes will naturally appear more than once in multiple jobs. Although information such as the job ID, number of nodes and files, origin, and submission time should be consistent between nodes of a job, the process ID, start time, and end time could be different. If certain nodes of a job are not available, data about a node will be derived from the job data, if possible, residing on the other nodes. The Last Access and Determined Via columns are easy ways to tell whether or not the data are fresh.

Network View



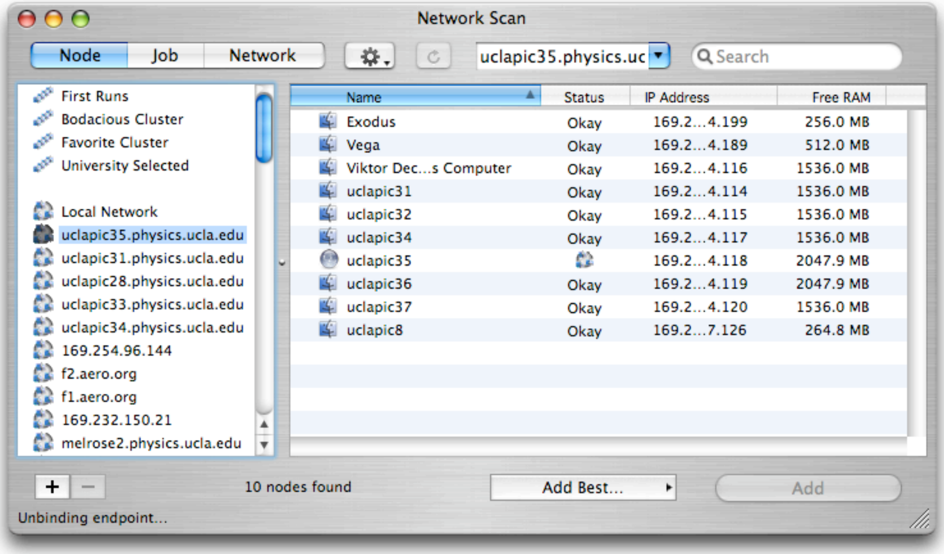
Name	Status	IP Address	Free RAM	Clock Speed	Load	Free Disk Space	Rating
▶ First Runs							
▶ Bodacious Cluster							
▼ Favorite Cluster							
Mercury X.4	Okay	10.0.1.4	640.0 MB	867 MHz	97 %	12772.3 MB	0.0006232262
Defiant	Okay	10.0.1.2	1024.0 MB	1500 MHz	45 %	29235.0 MB	0.4637306
Fifth Element	Okay	10.0.1.12	2047.9 MB	2000 MHz	45 %	2597.1 MB	2.233896
Igloo	Okay	10.0.1.10	1024.0 MB	1000 MHz	37 %	17952.9 MB	2.012097
▶ University Selected							
▶ Local Network							
▼ uclapic35.physics.ucla.edu							
uclapic35	Okay	169....4.118	2047.9 MB	2500 MHz	18 %	103.0 GB	3.026373
uclapic34	Okay	169....4.117	1536.0 MB	2000 MHz	41 %	132.1 GB	2.342041
uclapic36	Okay	169....4.119	2047.9 MB	2500 MHz	36 %	117.3 GB	2.799171
uclapic37	Okay	169....4.120	1536.0 MB	2500 MHz	33 %	138.6 GB	2.713188
uclapic31	Okay	169....4.114	1536.0 MB	2000 MHz	36 %	68087.0 MB	2.630926
Exodus	Okay	169....4.199	256.0 MB	800 MHz	27 %	43206.1 MB	1.9888
Viktor Decyk's Computer	Okay	169....4.116	1536.0 MB	2000 MHz	45 %	143.4 GB	2.304213
uclapic32	Okay	169....4.115	1536.0 MB	2000 MHz	12 %	127.4 GB	2.936219

14 nodes found

Refreshing Pooch status in 180 seconds.

In the Network View, the window organizes nodes into saved node lists followed by remote networks. Opening the disclosure triangle of a node lists displays the nodes it contains and includes those nodes for access. Opening a remote network triggers PCM to access the remote node and query the nodes in its local area network, just like when using the address box as in the Node View. However, what is different is that you can have multiple networks open and accessible at once, and multiple saved node lists open as well. You may mix and match nodes from these multiple networks when selecting nodes for your parallel computing job.

Network Pane



First Runs
Bodacious Cluster
Favorite Cluster
University Selected

Local Network
uclapic35.physics.ucla.edu
uclapic31.physics.ucla.edu
uclapic28.physics.ucla.edu
uclapic33.physics.ucla.edu
uclapic34.physics.ucla.edu
169.254.96.144
f2.aero.org
f1.aero.org
169.232.150.21
melrose2.physics.ucla.edu

Name	Status	IP Address	Free RAM
Exodus	Okay	169.2...4.199	256.0 MB
Vega	Okay	169.2...4.189	512.0 MB
Viktor Dec...s Computer	Okay	169.2...4.116	1536.0 MB
uclapic31	Okay	169.2...4.114	1536.0 MB
uclapic32	Okay	169.2...4.115	1536.0 MB
uclapic34	Okay	169.2...4.117	1536.0 MB
uclapic35		169.2...4.118	2047.9 MB
uclapic36	Okay	169.2...4.119	2047.9 MB
uclapic37	Okay	169.2...4.120	1536.0 MB
uclapic8	Okay	169.2...7.126	264.8 MB

10 nodes found

Unbinding endpoint...

When the Network Scan window is initially opened, a pane splitter appears on the far left side of the window. Dragging that to the right opens a network pane, listing both saved node lists and remote networks. The remote networks shown are the same that was entered via the address box earlier, and they list networks in the same order, with the Local Network listed first. This pane gives another way to quickly select and scan a remote network, much like the left pane in OS X's Finder windows.

Clicking on the plus sign below the Network Pane creates a new saved node lists. You may think of these as iTunes playlists, but for nodes. You may drag nodes from any network to these saved node lists, creating arbitrary groups of nodes useful for different kinds of parallel computing jobs. For example, you may have a group that you know is only available at night, or another set of nodes that are all G5s or dual-processor machines, but are on disparate, mixed networks or clusters. This gives you a way to group commonly used nodes together, allowing you to conveniently view, scan, and monitor these nodes as if they were together and include them in a new job in the Job window. Double-click a node list to scan them. When viewing a saved node list, select a node and press the delete key to remove them from a node list.

Network Scan Columns

The Network Scan Window displays information about each node PCM queries. This information is shown in a spreadsheet-like format. This window is invoked using the columns button at the top of the Network Scan Window or the **Show Columns…** item of the **Network Scan Columns** submenu of the Network menu. The columns can also be added or removed using the **Network Scan Columns** submenu of the Network menu. You can drag a column at its header and resize it by clicking on its right edge. Clicking on a column header re-sorts the list according to that column.

Sometimes, if the node is running a computational intensive task, PCM on that node may not be able to respond immediately, delaying the appearance of this information. In that case, it probably should not be used for a new job.

