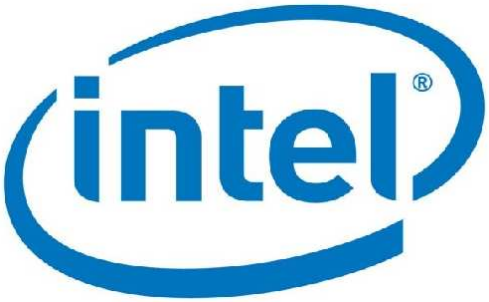


Build Your Own Cluster

With open source software and Intel hardware

Paul Guermonprez

Intel Software and Solutions Group

April 18, 2008

Goals

- Goal : Install a cluster yourself from scratch, after this training you'll be able to do it alone and understand the internal basics.
- Real life : At the end we'll install real life bioinformatics HPC applications to learn how to use your cluster.
  Just add your data and serve.
- Required : Off the shelf hardware and open source tools (we'll use OSCAR Toolkit 5.1b1).
- Operating system : We'll use RHEL Server 5.1, but you can use any OSCAR compatible distribution, including non commercial ones.
- Optional : Infiniband connection between compute nodes.
- Non-exhaustive : This document is just a use case, you still need to read the official OSCAR documentations, forums, lists ...
- Please send your feedback and bugs : paul.guermonprez@intel.com

Audience

For beginners

This is a simple tutorial for beginners, with step by step instructions. Knowledge of clusters, MPI, parallel programming are not required.
If you plan to do it alone, basic linux system administration knowledge is needed, you should be familiar with the terms : RPM, DHCP, gcc, shell export. With an instructor, only the shell basics are required.

Not only for sysadmins

Hopefully beginners in cluster sysadmin will find this tutorial useful, and may find a rack of servers to build one. Most bioinformaticians have to use a cluster from time to time, and the best way to use it efficiently is to build a simple one, play with it and see how it works.

Procedure

We will follow this procedure, a typical lab with 12 students constructing 6 minimal clusters and technical explanations will take a full day :

- Install the master node Operating System
- Master node post installation setup
- OSCAR installation on the head node
- OSCAR deployment on the compute nodes
- Compute nodes infiniband setup (optional)
- Intel Cluster Toolkit installation
- Hello World HPC software installation
- BioInformatics HPC software installation
- Monitoring, MPI trace, queuing system

Hardware

Hardware

We'll use Intel64 hardware (sometimes referred to as "x86_64", "EM64T", ...) the kind of server you can buy from many manufacturers.

Lab setup

For a learning lab it would also work on regular desktops or laptops with simple fast ethernet switches. The linux operating system itself and the OSCAR toolkit will work fine on a typical Dual-Core laptop.

Production

Of course if you plan to build a production cluster choosing optimal Intel64 HPC hardware and network equipment is important.

Network Hardware

Ethernet

By default we'll configure the cluster with integrated ethernet, because it's easy and integrated in most computers. It's important to know what kind of network hardware you have before you start installing, because the OS must be able to work with it. In our case we'll cover a more complicated case : a recent network card, so recent that the last version of the OS does not have a RPM driver included. We'll see how to include a driver in the installation process.

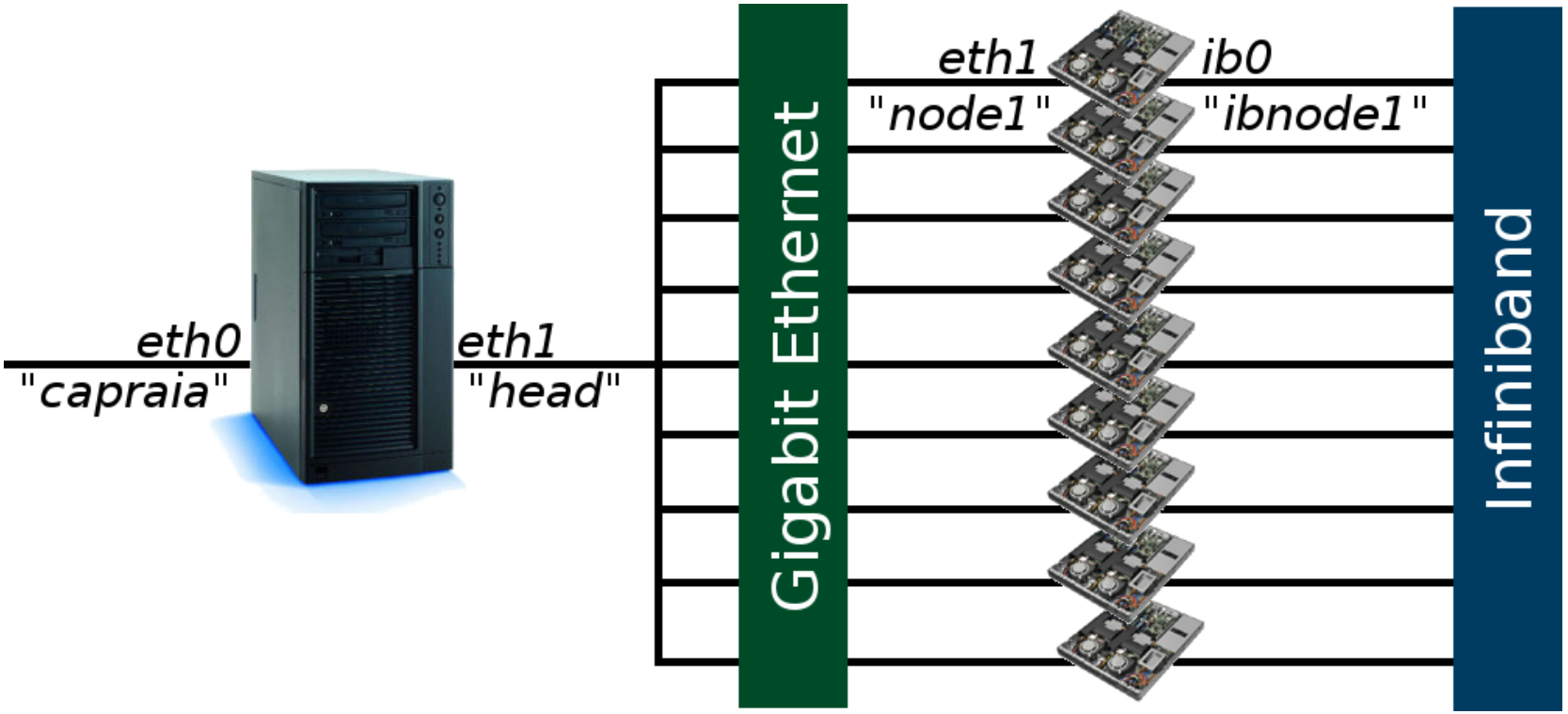
Infiniband

Gigabit ethernet works fine out of the box but some HPC software require a low latency network. Infiniband is one of these low latency technologies, we'll see how we can integrate and configure infiniband in our cluster. If you don't infiniband hardware don't worry this is optional.

Network planning

- An OSCAR server is a DHCP and PXE server, so to keep it simple you don't want such a server on your regular network, you want a separate network.
- 2 Steps : You can start the installation of the OSCAR main server on your network, and later put it on a separate network, when you'll want to install the OSCAR toolkit and install compute nodes. A simple VLAN separation between the networks is enough, the master node will act as a router.
- For the same reasons and to keep it simple, keep a separate network for each of your OSCAR clusters.

Network planning



Network IP and naming configuration

Master node

- eth0 ethernet network interface is "capraia" 192.168.0.118 on my lab network.
- eth1 ethernet network interface is "head" 192.168.10.1 on my cluster ethernet network.
- no infiniband on the master node.

Compute nodes (X between 1 and 9 in my case)

- eth1 ethernet network interface is "nodeX" 192.168.10.10X on my cluster ethernet network.
- ib0 infiniband network interface is "ibnodeX" 192.168.20.10X on my cluster compute network.
- eth0 ethernet interface is not used on compute nodes.

Booting

- BIOS : Make sure your server is booting from DVD.
- Boot : Insert RedHat DVD in the tray, during the boot from DVD prompt, you can ask to boot with VNC enabled passing kernel arguments "linux vnc". (You may not want to spend 2 hours in a cold and noisy server room)
- Basics : The installation procedure may ask you to choose the network card, choose eth0, the keyboard layout and language.
- IP : After these steps, a message appears giving you the IP address of the server, go back to your desk, and launch a VNC viewer on display :1 with the the IP (for example : "192.168.0.118:1"). ScreenShot i1 for a typical linux desktop and a windows desktop with TightVNC viewer.

Installation first steps

- Welcome : If the VNC setup works fine, you should see the welcome screen. ScreenShot i2.
- Serial number : You can enter your serial number or not. ScreenShot i3.
- Install or upgrade : Choose install. ScreenShot i4.
- Disk setup : In my case I remove everything on the disk sda and create a default layout. ScreenShot i5.

Network, I18N and password

- Nextwork eth1 : In my case eth0 is connected to my lab network and is able to receive settings from DHCP. I only need to configure eth1 as shown on ScreenShot i6 : "192.168.10.1"/"255.255.255.0", OK.
- Hostname : I choose to set the hostname "head" manually. For OSCAR it's important to have a defined hostname corresponding to the network interface managing the cluster. ScreenShot i7.
- Internationalization : Select your location. ScreenShot i8.
- Root : ScreenShot i9, Choose your root password.

Software packages

- Tasks : You can now select what software tasks you want to perform. For OSCAR you don't need much because a lot of packages come from OSCAR directly, like the DHCP server. You can select "Software Development" and "Customize now". ScreenShot i10.
- Customize : You need to check "Legacy Software Development" in the "Development" folder. ScreenShot i11. (in fact we only need "compat-libstdc++" to install Intel Compilers later).
- "Install everything", not a good idea : It may seem tempting to select all packages, but it would lead to strange problems, some packages installation having consequences. Keep it simple.
- Details : Be sure you have your favorite editors installed, software like "screen" are included by default.

Software packages

- Begin installation : ScreenShot i12.
- Installation : ScreenShot i13.
- Installation finished : ScreenShot i14. Remove the installation DVD and reboot.
- Done ! You should have a working OS, congratulations.

First Boot - Firewall,SELinux,SSH

You need to be in front of the server console for the first boot :

- Firewall and SELinux : Disable both. It would require a lot of configuration to make it work (or not). As a consequence this cluster won't be very secure. (You may enable firewall for the eth0 master node interface, not needed for the cluster).
- Reboot is needed.

Then, from your remote desktop, logged as root :

- SSH server : Edit /etc/ssh/sshd_config and uncomment the line (remove the "#") : "PermitRootLogin yes" and restart the ssh server : "/etc/init.d/sshd restart"
- SELinux : Even is not activated, you need to remove the test binary :

```
mv /usr/sbin/selinuxenabled /usr/sbin/selinuxenabled_echo
```

Post install configuration - VNC

After the first boot, the goal is now to allow remote connection via VNC. But if you are connected locally to the master node or from a linux machine, you won't need it.

- ssh to the master node : "ssh -X root@capraia" in my case. (Note that ssh "paul@capraia" then "su" are not equivalent)
- Vnc Password : "vncpasswd" and type a password twice.
- Vnc Server : Edit the file "/etc/sysconfig/vncservers", add :

```
VNCSERVERS="1:root"
VNCSERVERARGS[1]="-geometry 800x600"
```

- Restart VNC Server : Type "/etc/init.d/vncserver start" :

```
Shutting down VNC server: 1:root          [FAILED]
Starting VNC server: 1:root               [  OK  ]
```

- Connect : From your desktop, connect to the VNC server. ScreenShot i1.

Post install configuration - Host Name

- "hosts" file : Host names/IP associations are kept in a file called "/etc/hosts". The format is standard but different distributions configure the names differently.
- Requirements : We need to have a fixed name for the cluster side network interface (eth1), we need this name to be THE machine name and not only A machine name, and just to keep it clean, we need the reverse name to be machine name.
- Resulting "/etc/hosts" file : You don't need to edit the first and last line, but the second one is important (including order).

```
127.0.0.1       localhost.localdomain localhost
192.168.10.1    head head.cluster.lab.ipa.intel.com
::1             localhost6.localdomain6 localhost6
```

- If you type "hostname" you should see "head".

