OSCAR

Open Source Cluster Application Resources

Stephen L. Scott

Oak Ridge National Laboratory

scottsl@ornl.gov

www.csm.ornl.gov/~sscott

Workshop on: Operating Systems, Tools and Methods for High Performance Computing on Linux Clusters
EDF R&D – Clamart (France)
October 7, 2003
ORNL CS Research

Significant impact and world-wide influence on Parallel computing and the Science enabled by it

- Track record of developing very popular software
  - PVM – 400,000
  - OSCAR – 112,922
  - Cumulvs - 300

- Influencing Standards
  MPI, BLAS, LAPACK, PAPI

- Enabling Science
  PVM, MPI, enote, etc.
  are widely used in education, research, and industry

Goal is to accelerate the process of Scientific Discovery
Over Ten years of leadership in heterogeneous distributed computing

Gordon Bell Award (1st of many won using PVM)

Impact of our research has been recognized by numerous awards

1st release April 01

OSCAR will pass 113,000 downloads

Networks of Workstations

Wide-area GRID experiments

PC Clusters

1989 90 94 96 97 99 2001 2003

PVM R&D 100 SCxx Challenge Awards AMSE Award NetSolve R&D 100

SC92 SC93 SC95 SC96 (2)

SC92

SC93

SC95

SC96

OSCAR becomes most popular cluster software

1st release April 01

Impact of our research has been recognized by numerous awards

Oak Ridge National Laboratory -- U.S. Department of Energy
Scalable Systems Software for Terascale Computer Centers

www.scidac.org/ScalableSystems

Problem

- Computer centers use incompatible, ad hoc set of systems tools
- Present tools are not designed to scale to multi-Teraflop systems

Solution

- Collectively (with industry) define standard interfaces between systems components for interoperability
- Create scalable, standardized management tools for efficiently running our large computing centers

Impact

- Revolutionize the way system software is designed and used.

Oak Ridge National Laboratory  --  U.S. Department of Energy
OSCAR - the beginning
First cluster “distro”

- Extreme Linux
- May 13, 1998
- $29.95 CD
OSCAR Background

• Meeting back in April 2000
  – Cluster assembly is time consuming & repetitive
  – Nice to offer a toolkit to automate
  – First public release in April 2001

• Use “best practices” for HPC clusters
  – Leverage wealth of open source components
  – Target modest size cluster (single network switch)

• Form umbrella organization to oversee
  – Open Cluster Group
Open Cluster Group

• Informal group formed to make cluster computing more practical for HPC research and development

• Membership is open, direct by steering committee
  – Research/Academic
  – Industry

• Current active working groups
  – OSCAR
  – Thin-OSCAR (diskless)
  – HA-OSCAR (high availability)
OSCAR 2003 Core Organizations

- Dell
- IBM
- Intel
- MSC.Software
- Bald Guy Software
- Indiana University
- NCSA
- Oak Ridge National Laboratory
- Université de Sherbrooke
Open Source Cluster Application Resources

What is OSCAR?

- Framework for cluster installation configuration and management
- Common used cluster tools
- Wizard based cluster software installation
  - Operating system
  - Cluster environment
    - Administration
    - Operation

- Automatically configures cluster components
- Increases consistency among cluster builds
- Reduces time to build / install a cluster
- Reduces need for expertise
The OSCAR strategy

- OSCAR is a snap-shot of best-known-methods for building, programming and using clusters of a “reasonable” size.

- To bring uniformity to clusters, foster commercial versions of OSCAR, and make clusters more broadly acceptable.

- Consortium of research, academic & industry members cooperating in the spirit of open source.

Open Source
OSCAR with Linux

Commercially supported Value added instantiations of OSCAR

Other OSCAR Flavors
HA-OSCAR, Thin-OSCAR, SSS-OSCAR, SSI-OSCAR
OSCAR Components

• Administration/Configuration
  – SIS, C3, OPIUM, Kernel-Picker, NTPconfig cluster services (dhcp, nfs, ...)
  – Security: Pfilter, OpenSSH

• HPC Services/Tools
  – Parallel Libs: MPICH, LAM/MPI, PVM
  – OpenPBS/MAUI
  – HDF5
  – Ganglia, Clumon, … [monitoring systems]
  – Other 3\textsuperscript{rd} party OSCAR Packages

• Core Infrastructure/Management
  – System Installation Suite (SIS), Cluster Command & Control (C3), Env-Switcher,
  – OSCAR DAtabase (ODA), OSCAR Package Downloader (OPD)
System Installation Suite (SIS)

Enhancement suite to the SystemImager tool. Adds SystemInstaller and SystemConfigurator

- SystemInstaller – interface to installation, includes a stand-alone GUI – Tksis. Allows for description based image creation.