Column	Function
IP Address	the “dotted quad” Internet Protocol address of this node
Machine Type	the machine type, based on its ROMs
CPU Type	the processor type (e.g., G3 or G4), including number of processors

Clock Speed

the processor clock cycle speed in millions of cycles per second

OS Version

the version of the operating system (e.g., 9.0.4 or 10.0)

Load

the estimated load on the system in %. This item is calculated using the results of a call to the Unix “ps” command. Since the resolution is rather low and the process of measurement can influence the measurement (like Quantum Mechanics), this number can be somewhat inaccurate. It is intended, if the number is measurably nonzero, to hint that that this node is probably in use and should be avoided.

Free Memory

the amount of contiguous free RAM available on this machine, which is the largest available space for a newly launched application.

Free Disk Space

the amount of free space on the drive on which this node’s PCM resides

Achieved Performance

measured achieved single-precision floating-point performance according to a benchmark based on code from the Power Fractal app. Automatically recognizes the Velocity Engine and SSE, and updated once per day. (not multiprocessor aware)

Peak Single-Precision

measured single-precision floating-point performance of the FPU of this processor based on an IBM benchmark. Updated once per day. (not multiprocessor aware)

Peak Double-Precision

measured double-precision floating-point performance of the FPU of this processor based on an IBM benchmark. Updated once per day. (not multiprocessor aware)

Peak Vector Float

measured single-precision vector floating-point (`vector float` or `vFloat`) performance of the vector FPU (in the Velocity Engine or SSE) of this processor based on an IBM benchmark. Updated once per day. (not multiprocessor aware)

User Idle Time

the time that has elapsed since PCM last detected interaction from the user, such as moving the mouse or other activity

Rating

a nonnegative real number describing PCM's rating of that node. This number is a function of the information PCM has collected. If the node is busy, or PCM has not successfully communicated with the node, the rating is zero. Otherwise, it is related to the above properties according to the following heuristic function:

$$f = ln(1+(1-l)\,p)\frac{2}{\pi}\tan^{-1}(m)\frac{2}{\pi}\tan^{-1}(d)\frac{2}{\pi}\tan^{-1}(u)$$

where p is Achieved Performance divided by 300 MFlops, l is the load (from 0 to 1), m is Free Memory divided by 32 MB, d is Free Disk Space divided by 16 MB, and u is User Idle Time divided by 15 minutes. f is designed to be very low when the node is undesirable to use, such as when it is heavily loaded or is running out of memory or disk space. Note that, assuming load and memory space are non-issues, f is approximately logarithmic with processor performance. Also, many of these variables can easily change at any moment, causing the rating to fluctuate. This rating function may change in future versions of PCM. (To create and use your own rating, use the

AppleScript interface described in Chapter V.)

Jobs Queued the number of jobs that are queued on that particular node

Current Job Name if the node is running a job, the name of the first parallel executable it is running

Last Access the time and date this data was accessed, according to that node’s clock

PCM Version the version of PCM that node is running

Job ID the job identification number assigned to that job

Node Count the number of nodes that job will use, is using, or had used

File Count the number of files supplied for that job, as submitted to PCM

Process ID the process identification number of a particular node’s instance of the executable of a job when it is running or a job when it was running

Submission Time the time and date this job was originally submitted

Start Time the time and date a particular node’s instance of the executable of a job was first determined to be running

End Time the time and date a particular node’s instance of the executable of a job was first determined to be ended

Elapsed Time the amount of time a particular node’s instance of the executable of a job took that node

Job Origin the node that was the origin of the job

Determined Via the node that provided the information about this job or node

Finally, the Status column has a number of different possible displays.

Status Reading	Meaning
→●	attempting to connect to this node
▸	a connection was established this node and security authorization is being approved. If the node is under heavy load, PCM might not be getting enough CPU time to respond.
◗	access this node is in progress
◕	access this node is almost done
Okay	this node was successfully accessed and is not busy
BUSY	this node was successfully accessed and is busy running an application launched by PCM
((🔄))	("breathing") this node is being used for remote neighborhood access and is returning addresses of and/or accessing other nodes
📊	retrieving job data
🗂	retrieving file and folder data

ℹ exchanging data or commands

Queued this job was successfully queued and is awaiting launch

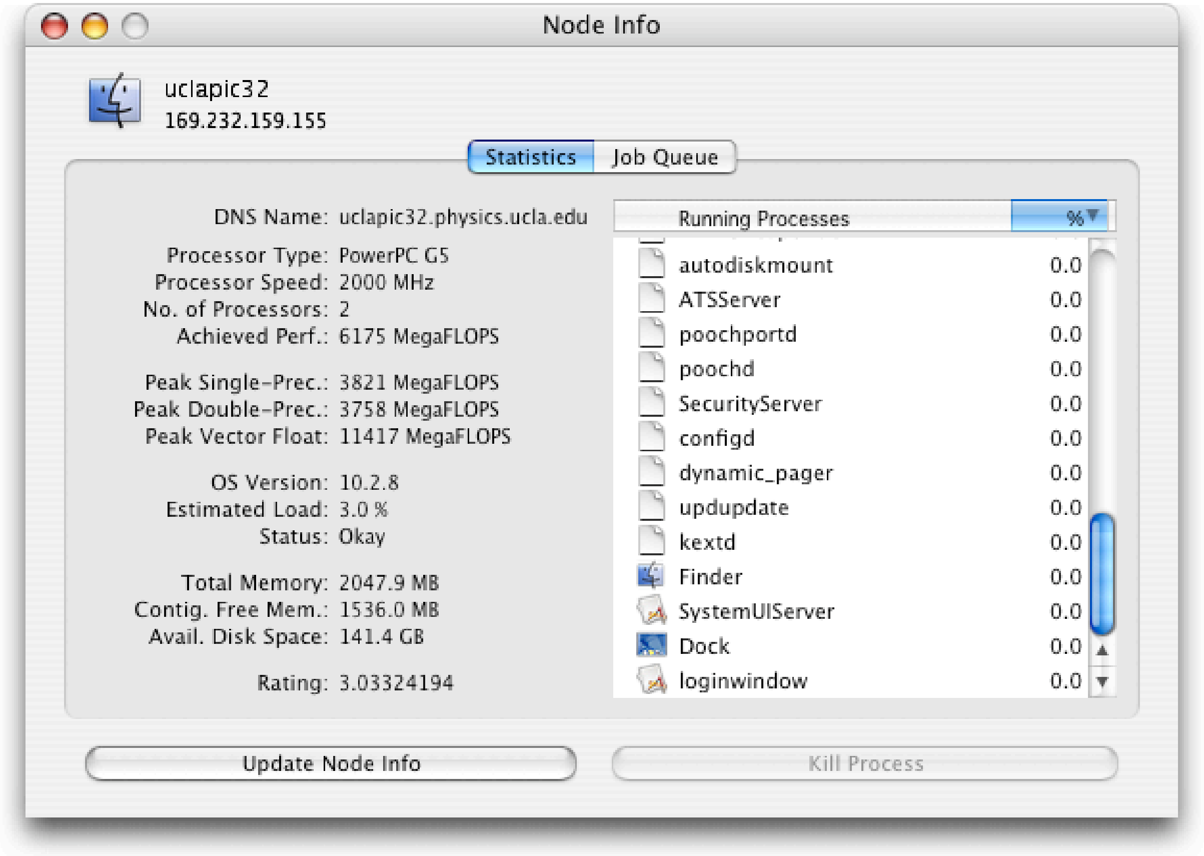
Launching... this node is presently attempting to launch this job