Before you begin

You should have 3 files, copy them from a USB drive or the network to a folder "/cluster/OSCAR" :

- igb-1.0.1-1.x86_64.rpm
- OSCAR-repo-common-rpms-5.1b1.tar.gz
- OSCAR-repo-rhel-5-x86_64-5.1b1.tar.gz
- rhel-5-x86_64-IB.rpmlist
- hello.c

I recommend you install "igb" even if you don't need it for your hardware, because it would require several file editing later.
I use OSCAR 5.1b1, but you may have the final version available.

Preparation - OS RPMs

- During the OSCAR installation, you'll need to provide a folder with all rpms of your RHEL OS (2798 files, more than 3 Go).
- But you can't just use "cp" to copy 2798 files (cp has its limits), you have to use find and pipe it to cp.

```
mkdir -p /tftpboot/distro/redhat-el-server-5-x86_64
cd /tftpboot/distro/redhat-el-server-5-x86_64
# with RHEL DVD inserted
find /media/redhat/Server/ -name "*.rpm" | xargs -i cp {} .
```

Preparation - Custom RPMs

- If you have a rpm you may need later, add it now in the folder, later you would need to "rebuild the repository". I need a driver called "igb" for my compute nodes recent network cards.
  cp "igb-1.0.1-1.x86_64.rpm" to
  "/tftpboot/distro/redhat-el-server-5-x86_64".
- OSCAR will deploy a configuration based on the master node configuration. But my master node does not have an "igb" network card so I am installing and loading the igb kernel module manually :

```
rpm -ivh igb-1.0.1-1.x86_64.rpm
modprobe igb
lsmod | grep igb
# i should see :
igb                    141892  0
```

- Note : My custom igb rpm only works with this kernel version ...

Preparation - OSCAR RPMs

Create folders and untar OSCAR files :

```
mkdir -p /usr/tftpboot/rpm
ln -s /usr/tftpboot /tftpboot
mkdir /tftpboot/OSCAR
cd /tftpboot/OSCAR/
tar -xvzf /cluster/OSCAR/oscar-repo-common-rpms-5.1b1.tar.gz
tar -xvzf /cluster/OSCAR/oscar-repo-rhel-5-x86_64-5.1b1.tar.gz
```

With OS and OSCAR, you now have 3 big folders in /tftpboot :

```
find  /tftpboot/ -mindepth 2 -type d
# should display :
/tftpboot/OSCAR/common-rpms
/tftpboot/OSCAR/rhel-5-x86_64
/tftpboot/distro/redhat-el-server-5-x86_64
```

Preparation - OSCAR RPMs installation

RedHat 5.1 is a yum based distribution, all packages are installed and managed with "yum".

- OSCAR needs "yume", so we need to install yume (using yum). But yum is configured by default to reject non signed packages, so we disable this feature first : edit /etc/yum.conf and set "gpgcheck=0".
- Then create a repository from OSCAR common rpms and install yume :

```
yum install createrepo /tftpboot/OSCAR/common-rpms/yume*.rpm
```

- You can now use yume to install all the OSCAR packages :

```
yume --repo /tftpboot/OSCAR/common-rpms install oscar-base
```

OSCAR Installer Launch

You should now have a folder /opt/oscar, among other things, installed.
Connection :

- If you are connected locally, login as root and graphically.
- If you are connected from a remote linux/unix machine, use "ssh -X root@capraia" (-X to allow graphical display, and direct root login to avoid shell problems, replace capraia by your machine name).
- If you are connected from a remote windows machine, use vnc to have a working display setup.

You can now launch the OSCAR main installation software, you just have to choose what network interface will be used for the cluster, in our case eth1 (verbose ouput level 3 is a good idea) :

```
cd /opt/oscar
OSCAR_VERBOSE=3 ./install_cluster eth1
```

OSCAR Packages Selection and Installation

- The first time you launch OSCAR, the rpm repository will be built, it may take a minute or two. Then you will see the main OSCAR window : ScreenShot o1.
- Choose "Step 1: Select OSCAR Packages to install", and add "jobmonarch" and exit. ScreenShot o2. In this dialog you can choose to have SGE instead of Torque and other packages, but we will only cover Torque in this tutorial.
- Choose "Step 2: Configure Selected OSCAR packages" (ScreenShot o3), then Torque (ScreenShot o4) : we want in our cluster to have the head node as free as possible, so we uncheck the box "run batch system client on head node". If you build a 2 node cluster you may want to check this box. Choose "log level" 16 and click "Save".
- Choose "Step 3: Install OSCAR Server Packages", wait a minute or two and you should see a message ScreenShot o5.

Build Client Image

- We need to include a few more things in the client image, in our case the igb driver rpm (custom) and the infiniband rpms (from RHEL rpms). Copy the file "rhel-5-x86_64-IB.rpmlist" to "/opt/oscar/oscarsamples".
- Choose "Step 4: Build OSCAR Client Image", now available. ScreenShot o6.
- You can now select the right "Base package file" as shown on the screenshot.
- Select a SCSI "Disk partition file". Modern SATA disk are seen as SCSI disks.
- "Build Image" and close.

Define OSCAR Clients

- You need to define your client nodes, just a list of hostnames and IPs. To do that choose "Step 5: Define OSCAR Clients". You should see parameters to define this list : ScreenShot o7.
- Let's say you want a node called "node1.cluster.lab.ipa.intel.com" 192.168.10.101, node2... with ...102, and so on.
- You can enter a domain name for your nodes ("cluster.lab.ipa.intel.com") and a base name ("node").
- I have 9 nodes in my cluster plus head node. The starting IP should be 192.168.10.101 because I want both names ans IPs to start at 1 and not 0.
- "Add Clients" and "Close".

Setup Nodes Networking

- We had a list of client hostnames and IPs, now we have to map this list with real hardware using network cards MAC addresses.
- Choose "Step 6: Setup Networking", ScreenShot o8.
- You typically have an order in mind for your servers (top server in the rack should be node1, bottom server node9 ?). So you want the mapping to be defined precisely. If you have a file with MACs sorted in a file, you can import it, if you don't click on "start collecting MACs", then power on your server one by one, waiting a little between each server.
  You should see the MACs appear in the left list in the right order.
- When all the MACs are collected or imported, click "Stop collecting MACs" then "Assign all MACs". Power off your nodes.
- Check "Enable UYOK", click "Enable install mode", "Configure DHCP Server" then "Setup Network Boot".

Deployment and test

- You can now choose "Monitor Cluster Deployment", ScreenShot o9.
- Power on your nodes (order is not important) and if everything works fine you should see them appear in the monitor, then monitor the progress. A lot of things can go wrong, like problems detecting storage or network.
- You can plug a monitor on one of the nodes to see what is happening locally.
- At the end all your nodes should appear rebooted, with a login screen on the node screen. Click quit.
- Choose "Complete Cluster Setup" then "Test Cluster Setup" (ScreenShot o10).
- If all the tests are "PASSED" in green you have a working cluster, congratulations !

Intel Tools

- Download the "Intel Cluster Toolkit Compiler Edition ®" for Linux from intel.com with a license (free evaluation available online).
- By default Intel tools are installed in /opt/intel, but here we'd like to have a unique installation shared via NFS, and it may not a good idea to share /opt in our case.
- So we'll install everything in /cluster/intel. Create this folder, we'll see how to share a folder with nodes.
- Untar the ICT file and run the install.sh file, the rest is a simple menu driven script.

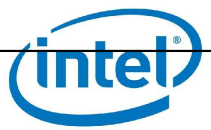
Adding Users and Groups

You may know how to add a user on a typical linux system, but here we'd like to do it for all the nodes in one operation. We'll use the "c3" tools :

```
/usr/sbin/groupadd cluster_users
/usr/sbin/useradd -u 1000 -g cluster_users paul
/usr/bin/passwd paul    # enter your password twice
/opt/sync_files/bin/sync_files
```

Let's test with cexecs, a script to launch commands locally on all nodes from the head node.
Let's try to find the word "paul" in the files "/etc/password" on the nodes :

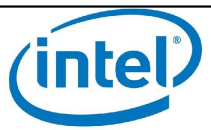
```
/opt/c3-4/cexecs grep paul /etc/passwd
************************* OSCAR_cluster  **************************
--------- node1---------
paul:x:1000:501::/home/paul:/bin/bash
--------- node2---------
paul:x:1000:501::/home/paul:/bin/bash
...
```



Create and share NFS folders

We have created a folder "/cluster" to put Intel tools but also shared data, the "/home" folder is already shared.

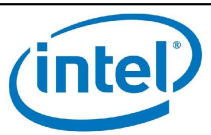
```
# First create an empty folder on remote nodes
/opt/c3-4/cexec mkdir /cluster
# add line in "/etc/fstab" (to import your shared folders)
# nfs_OSCAR:/cluster      /cluster        nfs     rw      0
# and push the /etc/fstab file to the nodes :
cd /etc ; /opt/c3-4/cpush fstab /etc/
# add line in "/etc/export" (to share the folder)
# /cluster 192.168.10.1/255.255.255.0(async,rw,no_root_squash)
/etc/init.d/nfs restart
# ask all nodes to mount the shared folder we've just defined
/opt/c3-4/cexecs mount /cluster
# testing on remote node : do you see a file in the shared folde
/opt/c3-4/cexecs ls -la /cluster/intel/ictce/3.1/ictvars.sh
```



Intel Tools Environment Variables

To use an Intel software you usually need to setup a few shell environment variables, here we would have to run "source ictvars.sh" for ICT.
It may be interesting to have this done automatically, but to let every user customize if needed. In OSCAR, there's a simple but interesting tool called switcher to allow you to switch from one compiler to another with a simple command. But here we'll see how to just use all Intel tools or not without switcher, launching the source from " /.bashrc" file.

```
# log as paul, not as root, you'll have to let users do it
echo "source /cluster/intel/ictce/3.1/ictvars.sh" >> ~/.bashrc
# logout+relog as paul, let's see which mpirun we use on nodes
cexecs which mpirun
************************* OSCAR_cluster  ************************
--------- node1---------
/cluster/intel/impi/3.1/bin64/mpirun
# ... OK
```



Setup MPI files

We now have a good shell environment, we need a configuration file to start working with MPI.

- Create a file "mpd.hosts" in your home with nodes names including the head node, one per line (head, node1, node2, ...)
- Boot the mpi daemons on nodes from head (10=9nodes+head) :

```
mpdboot -n 10 -f /home/paul/mpd.hosts -r ssh
```

- And run a simple test :

```
mpdtrace
        head
        node2
        ... OK
```

MPI HelloWorld

- Let's get the source file in a new folder, compile it and run it :

```
mkdir test ; cd test
cp /cluster/OSCAR/hello.c .
mpicc -o hello hello.c
mpiexec -np 10 /home/paul/test/hello
  Hello world from process 0 of 10
  Hello world from process 1 of 10
  # ... OK
```

- With ldd, you are sure you have an Intel MPI binary.

```
ldd hello | grep intel
  libmpi.so.3.2 => /cluster/intel/impi/3.1/lib64/libmpi.so.3.2
  libmpigf.so.3.2 => /cluster/intel/impi/3.1/lib64/libmpigf.so.3
```

mpirun vs mpiexec

- You may have seen mpirun command on other guides, mpdboot+mpiexec=mpirun, that's simple you just have to give mpirun all the parameters mpiboot and mpiexec will need.
- Let's close mpiboot with "mpdallexit".
- "mpdtrace" should not give any output.
- And try with "mpirun" :

```
mpirun -f /home/paul/mpd.hosts \
  -r ssh -np 10 /home/paul/test/hello
```

- But using mpiexec will save you the mpiboot startup time each time you run a MPI software during your session.

Different network setup

We've seen a MPI HelloWorld running, it was using sockets over the ethernet network.

- The head node and all compute nodes had a common ethernet network setup by OSCAR. If you go back to the network topology, you'll see our cluster has a dedicated infiniband network between compute nodes, not visible from the head node.
- We have to setup this second network, with both hostnames IPs and routes, then tell Intel MPI to use it.
- But there's an important technical detail : You have to configure your infiniband device as a network device, but make sure your MPI is not using regular sockets over this device, but a special rdma mode. If you use sockets over infiniband, it will work fine, faster then sockets over ethernet, but slower then rdma over infiniband. We'll see how.