Running this job was successfully launched and is running

Ended this node has determined that this job terminated

Aborted thid node was used to kill this job and it is no longer running

Node Info Window



uclapic32
169.232.159.155

DNS Name: uclapic32.physics.ucla.edu

Processor Type: PowerPC G5
Processor Speed: 2000 MHz
No. of Processors: 2
Achieved Perf.: 6175 MegaFLOPS

Peak Single-Prec.: 3821 MegaFLOPS
Peak Double-Prec.: 3758 MegaFLOPS
Peak Vector Float: 11417 MegaFLOPS

OS Version: 10.2.8
Estimated Load: 3.0 %
Status: Okay

Total Memory: 2047.9 MB
Contig. Free Mem.: 1536.0 MB
Avail. Disk Space: 141.4 GB

Rating: 3.03324194

Running Processes	%
autodiskmount	0.0
ATSServer	0.0
poochportd	0.0
poochd	0.0
SecurityServer	0.0
configd	0.0
dynamic_pager	0.0
updupdate	0.0
kextd	0.0
Finder	0.0
SystemUIServer	0.0
Dock	0.0
loginwindow	0.0

Selecting one node in the Job window or the Network Scan window allows you to access that node's information. Much of the information on the left is also available in the columns of the Network Scan Window, with a few additions:

Item	Function
DNS Name	the IP name of this node according to the DNS network, if available
Total Memory	the total memory available, including virtual memory

The right side shows a list of running processes each with an approximate distribution of processor load. The determination of this information is OS specific. PCM calls the Unix "ps" command, whose results can change at any moment and can be adversely influenced by the call to "ps".

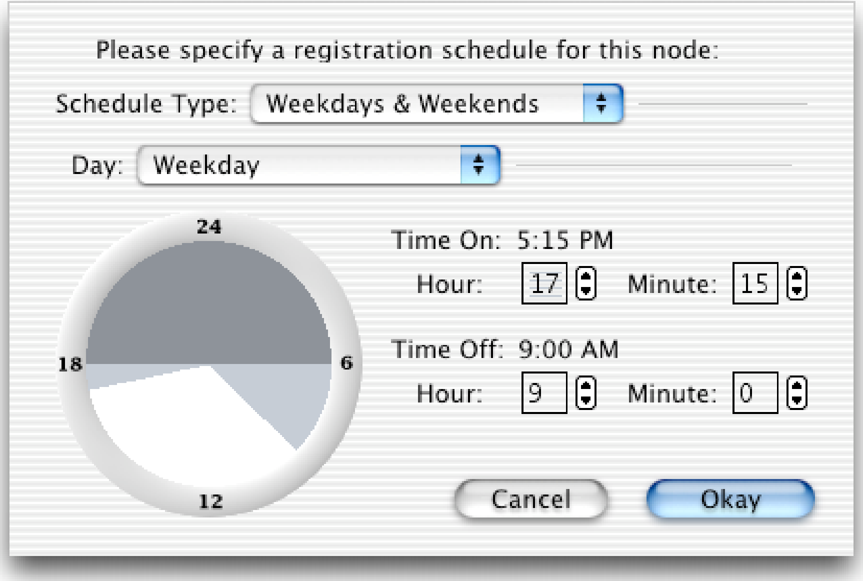
After selecting a process on this list, clicking **Kill Process** will direct the PCM on that node to use a "kill -9" command. However, in accordance with the Unix underpinnings of OS X, PCM can only kill processes that PCM has permission to kill.

Clicking on the pop-up menu will allow you to switch between **Stats & Apps** and **Job Queue**. In the **Job Queue** mode, this window will display parallel jobs in the queue on this node. The columns list the number of nodes and files selected in this job and the launch time of this job. If no launch time was selected, or if the launch time has passed, it will display Normal. Selecting one will allow you to kill that job using the **Kill Job** button.

Get Files window

Selecting a node in the Network Scan Window or the Job Window and selecting **Get Files** from the Node menu will have your PCM access the PCM on that node and access files there. It displays information about the files in a window much like a Finder window. You can drag files out of it to the Finder, you can double click on a folder to access its contents, and you can Command-click on the window title to pop up a menu to back out a directory. As a security feature, you can only access the folder in which the remote PCM resides and any of its subdirectories (folder aliases will not resolve), and you can only get files out, not drag files in. This access is completely independent of Mac OS File Sharing.

Node Registration Schedule Dialog



Please specify a registration schedule for this node:

Schedule Type: Weekdays & Weekends

Day: Weekday

Time On: 5:15 PM

Hour: 17 Minute: 15

Time Off: 9:00 AM

Hour: 9 Minute: 0

This dialog allows you to select the time PCM registers and deregisters itself on the network, which determines this node's visibility by other nodes. As seen in the above screen shot, you may set the schedule so that, on weekdays, this node is visible on the network at 5:15 pm, after you leave for home, then no longer is available for computation at 9 am when you arrive at work that morning. The Schedule Type pop-up menu can be set to Everyday, where the daily schedule will be the same every day, Weekdays & Weekends, where the schedule for weekends (Saturday and Sunday) is distinct from the schedule on Monday through Friday, and Weekly, where the schedule on each day of the week can be set separately. The Day pop-up menu selects which day you are editing, which depends on the Schedule Type setting.

For each day, the large circular dial is used to specify what portion of the day PCM will register itself. The circle represents a 24-hour clock with midnight at the top and noon at the bottom. The highlighted sector of the circle indicates the portion of the day registration will occur. Dragging the mouse inside the circle selects times rounded to the nearest 15 minutes. Dragging outside the circle, or using the text fields, allows you to select any minute of the day.

V. Administration

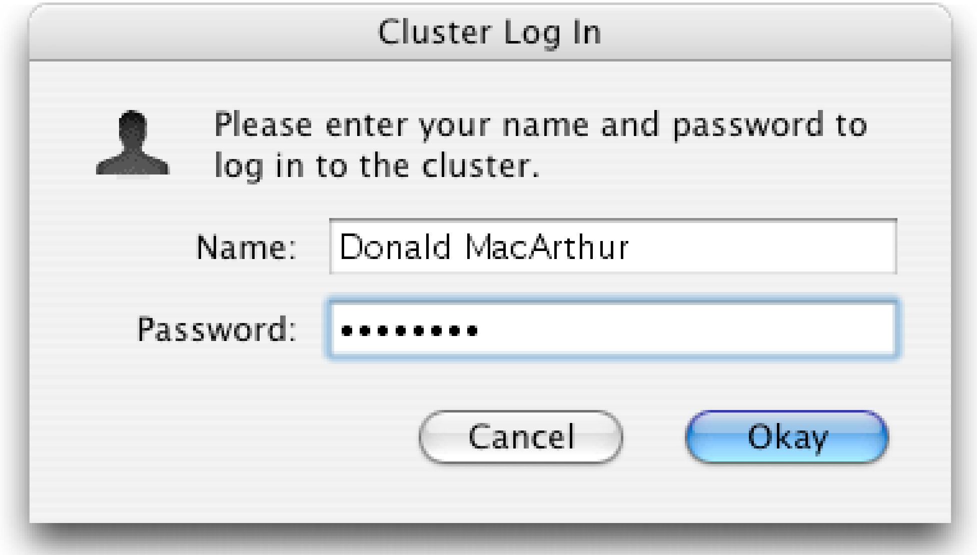
PCM enables larger clusters to be managed for use by a larger number of users. Administrators would like to accomplish as much as they can with the effort they employ to control over how the hardware is used and shared amongst users. PCM is designed to apply ease of use make it possible to apply the same amount of effort to achieve much more than could otherwise be done. PCM supplies the infrastructure of users, groups, and administrative accounts to manage the usage of cluster resources. These capabilities imply mechanisms to identify parallel computing jobs with users and track their time spent against usage policies decided by the cluster administrator as well as an account hierarchy between users and administrators. PCM allows one to administer a cluster for a large number of users, all via a user interface easy-to-use for administrators and users alike.

When PCM is first installed, it creates an initial user database file with two groups and two user accounts, user and admin. The password for the both accounts is set to a psuedorandomly generated string supplied with your PCM CD. (For the downloadable from the web site, the admin password is "pcm".) When you set up your accounts, we recommend first logging into your admin account and either set a new password or create a new administrator-level account for yourself and delete the old admin account. Then, after configuring the user accounts, use **Update User Database** to distribute your changes to all other cluster nodes. If you need to backup the encrypted user database file, it is at /Users/Shared/pcm/pcmuserdata.

Administration Menu

User menu

□ **Log In.../Log Out** - Opens a Cluster Log In window to initiate use of PCM after entering a user name and password



Cluster Log In

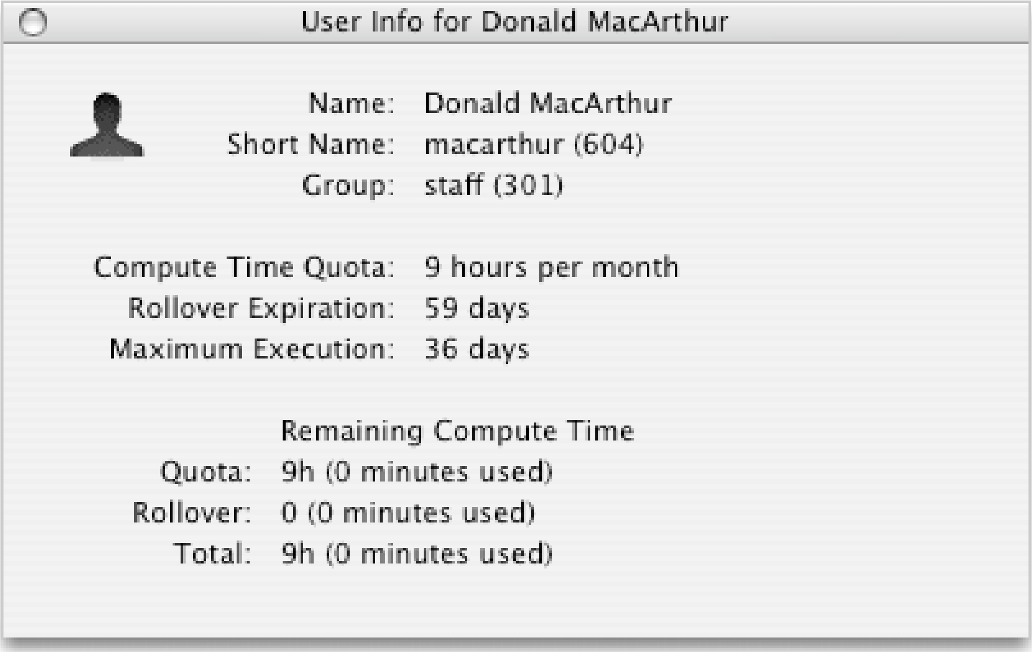
Please enter your name and password to log in to the cluster.

Name: Donald MacArthur

Password: •••••••••

□ **Change Password...** - Opens a Change Password Window where the user can enter and verify a new password for account log in.

□ **Show/Hide User Info** - Toggles a User Info Window for the currently logged in user, showing statistics on the amount of compute time the user has consumed out of their assigned quota. See the User Administration Columns section for more detail on these fields.



User Info for Donald MacArthur

Name: Donald MacArthur
Short Name: macarthur (604)
Group: staff (301)

Compute Time Quota: 9 hours per month
Rollover Expiration: 59 days
Maximum Execution: 36 days

Remaining Compute Time
Quota: 9h (0 minutes used)
Rollover: 0 (0 minutes used)
Total: 9h (0 minutes used)

□ **Migrate User Access…** - Adds a reference describing this user's account information, including password data, to the Job Window. The user may then choose nodes, just as with parallel computing jobs, where a new or updated copy of their user data will reside.

□ **Administrate Users…** - Opens the User Administration Window (see the User Administration Window section)

□ **Update User Database** - choosing from this submenu allows the distribution of this node's user database to other nodes on the cluster. Assuming all nodes are available and running on the local subnet, the **Automatically** choice will have PCM automatically acquire those nodes for updating. The **Manually…** item adds the user database to the Job Window, making it possible to choose nodes that may not have been previously available or are outside the local subnet. Ideally, this is an operation that should be performed on all nodes of the cluster.

- □ **Import Users…** - Opens a file dialog to choose a text file from which PCM will interpret data to create new users and modify existing ones. This feature allows an administrator to generate a user database based on an external source or text file, then import this user data into PCM with one action. The data format must be Unix linefeed text (such as using "Make Plain Text" in TextEdit), and the first line contains tab-delimited headers followed by corresponding tab-delimited data. The import ends with the first blank line or at the end of the file. We recommend examining the output of **Export Users** for an example, although the column order does not matter to the import, nor do all columns demonstrated in export need to exist for import. "Password" is also a recognized column.

- □ **Export Users…** - Opens a file dialog to choose the name and location of a new text file containing tab-delimited data about user settings, including CPU quota and rollover minutes, as well as calculated CPU time used and remaining for each user, parceled between quota versus rollover minutes. This text format should be straightforward to read by standard spreadsheet programs. These header fields are the header keywords used in **Import Users**.

Command-line Access

When using the macmpi.tar.gz package with PCM, we recommend that you create users on the system whose user names match the user names registered in PCM's user database and use the -m flag when launching jobs. This makes it possible correspond command-line based logins (e.g., ssh) with jobs submitted by users known to PCM. Without those consistent user names, PCM will not recognize a job submitted by users it does not know, and therefore kill those jobs because they appear to be foreign.