Test hardware

- First, I need to see if the infiniband hardware is seen on nodes :

```
cexecs ibaddr
  --------- node1---------
  GID 0xfe8000000000000002c90200245c9d LID start 0x3 end 0x3
  # OK ...
```

- Then if the device is seen as a network device "ib0", just like "eth0" and "eth1" are the network cards :

```
cexexs /sbin/ip addr show dev ib0
  --------- node1---------
  4: ib0: <BROADCAST,MULTICAST> mtu 2044 qdisc noop qlen 128
      link/infiniband 00:00:04:04:fe:80:00:00:00:00:00:00:00:08
  # OK ...
```

A few links ...

A few links missing, you need to be root to do that :

```
cexec ln -s /etc/ofed/dat64.conf /etc/dat.conf
cexec ln -s /usr/lib64/libdat.so.1 /usr/lib64/libdat.so
cexec ln -s /usr/lib64/libdaplcma.so.1 /usr/lib64/libdaplcma.so
```

Configure network devices

To configure a network device, you can use "ip addr" but the configuration would not be kept for the next reboot, so we need to create on compute nodes a file named after the interface "ib0" : "/etc/sysconfig/network-scripts/ifcfg-ib0".
I want a 192.168.20.* for ib0 just like i have a 192.168.10.* for eth1, the file for node1 would look like :

```
DEVICE="ib0"
ONBOOT="yes"
BOOTPROTO="none"
IPADDR="192.168.20.101"
NETMASK="255.255.255.0"
NETWORK="192.168.20.0"
BROADCAST="192.168.20.255"
```

Configure network devices - script

I just need to take ifcfg-eth1 as a template, replace eth1 by ib0, *.10.* IP by *.20.*., remove the HWADDR line and write the file ifcfg-ib0.
Let's write a simple script (in the shared folder) "/cluster/eth2ib" :

```
cat /etc/sysconfig/network-scripts/ifcfg-eth1 \
  | awk '{sub(/\.10\./,".20.");sub(/eth1/,"ib0");print}' \
  | grep -v HWADDR \
  > /etc/sysconfig/network-scripts/ifcfg-ib0
```

And run the script on each node (you need to be root) :

```
cexecs sh /cluster/eth2ib
```

Configure network devices - route

Another thing : the route. We have 2 ways to go between nodes : eth1 or ib0. It may be a good idea to specify routes precisely.
You just need to create 2 files "route-ib0" and "route-eth1" with a single line, and propagate them to nodes. Both files are the same for all nodes.

```
echo "192.168.20.0/24" > /tmp/route-ib0
echo "192.168.10.0/24" > /tmp/route-eth1
cd /etc/sysconfig/network-scripts/
cexec cp /tmp/route-eth1 .
cexec cp /tmp/route-ib0  .
cexecs /etc/init.d/network restart # restart network to apply
```

You may have a working infiniband setup, or you may just have trashed the entire cluster network configuration. It would be annoying because everything you do on nodes is using the network. In this case solving the issue directly on the node or reinstalling everything are valid solutions.

Configure hostnames

You have a (working ?) IP over infiniband network, but hostnames are always useful too. We don't want to change the node machine names, ("node1", "node2", ...), just add a name for the network devices ("ibnode1", "ibnode2", ...).
To do that we take the "/etc/hosts" file as template, consider the lines containing the work "node", and create a line where we replaced the ethernet IP/name by the infiniband IP/name. Then add the new lines to "/etc/hosts" :

```
grep node /etc/hosts
  | awk '{sub(/\.10\./,".20.");gsub(/node/,"ibnode");print}' \
  > /tmp/morehosts
cat /tmp/morehosts >> /etc/hosts
```

IP over Infiniband IP/naming test

Let's test :

- Connect from head to node1 : "ssh node1"
  (using ethernet)
- Then connect from node1 to ibnode2 : "ssh ibnode2"
  (using infiniband, ibnode1 and node1 are the same machine)
- It works !

Pushing the limits

The operating system is fixing limits for the execution of all your softwares, and some of them may be too strict for a typical HPC software. You can't change the limit directly, you first have to allow the change in a file as root and deploy it.
Let's get the file from a node, change the "max mem lock" limit to unlimited for both "hard" and "soft" then redeploy. As root :

```
cd /cluster
scp node1:/etc/security/limits.conf .
echo "*  soft  memlock  unlimited" >> limits.conf
echo "*  hard  memlock  unlimited" >> limits.conf
cexec cp /etc/security/limits.conf /etc/security/limits.ori
cexec cp /cluster/limits.conf  /etc/security/
```

Then relog as user paul on node1 and ask the limit :

```
ulimit -l
  unlimited
# OK !
```

OpenSM daemon

OpenSM is an InfiniBand compliant Subnet Manager and Administration, and runs on top of OpenIB. You need to run this daemon on the nodes, and make it part of the daemons started by default (as root on head) :

```
cexec chkconfig opensmd on
cexec /etc/init.d/opensmd start
```

Note : if you forget to do that, the providers will have strange default names and your MPI software won't run. You may force the MPI layer to work with the devices using the flag :

```
-env I_MPI_CHECK_DAPL_PROVIDER_MISMATCH none
```

MPI Infiniband Settings

- In the first MPI example, we created a "mpd.hosts" file in my home folder and gave the filename to mpdboot. Another way to do it is to create a file also called "mpd.hosts" but in my working directory. The contents are different : I had "node1", "node2", ... now I want "ibnode1", "ibnode2".
- In the MPI using the ethernet network I was launching the main process from the head node, but for infiniband I need to launch my software from a compute node, all processes need to communicate over the infiniband network.

MPI HelloWorld - Infiniband

Typically the MPI layer is supposed to take advantage of infiniband between nodes (if present and properly configured) and shared memory locally ("rdssm"). But it's better to ask specifically for infiniband "rdma", disable the fallback device and see if it works (you can later remove the switch to use the default "rdssm") :

```
mpirun -r ssh -n 2 \
  -env I_MPI_FALLBACK_DEVICE disable \
  -env I_MPI_DEVICE rdma \
  ./hello

  Hello world from process 1 of 2
  Hello world from process 2 of 2
  # It works !
```

MPI HelloWorld - More Infiniband

Let's ask for more hosts with, high debug level and grep lines of interest :

```
mpirun -r ssh -n 10 -env I_MPI_DEBUG 10 \
  -env I_MPI_FALLBACK_DEVICE disable \
  -env I_MPI_DEVICE rdma \
  ./hello | grep "pinned"

[8] MPI Startup(): process is pinned to CPU00 on node node7.clus
[9] MPI Startup(): process is pinned to CPU01 on node node1.clus
...
# OK !

# and grep "transfer mode" :
[7] MPI startup(): RDMA data transfer mode
[8] MPI startup(): RDMA data transfer mode
...
# OK !
```

mpiBLAST - Build

You need mpiBLAST-1.5.0-pio.tgz from mpiblast.org, the NCBI toolbox is included in the mpiBLAST archive, as user on head node :

```
mkdir -p /cluster/mpiblast/src ; cd !$
# get mpiBLAST-1.5.0-pio.tgz ...
tar -cvzf mpiBLAST-1.5.0-pio.tgz
cd mpiBLAST-1.5.0-pio
./ncbi/make/makedis.csh # had to launch it twice
./configure --prefix=/cluster/blast/mpiblast
make ; make install
ldd /cluster/blast/mpiblast/bin/mpiblast|grep intel
  libmpigc4.so.3.2 => /cluster/intel/impi/3.1/lib64/libmpigc4.s
  libmpi_dbg.so.3.2 => /cluster/intel/impi/3.1/lib64/libmpi_dbg
  libmpigf.so.3.2 => /cluster/intel/impi/3.1/lib64/libmpigf.so.
# OK you have an intel MPI binary
```

mpiBLAST - Input file format

Configure mpiBLAST creating a file ".ncbirc" :

```
[mpiBLAST]
Shared=/cluster/blast/sapiens/tmp/
Local=/tmp/
```

The goal of blast is to map one or several small sequences against huge sequences (typically genomes). The input are a FASTA file (.fa) and a "database". The database is a binary file generated from a huge fasta file. Here we won't generate a single one but smaller distributed databases, so that each MPI process can load a single file smaller in memory. I have 48 cores so I create 48 small databases. As user on head node :

```
mkdir /cluster/blast/sapiens ; cd !$
# copy the all the .fa.* files in the current directory
cat hs_ref_chr* | gzip -d > hs_ref_all.fa
../mpiblast/bin/mpiformatdb -N 48 -i hs_ref_all.fa -o T
  Created 48 fragments.
```

mpiBLAST - Run

Now I need to run mpiBLAST over infiniband from node1, the host list "mpd.hosts" is taken automatically from the working directory (my home folder), as user on node1 :

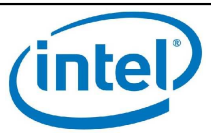
```
mpirun -r ssh -n 50 -env I_MPI_DEBUG 10 \
  /cluster/blast/mpiblast/bin/mpiblast -p blastn \
  -d hs_ref_all.fa -i /cluster/blast/sapiens/seqs.fa \
  -o /cluster/blast/sapiens/blast_results.txt > mpi.log
```

I've launched 50 processes for 48 working processes because mpiBLAST is built with 2 management processes. You can check the output.

mpiBLAST - MPI Trace Collector

So we have a working MPI software, but unless you are a developper you know little about the communication pattern of your software. You need to trace your software with a tool, Intel Trace Analyzer and Collector ® "ITAC" is one of them, part of Intel Cluster Tools, so the environment is already set by the ictvars.sh script. You just need to let "itcpin" run your software and instrument the MPI calls.

```
mpirun -r ssh -n 40 \
  -env I_MPI_FALLBACK_DEVICE disable -env I_MPI_DEVICE rdma \
  itcpin --run --insert libVT -- \
  /cluster/blast/mpiblast/bin/mpiblast -p blastn \
  -d hs_ref_all.fa -i /cluster/blast/sapiens/seqs.fa \
  -o /cluster/blast/sapiens/blast_results.txt
Total Execution Time: 42.5881
[0] Intel(R) Trace Collector INFO:
Writing tracefile mpiblast.stf in /home/paul
```



mpiBLAST - MPI Trace Analyzer

Launch Intel Trace Analyzer

You just need to launch "traceanalyzer mpiblast.stf" (graphical tool, you need X window running).
I've included two screenshots, ScreenShot t1 and t2, to show the timeline and interprocess communication.

mpiBLAST - MPI Trace Analyzer

Two management processes

From the first view you can see the 2 management processes at the top, one for the MPI work, one for the IO work (IO is not visible here, I know there's a IO process from the mpiBLAST documentation).

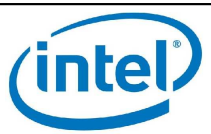
Load balance

You can see that some processes finish early, waiting. Data granularity is far from perfect in our very simple workload, we would have to launch a real load to test the software.

NCBI BLAST

We've seen a MPI example with mpiBLAST, where a single task is sliced and spreaded on a cluster with MPI. Sometimes we need to do that if the database is so huge a single machine can't load it.
But we'd like to use our cluster to spread independant calculations on independant data. A typical genome can be loaded on a regular server nowadays. In this case we need to use the regular NCBI blast, with a regular database binary. Let's compile the NCBI toolbox :

```
cd /cluster/blast
wget ftp://ftp.ncbi.nih.gov/toolbox/ncbi_tools/ncbi.tar.gz
tar -xvzf ncbi.tar.gz
cp ncbi/platform/linux_icc9.ncbi.mk ncbi/platform/linux.ncbi.mk
./ncbi/make/makedis.csh
ll ncbi/build/blastall
  -rwxrwxr-x 2 paul cluster_users 8681956 Apr 17 18:55 ncbi/bui
# OK
```



Queuing and software caracterization

Why do I need queuing ?

You can't afford to launch your software on a production cluster just like you would on a regular workstation, many users have to share a single resource to launch complex requests "jobs". That's why you need a queuing system to keep track of tasks, users, working files, priorities.

What do I need to queue a job ?

Basically you create a script to launch your application, and give the system a few details about your software execution, then run "qsub" with your parameters and script name.
You should give : number of cores per process, memory and local disk usage, and an estimation of the CPU execution time.