Administration Windows

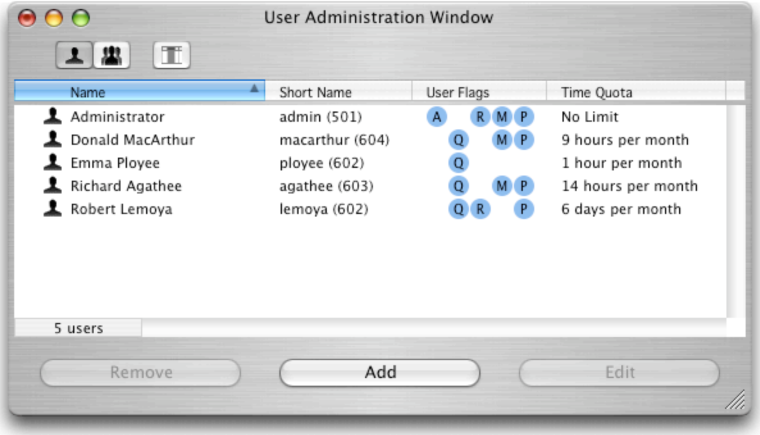
User Administration Window

PCM's User Administration Window provides access to the user and group account database. Usually only administrators would have access to this window. This window features two different views, toggled by the view buttons in its upper left corner. A dot in the close box indicates that user or group data has recently been changed.

The User View displays all user accounts in the cluster, including statistical data about these users. This view is used to administer, create, edit, and delete user accounts.

The Group View lists all groups of user accounts on the cluster and the users they contain in a hierarchal format. This view is used to administer groups and organize which users they include.

User View

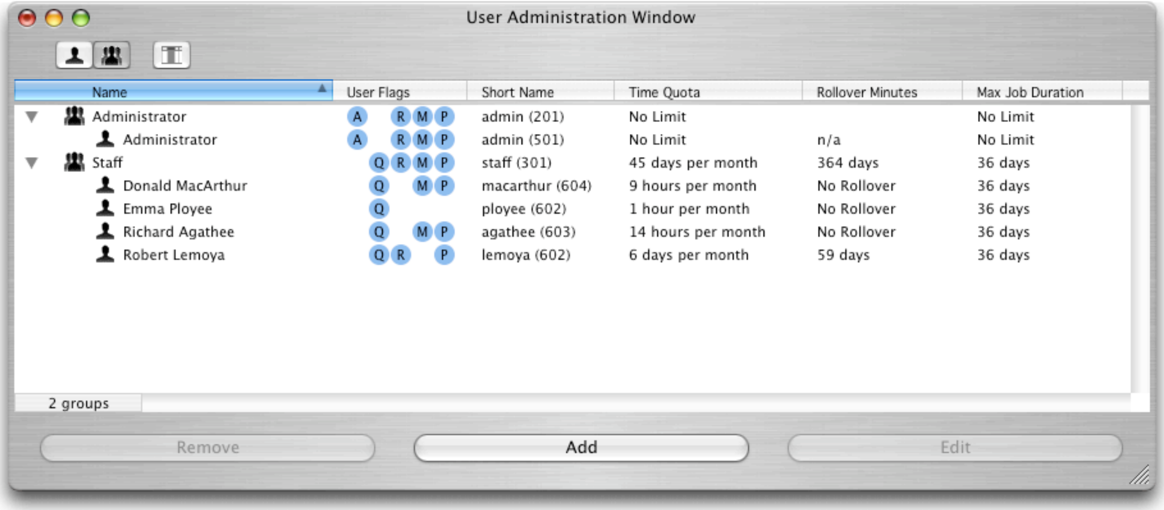


Name	Short Name	User Flags	Time Quota
Administrator	admin (501)	A R M P	No Limit
Donald MacArthur	macarthur (604)	Q M P	9 hours per month
Emma Ployee	ployee (602)	Q	1 hour per month
Richard Agathee	agathee (603)	Q M P	14 hours per month
Robert Lemoya	lemoya (602)	Q R P	6 days per month

5 users

The User View of the User Administration Window is where user accounts can be created, edited, and deleted by selecting the account and clicking on the buttons at the bottom of the window.

Group View



Name	User Flags	Short Name	Time Quota	Rollover Minutes	Max Job Duration
▼ Administrator	A R M P	admin (201)	No Limit		No Limit
Administrator	A R M P	admin (501)	No Limit	n/a	No Limit
▼ Staff	Q R M P	staff (301)	45 days per month	364 days	36 days
Donald MacArthur	Q M P	macarthur (604)	9 hours per month	No Rollover	36 days
Emma Ployee	Q	ployee (602)	1 hour per month	No Rollover	36 days
Richard Agathee	Q M P	agathee (603)	14 hours per month	No Rollover	36 days
Robert Lemoya	Q R P	lemoya (602)	6 days per month	59 days	36 days

The Group View of the User Administration Window is where groups can be created, edited, and deleted.   Users can be dragged between groups here.

User Administration Columns

The User Administration Window displays information users and groups in a spreadsheet-like format. This window is invoked using the columns button at the top of the User Administration Window. You can drag a column at its header and resize it by clicking on its right edge. Clicking on a column header re-sorts the list according to that column.

<u>Column</u>	<u>Function</u>
Name	the full name of a user or group
Short Name	the short name of a user (a.k.a. user name) or group. The ID number of that group or user is shown in parentheses.
Group	the short name of the user’s group, with the group ID in parentheses
Compute Time Quota	

the compute time allocated to a user. The quota is specified as a particular amount of time, summed over all nodes used by that user, allowed within a certain wall-clock time. For example, if a user is allocated five hours per day, then PCM will allow that user's jobs, in aggregate, to use up to five nodes one hour each, or one node for five hours, every day

Rollover Minutes Expiration

the expiration limit of this user's rollover minutes. If a user does not use up their quota on a previous quota interval, the user may carry those minutes over to the next interval. However, those minutes will expire after the time specified here. Using the above example of a user quota set to five hours per day, if this rollover expiration is set to two days and the user did not use any time yesterday, then that user could use up to ten hours on the cluster today.

Maximum Job Duration

the wall-clock time limit of this user's jobs. Jobs submitted by the user will be limited to running for the amount of time specified here.

Total Time Remaining

the sum of the compute time available to the user if that user were to start a job now. PCM keeps track of jobs submitted by users compares that time against the policies specified by the quota and rollover minutes policy settings. The time left is displayed in this column.

Total Time Used the sum of the compute time used by the user within the quota and rollover minute time intervals

Quota Time Remaining

the sum of the compute time available to the user according to the compute time quota rule only

Quota Time Used the sum of the compute time used by the user within the quota time intervals

Rollover Time Remaining

the sum of the compute time available to the user due to rollover minutes only

Rollover Time Used the sum of the compute time used by the user outside the quota time interval but within the rollover minute expiration

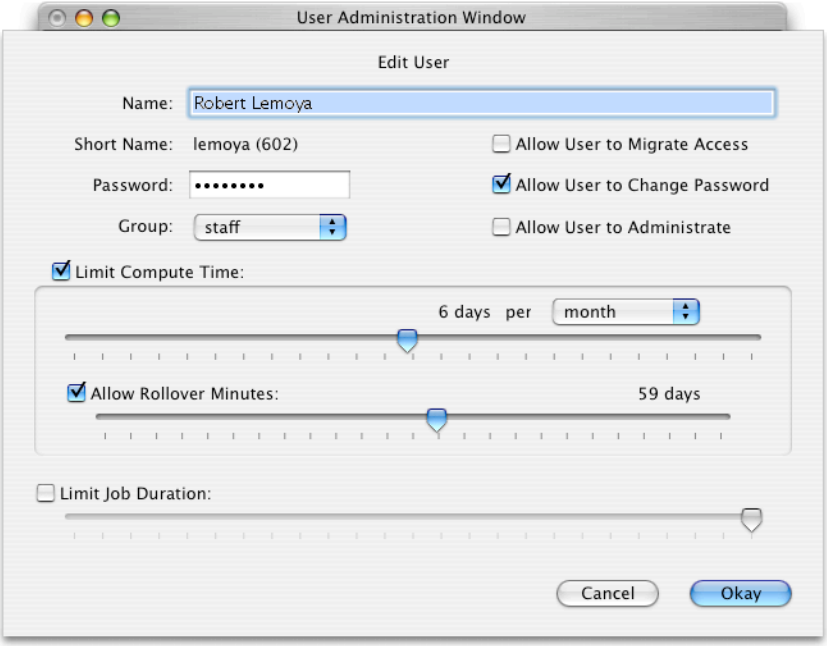
User Flags the flags of a user or group, specifying user policy features, shown graphically. Their meanings are shown in the chart below.