Single run

Add 2 folders definitions in the " /.ncbirc" file :

```
[NCBI]
DATA=/cluster/blast/ncbi/data
[BLAST]
BLASTDB=/cluster/blast/sapiens
```

Prepare the data and run (for a quad-core machine) :

```
cd /cluster/blast/sapiens
../ncbi/build/formatdb -i hs_ref_all.fa -p F -o T
time ../ncbi/build/blastall -a 4 -p blastn \
  -d hs_ref_all.fa -i seqs.fa -o ouput.txt
```

With top you see you need more than 2Go of memory, 4 cores and 30 minutes total. The output is a small file.

Batch run

During the execution, you can access variables like $PBS_JOBID interesting if you want to create a unique file output.
Example blast.sh file in your home folder :

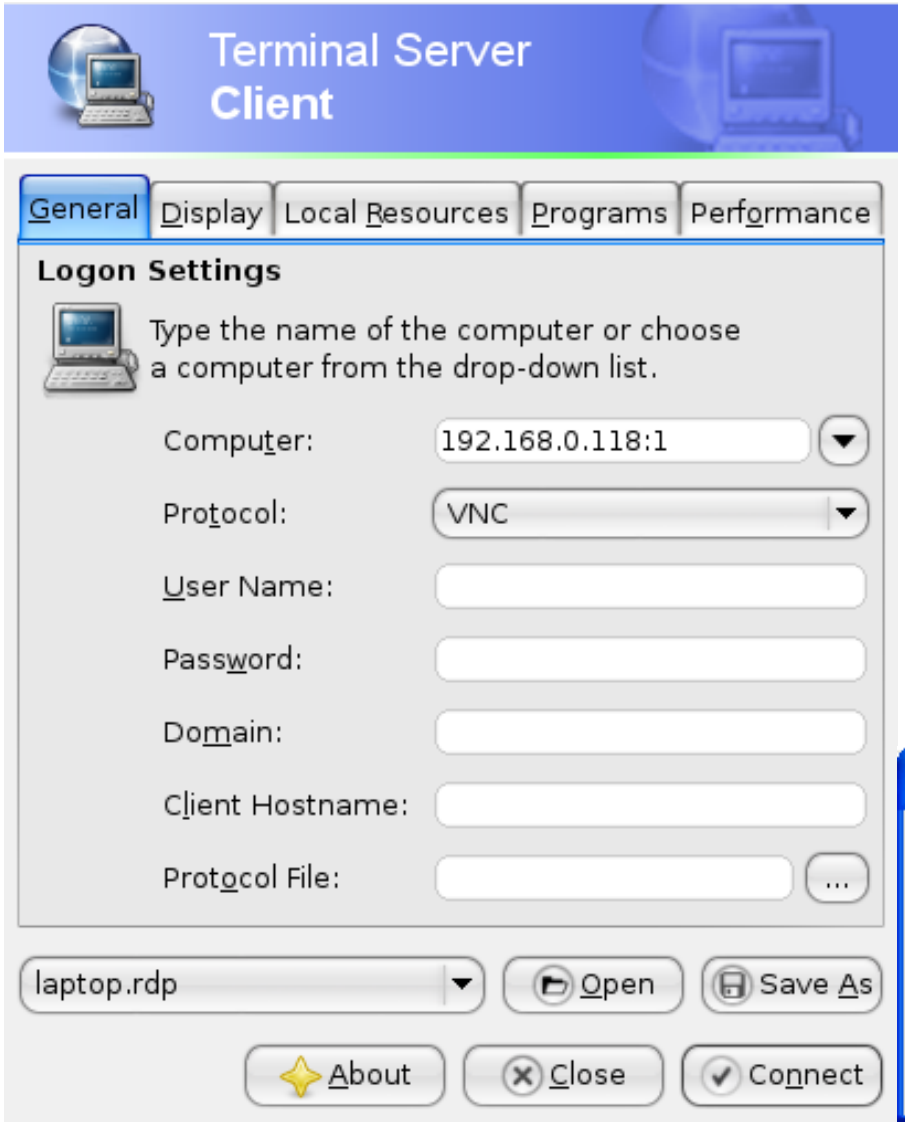
```
/cluster/blast/ncbi/build/blastall -a 4 -p blastn \
  -d hs_ref_all.fa -i /cluster/blast/sapiens/seqs.fa
  -o ouput.$PBS_JOBID.txt
```

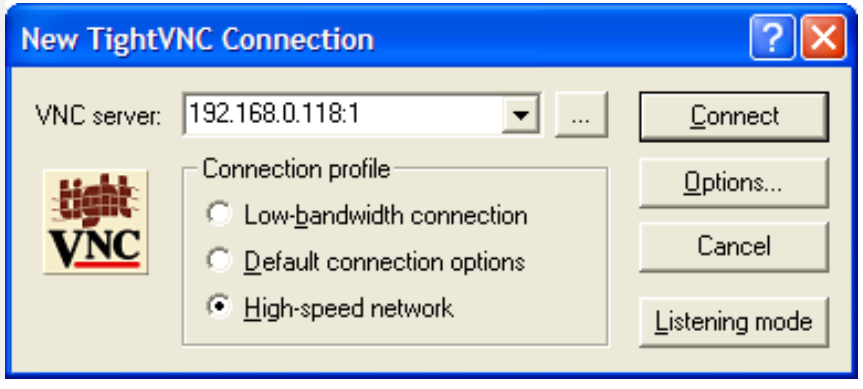
Launch and monitor :

```
qsub -l ppn=4,mem=3gb,cput=01:00:00 /home/paul/blast.sh
# you can launch several times the script file
```

To monitor your jobs, "showq" from maui is better than the basic "qstat". ScreenShots t3 and t4.
Check the web based cluster monitor "ganglia" available from http://yourservername/ganglia : ScreenShot t5.