User Flag Icon	Meaning
A	this user or users of this group group are allowed to administrate on the cluster
Q	this user or users of this group have a compute time quota limiting their allocation on the cluster
R	this user or users of this group have been allowed rollover minutes
M	this user or users of this group have been allowed to migrate their access to the cluster from node to node

P this user or users of this group have been allowed to change their own password

Edit User Sheet

Editing a user via double-clicking or the Edit button invokes the Edit User Sheet.



User Administration Window

Edit User

Name: Robert Lemoya

Short Name: lemoya (602)

Password: ••••••••

Group: staff

□ Allow User to Migrate Access

☑ Allow User to Change Password

□ Allow User to Administrate

☑ Limit Compute Time:

6 days per month

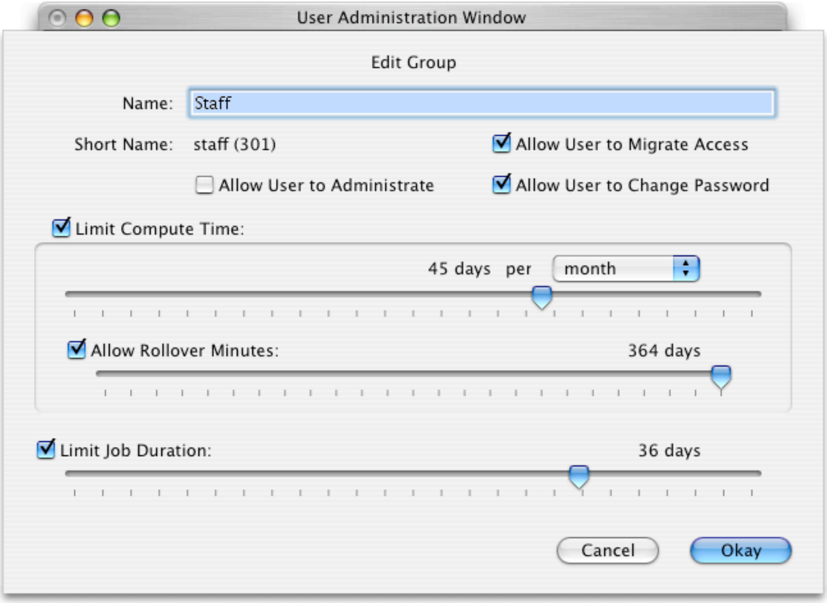
☑ Allow Rollover Minutes: 59 days

□ Limit Job Duration:

The sliders set the compute time policy for this particular user. A user's total compute quota can be limited to a certain amount of wall-clock node time per time interval, which can be a day, a week, a month, or a year. This time is the sum of all the time this user keeps any node or nodes busy, as tracked by PCM. This calculation is compared against the quota specified in this dialog and the quota of this user's group. In addition, the user may be allowed minutes to be "rolled over" from previous quota intervals. This feature allows users to combine previously unused minutes with their current quota for their compute time. Users' individual jobs can also be limited to running for a specified amount of time using the Limit Job Duration slider. When creating a new user, a New User Sheet appears that is very similar to the Edit User Sheet. Unless one is entered, the default password given to new users is “pcm”, all lowercase. Once the short name and user ID is set, they cannot be changed without deleting the user.

Edit Group Sheet

Editing a group via double-clicking or the Edit button while in the Group View invokes the Edit User Sheet.



Name: Staff

Short Name: staff (301)

☑ Allow User to Migrate Access

□ Allow User to Administrate

☑ Allow User to Change Password

☑ Limit Compute Time: 45 days per month

☑ Allow Rollover Minutes: 364 days

☑ Limit Job Duration: 36 days

Like that for an individual users, the sliders set compute time policy, but instead of just a particular user, a policy can be set for all the users in this group at once. A user’s compute time is limited to either that user’s particular quota or the group’s, whichever is more limiting. This hierarchy applies individually to the rollover minutes and job duration limits as well. When creating a new group, a New Group Sheet appears that is very similar to the Edit Group Sheet. Once the short name and group ID are set, they cannot be changed without deleting the group.

AppleScript

PCM supports three AppleScript types regarding user functions.

```
      user   noun  -- a user of the cluster
      group   noun  -- a user group of the cluster
```

The user record is data about the user, including settings specific to the user as well as calculated quota and rollover CPU time used. Similarly, a group record is data about a group. The units of the time-valued properties, such as quota or rollover time, are in minutes. Via an administrator account, these records can be accessed using the usual Standard Suite in AppleScript vocabulary. For example, to get a complete list of users, use the following command:

```
      get every user
```

To create a new user, use make:

```
      make new user with properties {{name:”Test User”, user name:“testuser”,
has quota:yes, quota:10000, group ID:301}}
```

These capabilities are useful to set up and create a new user programmatically via AppleScript or a Unix script.   If not all properties are provided, as in the make example, PCM will fill in the remaining fields with default values. Properties that return calculated time usage and remaining , of course, are ignored if fed into PCM.

```
      login   noun  -- a user’s login session to the cluster
```

The login record is about the login state of PCM. Data about the currently running user can be returned via the get command, while delete can log out the user. To log in from AppleScript, use make:

```
      make new login with properties {{user:{user name:“testuser”},
password:”pcm”}}
```

VI. AppleScript

PCM has the ability to respond to commands from a script written in AppleScript. This feature makes possible customized launch configurations and queuing systems. For example, simply by dragging an application to it, a script could launch that application automatically onto the four “best” nodes found on the network. Also, it is possible to write and launch scripts from OS X’s Unix command line; thus, launching jobs through PCM from OS X’s Unix shell is possible. In addition, because the underlying mechanism of AppleScript are AppleEvents, the inter-application messaging system on the Mac OS, it is possible for Mac applications, including mainstream ones, to ask PCM to query the network and launch parallel computing jobs.

You may look at the PCM dictionary by selecting Open Dictionary… from the File menu of Script Editor (an application commonly supplied with the Mac OS for editing AppleScript scripts) and selecting PCM in that dialog. We also present that vocabulary here. PCM fits its AppleScript vocabulary within the official AppleScript object-oriented approach, accompanied by only a few PCM-specific commands and data structures.