OS Installation - 1



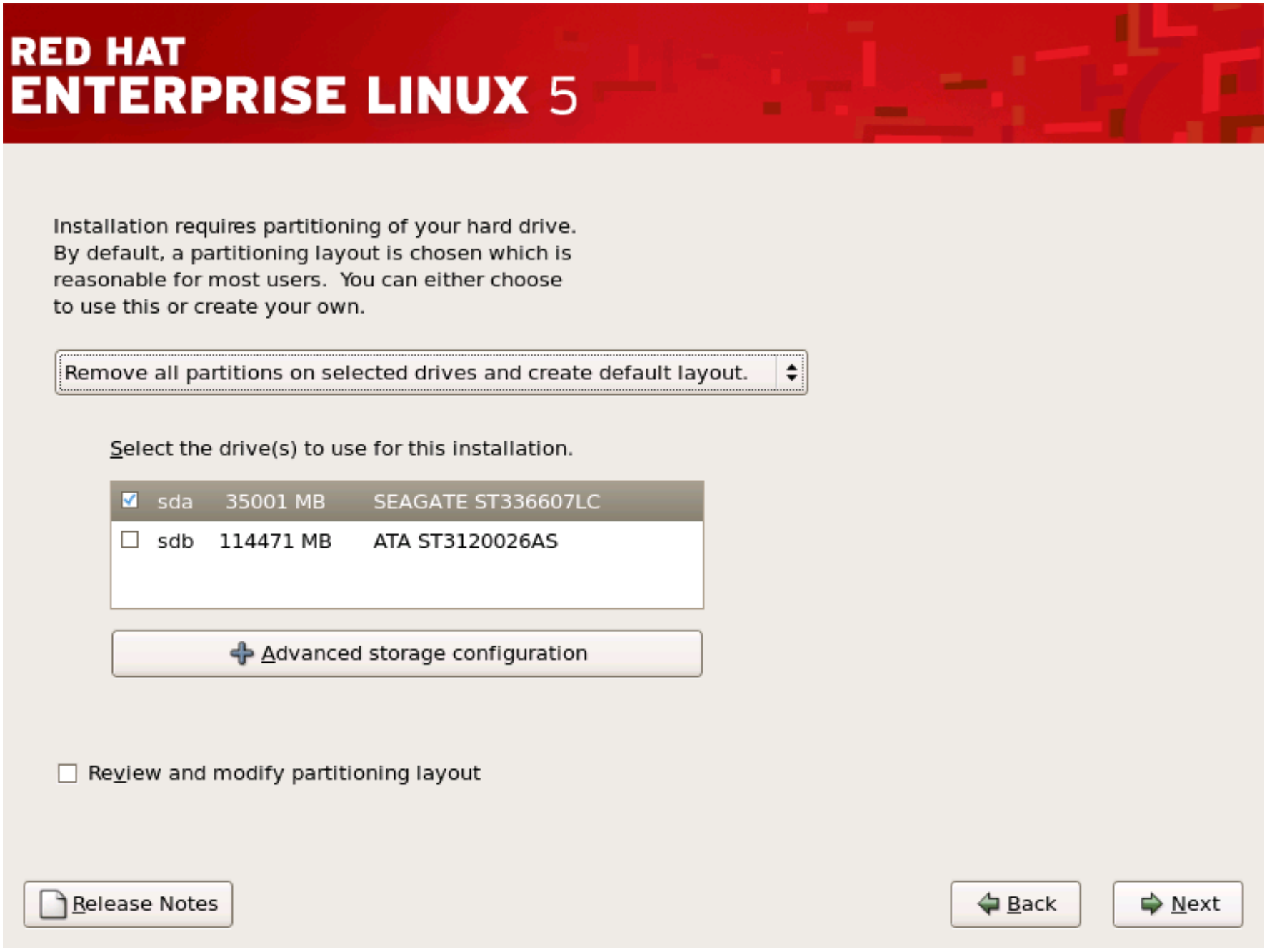


OS Installation - 2

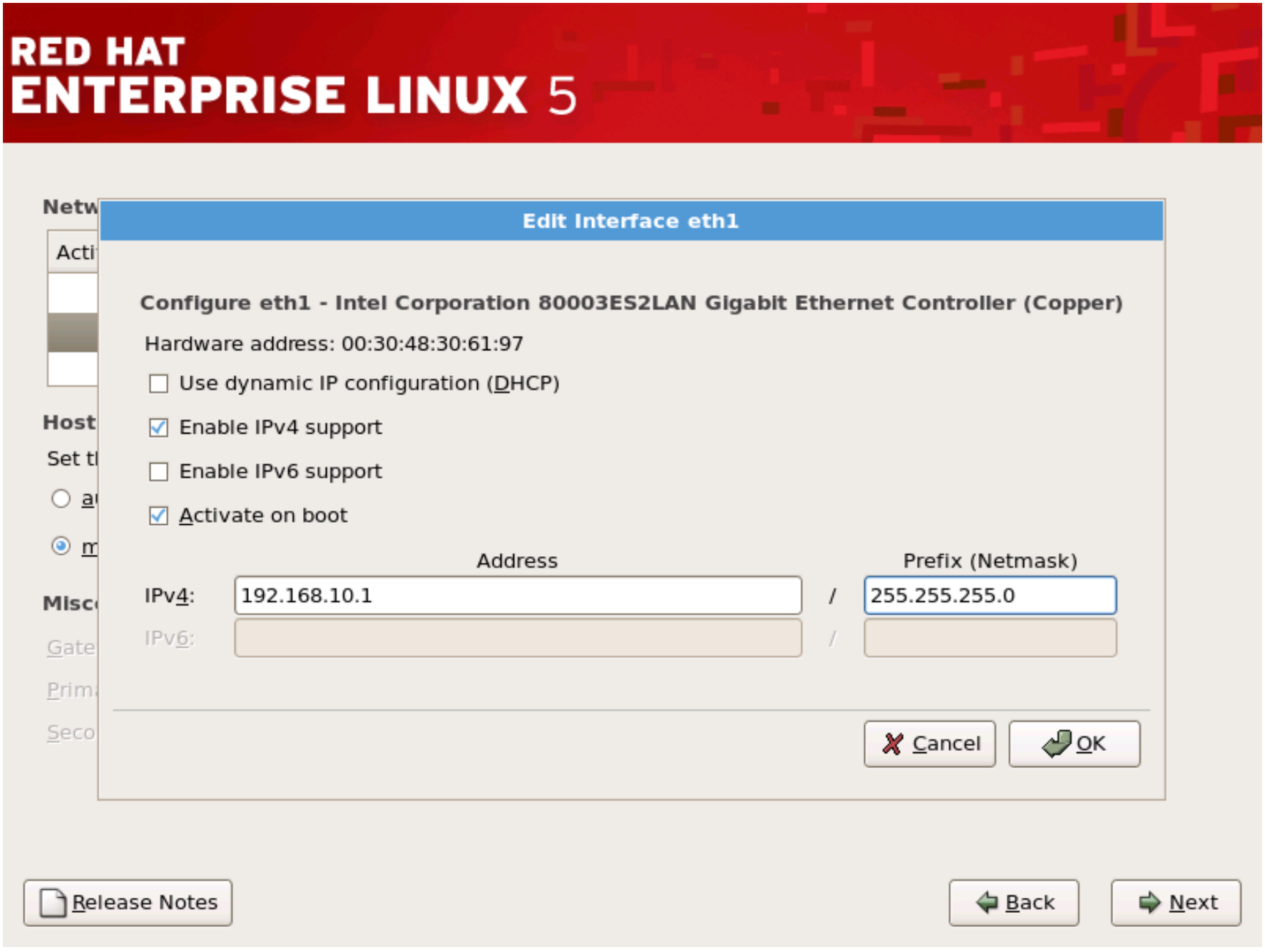
OS Installation - 3

OS Installation - 4

OS Installation - 5



OS Installation - 6

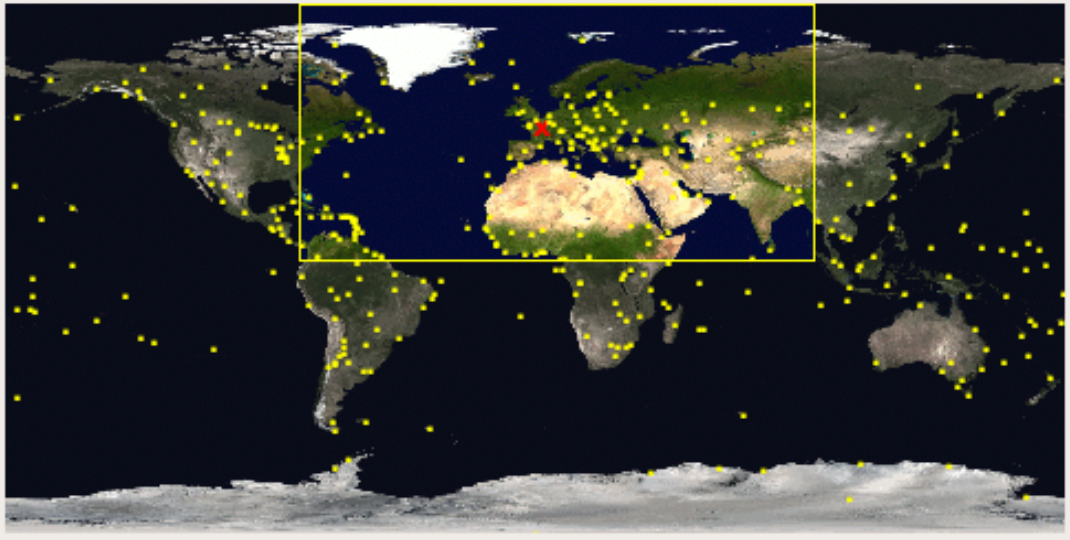


OS Installation - 7

OS Installation - 8

RED HAT
ENTERPRISE LINUX 5

Please click into the map to choose a region:



Europe/Paris

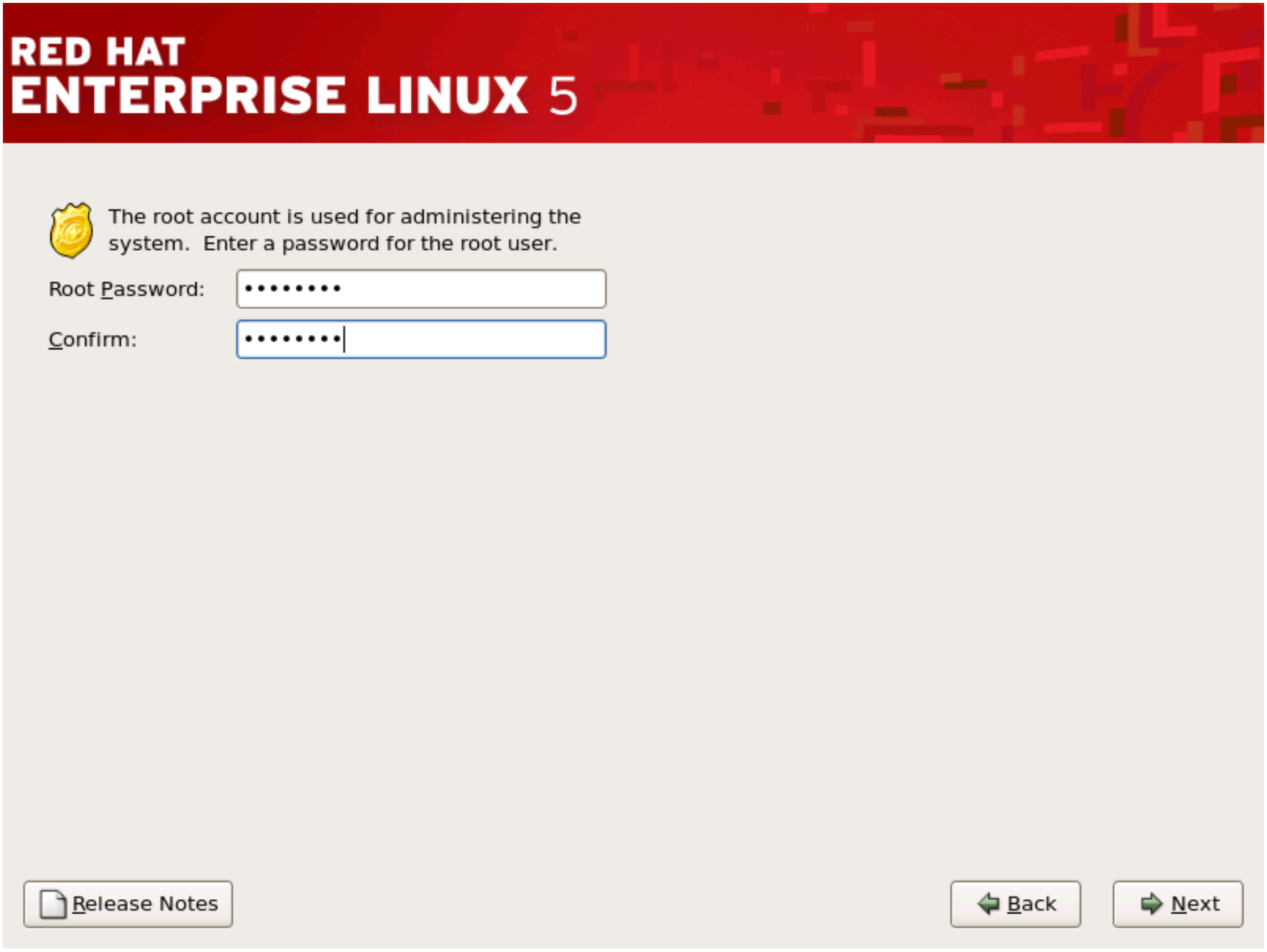
☑ System clock uses UTC

Release Notes

Back

Next

OS Installation - 9



OS Installation - 10

The default installation of Red Hat Enterprise Linux Server includes a set of software applicable for general internet usage. What additional tasks would you like your system to include support for?

- [x] Software Development
- [ ] Web server

You can further customize the software selection now, or after install via the software management application.

( ) Customize later (•) Customize now

Release Notes Back Next

OS Installation - 11

RED HAT
ENTERPRISE LINUX 5

Click next to begin installation of Red Hat Enterprise Linux Server.

A complete log of the installation can be found in the file '/root/install.log' after rebooting your system.

A kickstart file containing the installation options selected can be found in the file '/root/anaconda-ks.cfg' after rebooting the system.

Release Notes

Back

Next


RED HAT
ENTERPRISE LINUX 5
redhat
Installing db4-4.3.29-9.fc6.i386 (2 MB)
The Berkeley DB database library (version 4) for C.
Remaining time: 8 minutes
Release Notes
Back
Next

RED HAT
ENTERPRISE LINUX 5

Congratulations, the installation is complete.

Remove any media used during the installation process and press the "Reboot" button to reboot your system.

Release Notes

Back

Reboot

OSCAR Installation - 1

Welcome to the OSCAR Wizard!

OSCAR Version: 5.1b1

- INSTALL MODE -

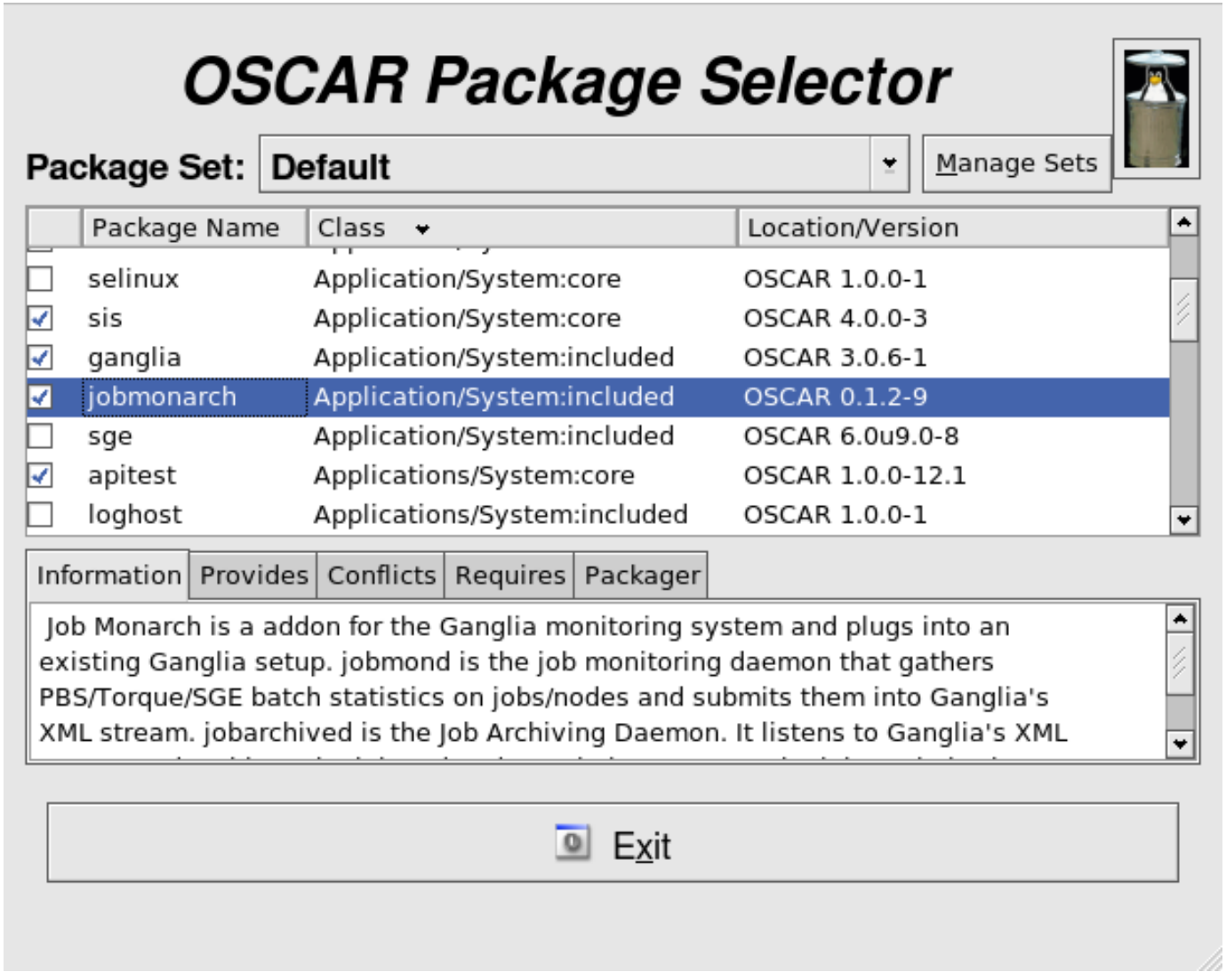
Step 0:	Manage OSCAR Repositories...	Help
Step 1:	Select OSCAR Packages To Install...	Help
Step 2:	Configure Selected OSCAR Packages...	Help
Step 3:	Install OSCAR Server Packages	Help
Step 4:	Build OSCAR Client Image...	Help
Step 5:	Define OSCAR Clients...	Help
Step 6:	Setup Networking...	Help
	Delete OSCAR Clients...	Help
	Monitor Cluster Deployment	Help

Before continuing, network boot all of your nodes.
Once they have completed installation, reboot them from the hard drive. Once all the machines and their ethernet adaptors are up, move on to the next step.