Standard Suite

The AppleScript Standard Suite was defined by Apple Computer, Inc., to create a standard vocabulary that all AppleScript-aware applications would understand and respond to in a consistent manner. This vocabulary is meant to encompass as many operations as possible that a user could do with an application, while allowing only a minimum of application-specific operations to fall outside this suite. The goal is, once an AppleScript user learns this suite once for one application, that user may apply that knowledge to all other applications with the greatest possible efficiency.

```
        make
               new   type class  -- the class of the new element. Keyword 'new' is optional
in AppleScript
               [at  reference]  -- the location at which to insert the element
               [with data  anything]  -- the initial data for the element, such as a
list of alias records for the file list of a job
               [with properties  record]  -- the initial values with properties of
these items, needed for nodes in a node list of a job
               [with last  type class]  -- use the data from the last submitted job for
the new job
        Result:  reference  -- to the new object(s)
```

`make` is probably the second most common command you will use with PCM. You may use it to create a new job.

```
        set myjob to make new job
```

This command directs PCM to open a new Job Window with only the node this PCM is running on as node zero. It then sets the variable `myjob` to be a reference to the new job just created.

By default, `make new job` creates a new job with an empty file list and a node list containing the node this PCM is running on. "`make new job with last job`" causes PCM to derive the new job from the last job this PCM launched according to the settings in the New Job Automatically Reloads submenu.

To add an application or files to the job, use the `make new file` form:

```
        make new file at myjob with data alias "Power Fractal" of the startup
disk
```

This command adds the Power Fractal application, which resides on the root directory of the startup disk, to the file list of `myjob`. The word `alias` typecasts the reference into an alias. Script Editor often checks to make sure the file exists before allowing you to compile the script. Adding files is similar, and this mechanism also accepts lists of files.

To add a node explicitly to the job, use the make new node form:

```
      make new node at myjob with properties node {name:"Mac number 2",
address:"192.168.1.2"}
```

The information between the brackets is interpreted as a record containing the name and address of the node. This record is typecast as type `node`, then added to the node list of `myjob`. You may also construct a list of nodes,

```
      set mynodelist to {{name:"computer01", address:"192.168.0.1"},
{name:"computer02", address:"192.168.0.2"}}
```

and add that list to the job.

```
      make new node at myjob with properties mynodelist
```

In addition, you may add the result of a node scan directly to the node list.

```
      make new node at myjob with properties (node scan)
```

The `node scan` command is described in the PCM Suite section below.

The remaining AppleScript commands of the Standard Suite that PCM recognizes are:

```
      close  reference  -- the object to close

      count  reference  -- the object whose elements are to be counted

      delete  reference  -- the element to delete

      exists  reference  -- the object in question

      get  reference  -- the object whose data is to be returned

      open  reference  -- list of objects to open

      set  reference  -- the object to change
            to  anything  -- the new value
```

These commands are general purpose because they can refer to almost any object, within reason, with PCM. Some of them are used simply, e.g., `close job window` closes the job window, or `close every window` closes all windows. `count the node list of myjob` returns an integer how many nodes are in `myjob`. Using the `open` command is the equivalent of dragging the item onto the PCM icon, so its function overlaps with `make new file`. Commands that do not make logical sense (such as `close BUSY check of myjob`) either return an error or do nothing.

The objects that these commands act upon can vary widely. Individual nodes of a `node list` or files of a `file list` can be referred to by number, such as `node 2` or `file 4` or `item 5`. Note that, in AppleScript, all counting is 1-based, so the nodes of an `n`-node `node list` are `node 1` through `node n`. The `node list` or `file list` of a job can also be specified. For example, `exists file 4 of the file list of myjob` tests if that file of the job exists, while `delete the node list of myjob` clears the node list. In the former example, `file 4 of the file list of myjob` may also be simplified to `file 4 of myjob` because the reference is clear; however `item 4 of myjob` is not specific enough.

The **`set`** command is one of the most versatile members in this suite and will probably be the most common command you use. Once you create a job and set the variable j to be a reference to that job:

```
      set j to make new job
```

you can set entire file or node lists:

```
      set the file list of j to {alias "Enterprise:PCM folder:pinput2d"}
      set the node list of j to {{name:"computer01", address:"192.168.0.1"},
{name:"computer02", address:"192.168.0.2"}}
```

set a particular file of a file list to another member of the same list:

```
      set file 2 of j to file 1 of j
```

or set particular options of the current job:

```
      set BUSY check of j to no
      set launch failure queues job of j to yes
      set target subfolder of j to "run001 folder"
      set start time of j to date "Wednesday, October 31, 2001 12:00:00 AM"
      set tasks per computer of j to use processor count
```

Setting the `target subfolder` option to `""` and `start time` to `current date` or an earlier date turns off those job options. All the substructures that make up a job are listed in the `job` class listing in the PCM Suite section of PCM's dictionary.

In addition PCM's Application Class has a two additional properties from the normal suite. You can query the **`launching a parallel job`** property to find out whether or not PCM is currently launching a parallel job onto nodes. This information is useful for scripts or parallel applications to coordinate its activities with PCM's launch process. Also, you may find out whether or not this PCM is registered by accessing the **`node registration`** property.

PCM Suite

The commands particular to PCM are in the PCM Suite.

```
      node scan  reference
            [for best  integer]  -- Search for a given number of 'best' nodes
            [for [best] processors  integer]  -- Search for a given number of
processors using the 'best' nodes
            [starting at  string]  -- scan starting at a remote node specified by an
IP address in a character string
            [BUSY  boolean]  -- Include the nodes labeled BUSY
            [all  boolean]  -- Include all nodes found
            [new scan  boolean]  -- Explicitly perform a new node scan
      Result:   node  -- list of nodes
```

```
      job scan  reference
            [starting at  string]  -- scan starting at a remote node specified by an
IP address in a character string
            [all  boolean]  -- Include all jobs found
            [new scan  boolean]  -- Explicitly perform a new node scan
            [lists  boolean]  -- Explicitly include node and file lists in job records
            [recalling  integer]  -- Recalls job records back this far in time
(e.g., days, weeks, or months)
      Result:   job  -- list of jobs
```

```
      network scan  reference
            [for best  integer]  -- Search for a given number of 'best' nodes
            [for [best] processors  integer]  -- Search for a given number of
processors using the 'best' nodes
            [starting at  string]  -- scan starting at a remote node specified by an
IP address in a character string
            [BUSY  boolean]  -- Include the nodes labeled BUSY
            [all  boolean]  -- Include all nodes found
```

```
          [new scan  boolean]  -- Explicitly perform a new node scan
          [nodes  boolean]  -- return found nodes
          [jobs  boolean]  -- return found jobs
          [lists  boolean]  -- Explicitly include node and file lists in job records
          [hidden  boolean]  -- hide the network scan to reduce its visual impact
     Result:   node or job  -- list of nodes or jobs
```

`network scan` directs PCM to open the Network Scan Window and begin scanning the network. By default it returns node records, but the `with jobs` switch with have it return job records instead. `job scan` is equivalent to `network scan with jobs`, and `node scan` is equivalent to `network scan with nodes`. PCM will wait about eight seconds before returning the results of its search. The command will only return nodes which, in that time, successfully reported their status as Okay after being discovered. Note that the results of this search may or may not overlap with the nodes already present in a job you are constructing.

When the returned object is a list of nodes, each item of this list is a record containing information about the node's name, IP address, load, free memory, OS version, achieved performance, rating, and so on. These items generally correspond to the node scan columns described in detail in Chapter IV. A complete list of the members of these records are listed and defined in the `node` class section of the PCM Suite in the PCM dictionary. Much of the information about a node can change at any time, so it is not recommended to cache data about other nodes. Likewise, job data can change and evolve over time, as queued jobs are launched and running jobs end.

Passing an IP address (in a string format, such as `"192.168.1.7"` or `"farawaymac.otherdomain.edu"`) in the `starting at` option directs PCM to perform a Remote Node Scan using a PCM at that address. The success of this operation depends on whether or not a PCM is actually running on a Mac at that address. Assuming that PCM successfully relays information about other nodes, your PCM will request the status of those nodes and return the results to your script as before.

Passing an integer n in the `for best` option of this command directs PCM to return only the best n nodes of the search for nodes. Nodes that are busy or cannot be contacted are never included in this list. It is possible for this command to return less than n nodes. Which nodes are the "best" nodes is determined by PCM's internal rating calculation, which depends upon data about that node which change from moment to moment. To determine the "best" nodes according to your own criteria, you can construct a script that sorts the results of `node scan` using a rating function of your own design.

To specify a number of processors to use, use the `for processors` option of this command. PCM uses the same criteria as the `for best` option, except it limits the nodes it returns so that the sum of the number of processors on the return list most closely match the number of processors specified here. This is useful if you are using the Use Processor Count setting of the Tasks per Computer job option and want to launch only a limited number of tasks.

The output of this routine can be placed directly into a job's node list:

```
      make new node at j with properties node scan for best 4
```

but you may instead wish to edit the search results yourself before passing it to your job:

```
      set nodesearch to node scan for best 4 starting at
"farawaymac.otherdomain.com"
-- edit the list in nodesearch here
                                        …
--
      make new node at j with properties nodesearch
```

Be creative!

```
      launch  job  -- job to launch: required
```

The `launch` command directs PCM to submit a job that you pass by reference. For example,

```
      launch myjob
```

would launch a job that you originally made using **set** `myjob` **to** `make new job`.

Additional Reading

Documentation about AppleScript in general is available in print and on the web.

Derrick Schneider (with Hans Handsen & Tim Holmes), *The Tao of AppleScript*, 2nd edition, (BMUG and Hayden Books, Indianapolis, Indiana, USA, 1994).

Apple's official AppleScript web site http://applescript.apple.com/

MacScripter - an independent web site with links, information, and resources, including numerous examples, related to scripting on the Mac http://www.macscripter.net/

Apple's AppleScript mailing list http://www.lists.apple.com/applescript-users

VII. Security

PCM is its own lock and key. You should keep track of your PCM like you keep track of your keys.

Before PCM will accept commands from another PCM, it must receive a passcode that matches its own. Then, all subsequent commands use a 512-bit encryption key that rotates for each message in a psuedo-random manner. Only those two PCMs can predict the next encryption and decryption keys. If a mistake in the passcode or commands is made at any time, PCM will reject the connection. Since PCM waits a second or two before it accepts another connection, an exhaustive search for the correct encryption keys (2^{512} possibilities once per second would take over 10^{145} years) will be extraordinarily unlikely to succeed.

The first passcode and the start of the rotating key are 512-bit psuedo-random numbers derived from the registration name of that PCM (which is set at compile time). Therefore, only PCMs of the same registration will be able to communicate with one another. Because the registration name is unique for each PCM customer, a copy of PCM registered to, say, MIT, will not be able to communicate with a PCM registered to UC Berkeley. (For cross-registered PCMs or other custom configurations or encryption methods, please e-mail Dauger Research.)

Security for your cluster then becomes dependent on the security of your PCM registered with your registration name. Your PCM can be installed on the Macs of your cluster, and, if no additional copies of PCM exist, no one can get in. But if you make a copy of that PCM and bring it home to access the cluster, the security of that cluster depends on how securely you keep that extra copy of PCM.

The nature of this security is analogous to having the ability to copy a key to a locked office. It is not uncommon to entrust a group of people with the keys to a shared resource, such as office equipment. The security of the equipment is shared by those who have copies of the key. These people understand the responsibility that comes with the privilege for that access. Access to PCM can be shared in a similar way.

In testing by Dauger Research, PCM's ability to break through firewalls is no greater than that of a typical web browser.

If someone loses a key to an office, it is not uncommon for the office locks to be rekeyed and a new set of keys distributed to the users of that office. Likewise, if you feel the security of your PCM has been compromised, it is possible to obtain a "rekeyed" PCM. The PCM code allows psuedo-random variants of the 512-bit passcode based on the same registration name. The rekeyed PCM will not communicate with the previous PCM of the same registration. An active PCM subscription with Dauger Research will enable you to obtain a limited number of rekeyed PCMs.

If you are using the downloadable demonstration version of PCM, you should be aware that the same version can be downloaded by anyone else on the web. So, if they have the IP address of your Mac, they could access your Mac, to the extent that PCM allows, over the Internet. Although guessing your Mac's IP address is unlikely, a uniquely registered PCM makes for much better security.

VIII. Frequently Asked Questions

Does PCM work on Intel-powered Macs?

Yes. PCM is a Universal app, so it runs on both PowerPC and Intel Macs.

How do I uninstall PCM completely?

You may remove PCM and all its components by allowing PCM to run normally, then holding down the Option key after launching the PCM Installer. In the PCM Installer, a dialog should appear that allows you to select either to upgrade or uninstall PCM. Clicking on uninstall will delete the running PCM and the components that allow it to run at logout. The latter process may require administrative authorization.

How do I obtain a PCM subscription, and what do I get?

With your ordered copy of PCM, you get a one-year subscription, which can be renewed annually at 25% of the purchase price (at the time of renewal) of a new copy. As a subscriber, you will receive free updates to the software (approximately twice per year) and technical support via e-mail. The subscription also entitles you to up to six free rekeyed PCMs, if you feel your security has been compromised, per year. If you would like to upgrade your PCM, please contact Dauger Research.

When I get an update, do I have to reinstall it on every Mac myself?

On the CD you receive, Dauger Research, Inc., will supply an updater program we call a PCM Package. You can use the usual PCM launching mechanism to deliver this Package to all your nodes, which will then unpack and update PCM to the new one on all your nodes automatically and simultaneously. No reboot necessary.

IX. Contact Information

PCM is released by Dauger Research, Inc. The latest version of this document is available from the web site. We are open to suggestions and wish lists regarding parallel computing issues. Depending on your needs and the nature of the request, custom configurations of PCM can be arranged.

Dauger Research, Inc.	http://daugerresearch.com/
PCM web site	http://daugerresearch.com/pcm/
The Dauger Research Vault, providing tutorials, sample code, and other documentation	http://daugerresearch.com/vault/
PCM technical support, questions, and feedback	pcm@daugerresearch.com
Dauger Research mailing list info	http://daugerresearch.com/mailing-list/

The latest MacMPI libraries are available from the AppleSeed web site.

AppleSeed	http://exodus.physics.ucla.edu/appleseed/
UCLA Plasma Physics Group	http://exodus.physics.ucla.edu/

Last Update: May 17, 2006