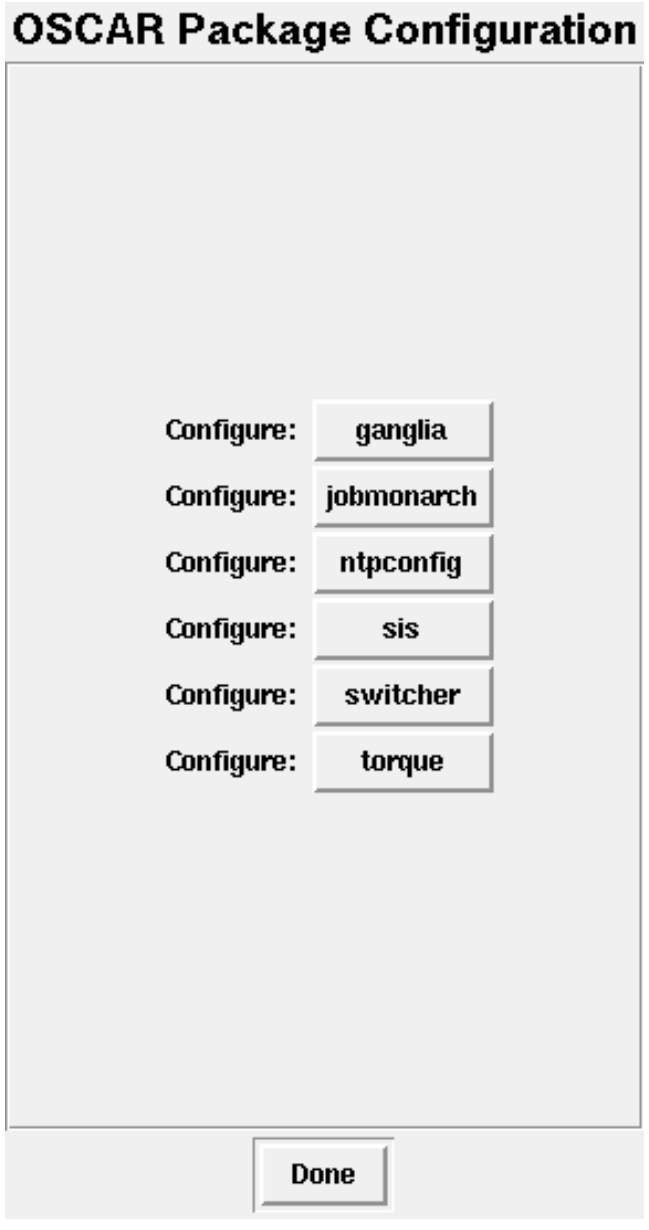
Step 7:	Complete Cluster Setup	Help
Step 8:	Test Cluster Setup	Help

Quit

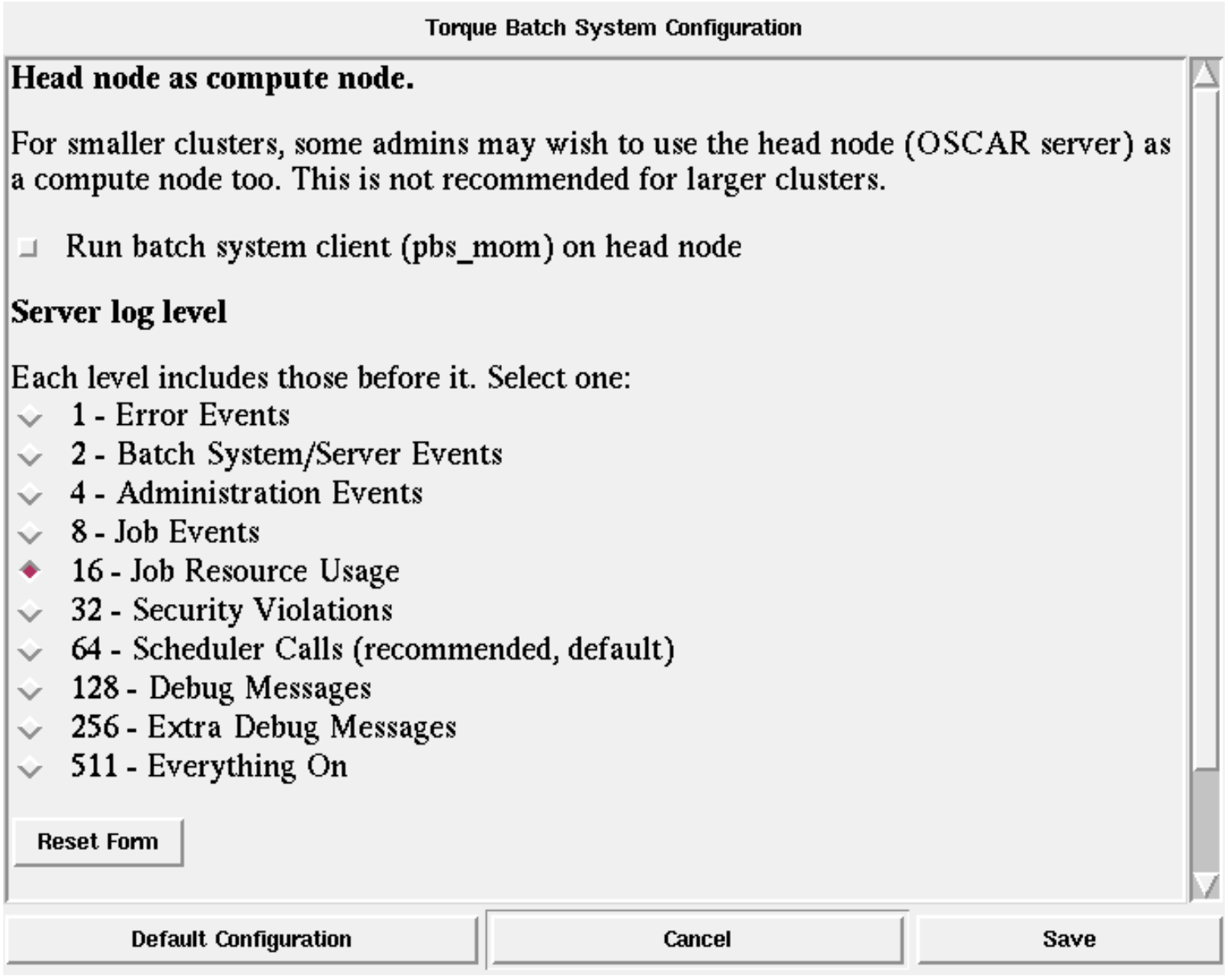
OSCAR Installation - 2



OSCAR Installation - 3

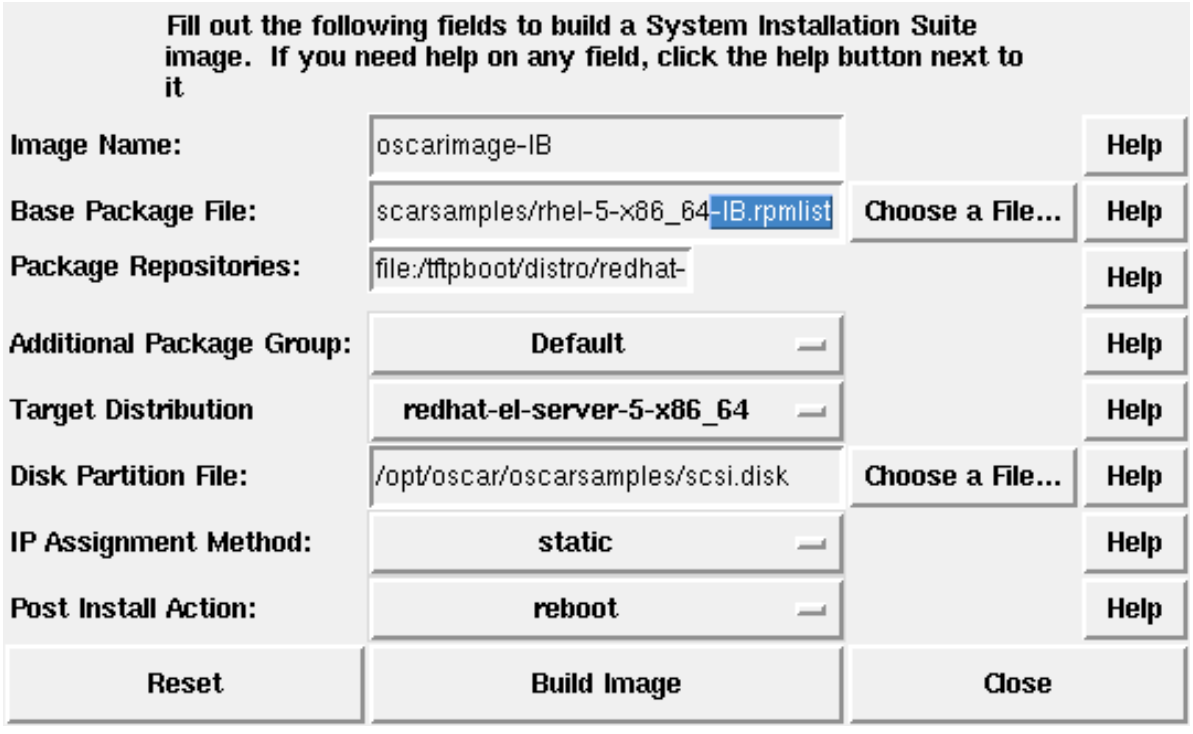


OSCAR Installation - 4

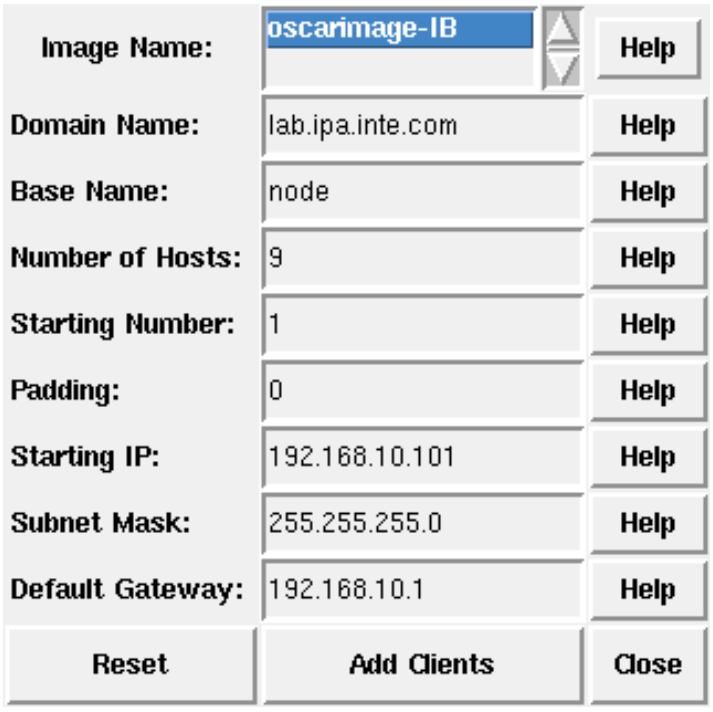


OSCAR Installation - 5

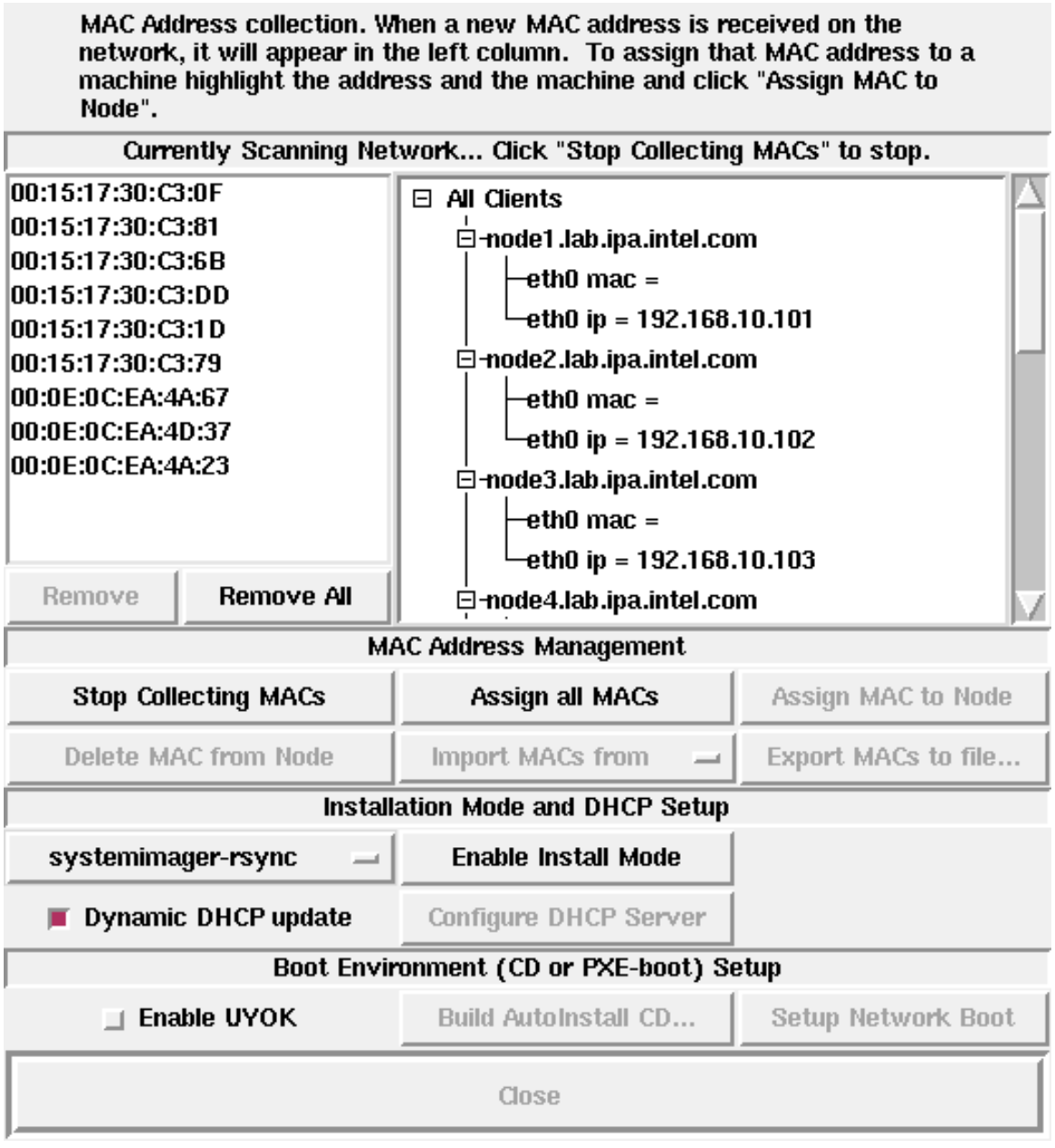
OSCAR Installation - 6



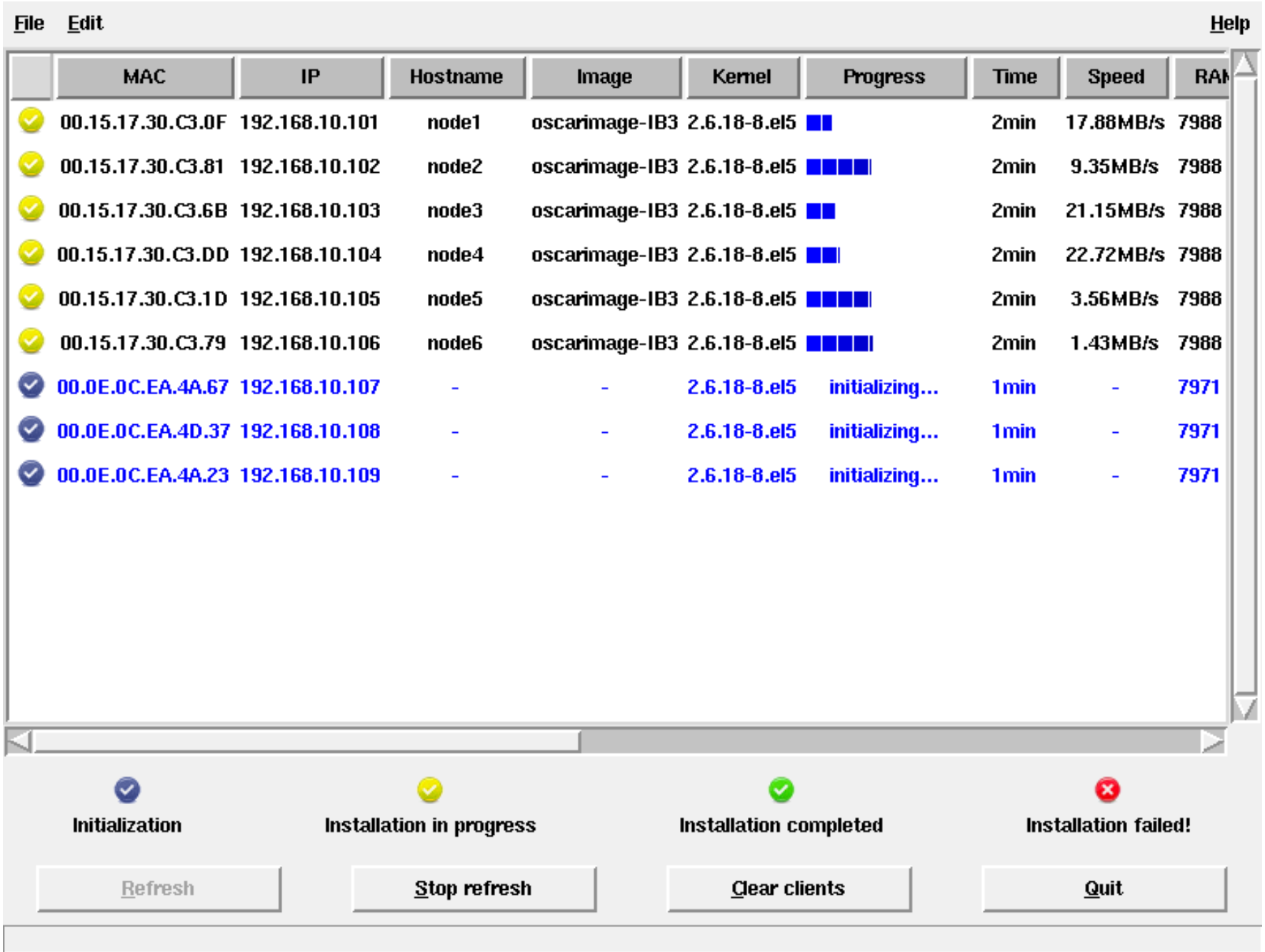
OSCAR Installation - 7



OSCAR Installation - 8



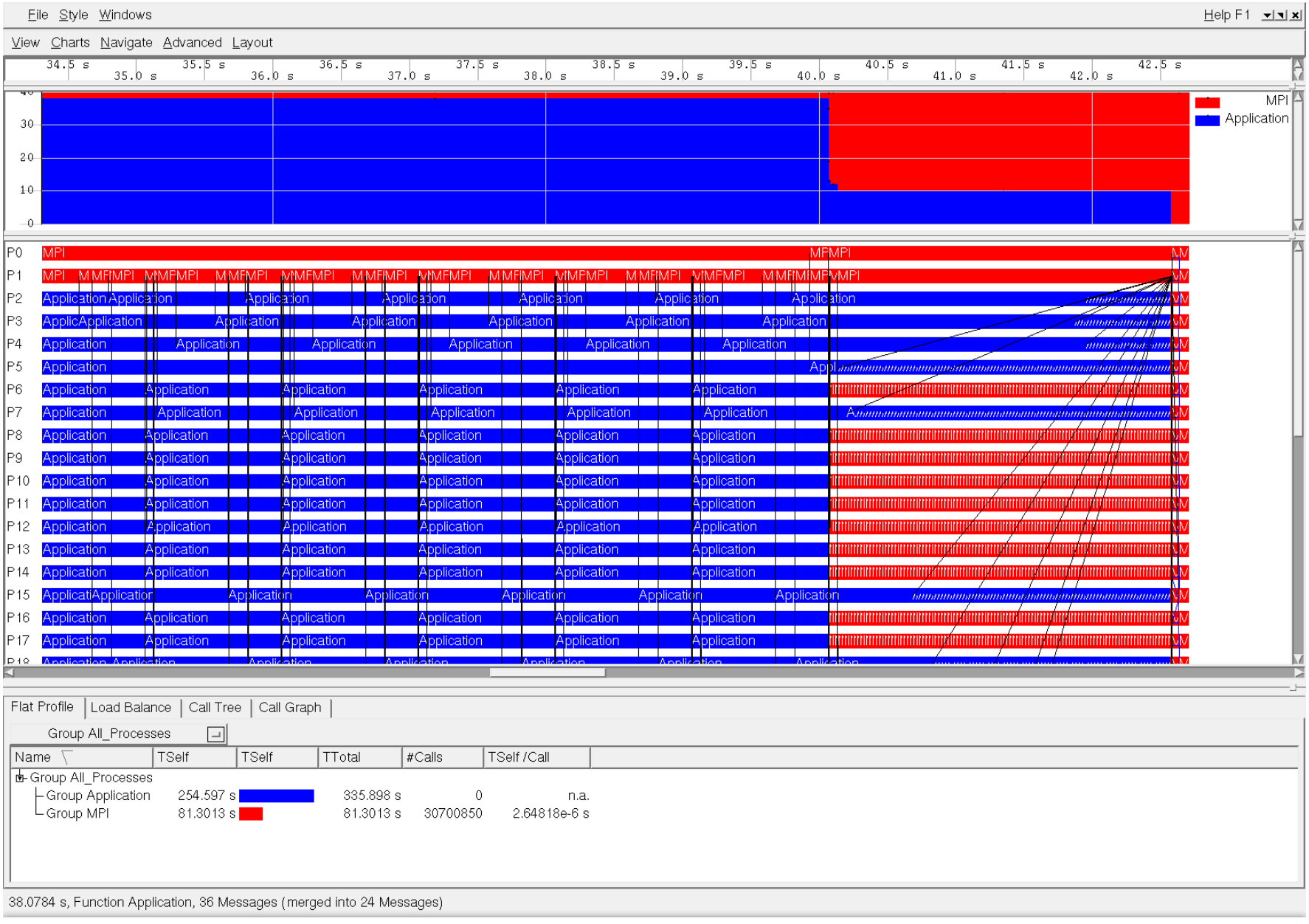
OSCAR Installation - 9



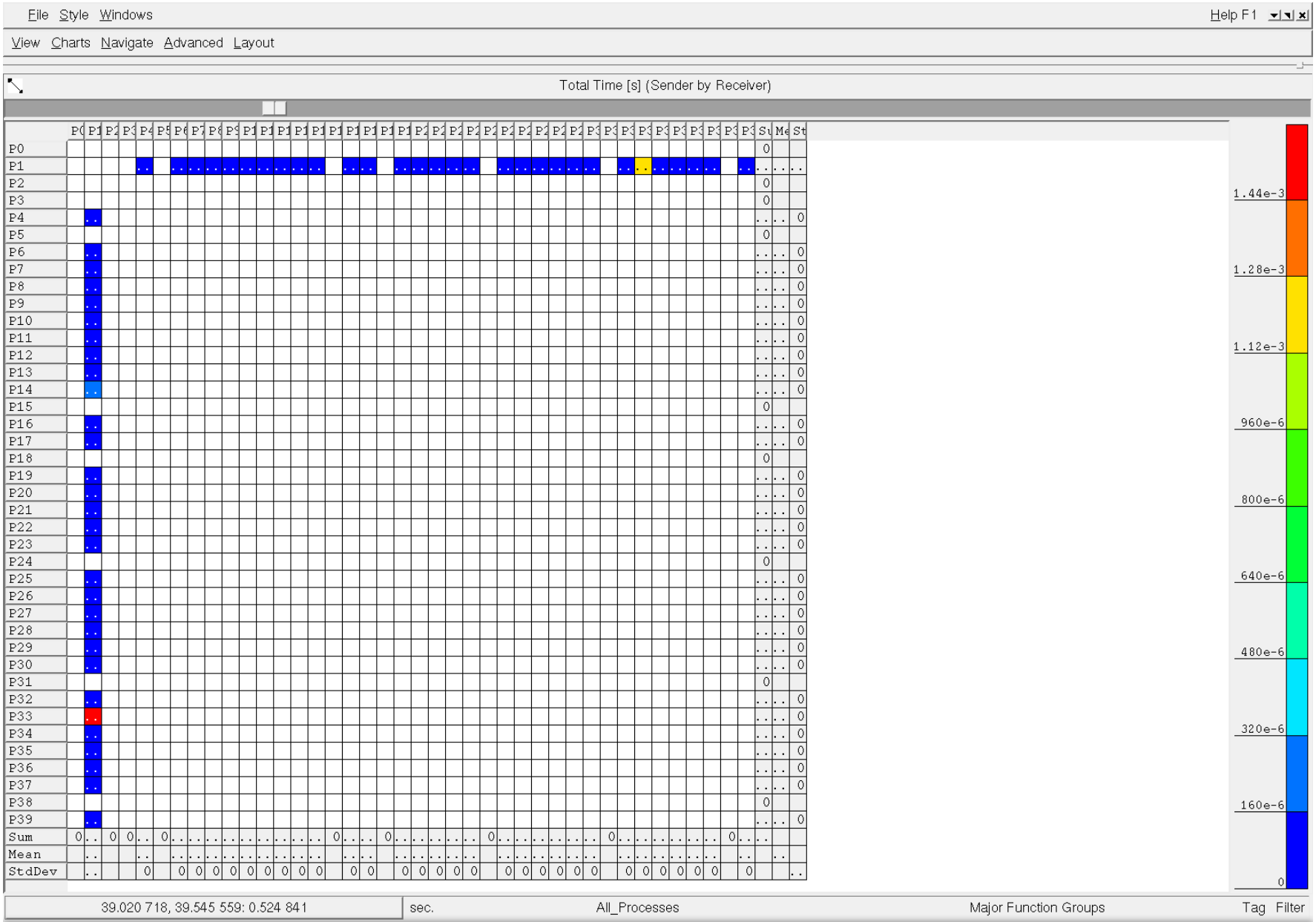
OSCAR Installation - 10

```
Performing root tests...
Maui service check:maui                                          [PASSED]
TORQUE node check                                                [PASSED]
TORQUE service check:pbs_server                                  [PASSED]
/home mounts                                                     [PASSED]

Preparing user tests...
Performing user tests...
SSH ping test                                                    [PASSED]
SSH server->node                                                 [PASSED]
SSH node->server                                                 [PASSED]
TORQUE default queue definition                                  [PASSED]
TORQUE Shell Test                                                [PASSED]
LAM/MPI (via TORQUE)                                             [PASSED]
MPICH (via TORQUE)                                               [PASSED]
Open MPI (via TORQUE)                                            [PASSED]
Ganglia setup test                                               [PASSED]
Ganglia node count test                                          [PASSED]
PVM (via TORQUE)                                                 [PASSED]

Run APItests...

Running Installation tests for pvm
[PASS]       2008-03-10 17:58:29   pvmd-path-ls.apt
[PASS]       2008-03-10 17:58:29   envvar-pvm_arch.apt
[PASS]       2008-03-10 17:58:29   envvar-pvm_root.apt
[PASS]       2008-03-10 17:58:29   envvar.apb
[PASS]       2008-03-10 17:58:29   pvmd-path-which.apt
[PASS]       2008-03-10 17:58:29   modulecmd-path-ls.apt
[PASS]       2008-03-10 17:58:29   pvm-module-list.apt
[PASS]       2008-03-10 17:58:29   pvm-module-show-pvm_rsh.apt
[PASS]       2008-03-10 17:58:29   pvm-module-show-pvm_arch.apt
[PASS]       2008-03-10 17:58:29   pvm-module-show-pvm_root.apt
[PASS]       2008-03-10 17:58:29   pvm-module-show.apb
[PASS]       2008-03-10 17:58:29   pvm-module.apb
[PASS]       2008-03-10 17:58:29   install_tests.apb

All tests passed, your OSCAR cluster is now ready to compute!

Please consider registering your OSCAR cluster at:
http://oscar.openclustergroup.org/register

...Hit <ENTER> to close this window...
```

Tools - 1


File Style Windows
Help F1
View Charts Navigate Advanced Layout
34.5 s
35.0 s
35.5 s
36.0 s
36.5 s
37.0 s
37.5 s
38.0 s
38.5 s
39.0 s
39.5 s
40.0 s
40.5 s
41.0 s
41.5 s
42.0 s
42.5 s
40
30
20
10
0
MPI
Application
P0
P1
P2
P3
P4
P5
P6
P7
P8
P9
P10
P11
P12
P13
P14
P15
P16
P17
Flat Profile
Load Balance
Call Tree
Call Graph
Group All_Processes
Name
TSelf
TSelf
TTotal
#Calls
TSelf /Call
Group All_Processes
Group Application
254.597 s
335.898 s
0
n.a.
Group MPI
81.3013 s
81.3013 s
30700850
2.64818e-6 s
38.0784 s, Function Application, 36 Messages (merged into 24 Messages)

Tools - 2


File Style Windows
Help F1
View Charts Navigate Advanced Layout
Total Time [s] (Sender by Receiver)
1.44e-3
1.28e-3
1.12e-3
960e-6
800e-6
640e-6
480e-6
320e-6
160e-6
0
39.020 718, 39.545 559: 0.524 841
sec.
All_Processes
Major Function Groups
Tag Filter

Tools - 3

```
[paul@head ~]$ qstat
Job id                    Name             User            Time Use S Queue
------------------------- ---------------- --------------- -------- - -----
284.head                  testpbs          paul            00:00:00 E workq
285.head                  testpbs          paul                   0 R workq
286.head                  testpbs          paul                   0 R workq
287.head                  testpbs          paul                   0 R workq
288.head                  testpbs          paul                   0 Q workq
289.head                  testpbs          paul                   0 Q workq
290.head                  testpbs          paul                   0 Q workq
291.head                  testpbs          paul                   0 Q workq
292.head                  testpbs          paul                   0 Q workq
293.head                  testpbs          paul                   0 Q workq
294.head                  testpbs          paul                   0 Q workq
295.head                  testpbs          paul                   0 Q workq
296.head                  testpbs          paul                   0 Q workq
297.head                  testpbs          paul                   0 Q workq
298.head                  testpbs          paul                   0 Q workq
299.head                  testpbs          paul                   0 Q workq
300.head                  testpbs          paul                   0 Q workq
301.head                  testpbs          paul                   0 Q workq
302.head                  testpbs          paul                   0 Q workq
303.head                  testpbs          paul                   0 Q workq
304.head                  testpbs          paul                   0 Q workq
305.head                  testpbs          paul                   0 Q workq
```

Tools - 4

```
[paul@head ~]$ /opt/maui/bin/showq
ACTIVE JOBS--------------------
JOBNAME            USERNAME      STATE  PROC   REMAINING            STARTTIME

276                   paul    Running     9    INFINITY  Thu Apr 17 15:59:09
277                   paul    Running     9    INFINITY  Thu Apr 17 15:59:09
278                   paul    Running     9    INFINITY  Thu Apr 17 15:59:09
279                   paul    Running     9    INFINITY  Thu Apr 17 15:59:09

     4 Active Jobs       36 of   36 Processors Active (100.00%)
                          8 of    8 Nodes Active      (100.00%)

IDLE JOBS----------------------
JOBNAME            USERNAME      STATE  PROC     WCLIMIT            QUEUETIME

280                   paul       Idle     9    INFINITY  Thu Apr 17 15:57:40

1 Idle Job

BLOCKED JOBS----------------
JOBNAME            USERNAME      STATE  PROC     WCLIMIT            QUEUETIME


Total Jobs: 5   Active Jobs: 4   Idle Jobs: 1   Blocked Jobs: 0
[paul@head ~]$
```

Tools - 5


File Edit View History Bookmarks Tools Help
http://capraia/ganglia/
Google
OSCAR PRO
Cluster Report for Thu, 17 Apr 2008 13:00:02 +0200
Get Fresh Data
Metric load_one
Last hour
Sorted descending
Physical View
cluster Grid > Intel iPa lab cluster > --Choose a Node
Overview of Intel iPa lab cluster
CPUs Total: 44
Hosts up: 10
Hosts down: 0
Avg Load (15, 5, 1m):
9%, 9%, 9%
Localtime:
2008-04-17 12:59
Joblist
Cluster Load Percentages
25-50 (40.00%)
0-25 (60.00%)
Intel iPa lab cluster Load last hour
Load/Procs
1-min Load
Nodes
CPUs
Running Processes
Intel iPa lab cluster CPU last hour
Percent
User CPU
Nice CPU
System CPU
WAIT CPU
Idle CPU
Intel iPa lab cluster Memory last hour
Bytes
Memory Used
Memory Shared
Memory Cached
Memory Buffered
Memory Swapped
Total In-Core Memory
Intel iPa lab cluster Network last hour
Bytes/sec
In
Out
Show Hosts: yes no | Intel iPa lab cluster load_one last hour sorted descending | Columns 4
node5.cluster.lab.ipa.intel.com
node7.cluster.lab.ipa.intel.com
node3.cluster.lab.ipa.intel.com
node4.cluster.lab.ipa.intel.com
node8.cluster.lab.ipa.intel.com
node9.cluster.lab.ipa.intel.com
node1.cluster.lab.ipa.intel.com
node6.cluster.lab.ipa.intel.com
node2.cluster.lab.ipa.intel.com
head
load_one last hour (now 0.00)
Done

Alternatives

Rocks

Another easy way to get a cluster running in a short time, Rocks comes a full OS+tools package.

Parallel Knoppix

Boot CD with nodes, monitoring, MPI, PVM, ... autoconfiguration. You don't need to change the contents of your hard drive. Easy to reproduce.

Cluster Knoppix

Boot CD with nodes, monitoring, openmosix ... autoconfiguration. You don't need to change the contents of your hard drive. Easy to reproduce. Using openmosix, your processes can migrate from one node to another automatically using shared memory on the kernel level.

URLs

- OSCAR Toolkit : `http://oscar.openclustergroup.org`
  (see the official documentation, this document is heavily based on it)
- Intel software : `http://www.intel.com/software`
- OpenFabrics Alliance : `http://www.openfabrics.org`
- Cluster Resources MAUI / TORQUE :
  `http://www.clusterresources.com/pages/resources/documentation.php`
- mpiBLAST : `http://www.mpiblast.org`
- National Center for Biotechnology Information - BLAST :
  `http://www.ncbi.nlm.nih.gov`
- NCBI - Download Homo sapiens genome files :
  `ftp://ftp.ncbi.nih.gov/genomes/H_sapiens`

Thanks

Thanks a lot !

- All OSCAR contributors of course ;-)
- oscar-devel : Michael Edwards, Geoffroy Vallee, Thomas Naughton
- Intel linux guru : Frederic Lepied
- Intel OSCAR : Heller Christopher
- Intel HPC Paris : Philippe Thierry, Laurent Duhem, Alain Dominguez
- Intel MPI : Andrey Derbunovich, Alexander Supalov

Legal

License

Copyright

Trademarks

Contact

Feel free to contact me if you see bugs in this document,
or just to tell me how it works for you :

- Paul Guermonprez
- email : `mailto:paul.guermonprez@intel.com`
- postal : Intel - 2 rue de Paris - 92196 Meudon - France
- `http://softwareblogs.intel.com/author/paul-guermonprez`