- SystemImager – base tool used to construct & distribute machine images.

- SystemConfigurator – extension that allows for on-the-fly style configurations once the install reaches the node, e.g. `/etc/modules.conf`.
System Installation Suite (SIS)

- Used in OSCAR to install nodes
  - partitions, formats and installs nodes

- Construct “image” of compute node on headnode
  - Directory structure that is what the node will contain
  - This is a “virtual”, chroot-able environment
    /var/lib/systemimager/images/oscarimage/etc/
    .../usr/

- Use rsync to copy only differences in files, so can be used for cluster management
  - maintain image and sync nodes to image
C3 Power Tools

- Command-line interface for cluster system administration and parallel user tools.

- Parallel execution `cexec`
  - Execute across a single cluster or multiple clusters at same time

- Scatter/gather operations `cpush/cget`
  - Distribute or fetch files for all node(s)/cluster(s)

- Used throughout OSCAR and as underlying mechanism for tools like OPIUM’s `useradd` enhancements.
C3 Power Tools

Example to run hostname on all nodes of default cluster:

$ cexec hostname

Example to push an RPM to /tmp on the first 3 nodes

$ cpush :1-3 helloworld-1.0.i386.rpm /tmp

Example to get a file from node1 and nodes 3-6

$ cget :1,3-6 /tmp/results.dat /tmp

* Can leave off the destination with cget and will use the same location as source.
Switcher

• Switcher provides a clean interface to edit environment without directly tweaking .dot files.
  – e.g. PATH, MANPATH, path for ‘mpicc’, etc.

• Edit/Set at both system and user level.

• Leverages existing Modules system

• Changes are made to future shells
  – To help with “foot injuries” while making shell edits
  – Modules already offers facility for current shell manipulation, but no persistent changes.
OSCAR DAtabase (ODA)

- Used to store OSCAR cluster data
- Currently uses MySQL as DB engine
- User and program friendly interface for database access
- Capability to extend database commands as necessary.
OSCAR Package Downloader (OPD)

Tool to download and extract OSCAR Packages.

- Can be used for timely package updates
- Packages that are not included, i.e. “3rd Party”
- Distribute packages with licensing constraints.
OSCAR Installation
Server Installation and Configuration

- Install Linux on server machine (cluster head node)
  - workstation install w/ software development tools
  - 57-page installation document!
    - (quick install available)
- Download copy of OSCAR and unpack on server
- Configure and install OSCAR on server
  - readies the wizard install process
- Configure server Ethernet adapters
  - public
  - private
- Launch OSCAR Installer (wizard)
### OSCAR Wizard

Welcome to the OSCAR wizard!

<table>
<thead>
<tr>
<th>Step 0:</th>
<th>Download Additional OSCAR Packages...</th>
<th>Help...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1:</td>
<td>Select OSCAR Packages To Install...</td>
<td>Help...</td>
</tr>
<tr>
<td>Step 2:</td>
<td>Configure Selected OSCAR Packages...</td>
<td>Help...</td>
</tr>
<tr>
<td>Step 3:</td>
<td>Install OSCAR Server Packages</td>
<td>Help...</td>
</tr>
<tr>
<td>Step 4:</td>
<td>Build OSCAR Client Image...</td>
<td>Help...</td>
</tr>
<tr>
<td>Step 5:</td>
<td>Define OSCAR Clients...</td>
<td>Help...</td>
</tr>
<tr>
<td>Step 6:</td>
<td>Setup Networking...</td>
<td>Help...</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Step 7:</th>
<th>Complete Cluster Setup</th>
<th>Help...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 8:</td>
<td>Test Cluster Setup</td>
<td>Help...</td>
</tr>
</tbody>
</table>

The following buttons are for managing your node definitions after the initial install.

<table>
<thead>
<tr>
<th>Help...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add OSCAR Clients...</td>
</tr>
<tr>
<td>Delete OSCAR Clients...</td>
</tr>
</tbody>
</table>
Step 0

Enables you to download additional packages

OPD – Oscar Package Downloader does download

OPDer – GUI frontend to OPD
OPDer

culumon and PVFS
selected for download
Step 1

Create your own flavor of cluster distribution

Select OSCAR packages to install.
Package Selector

Core packages are automatically selected for you and can not “unselect”

Download does not equal installation!

Packages downloaded with OPDer are selected for installation here
Step 2

Configure OSCAR packages that require special configuration tasks
Package configuration

Environment Switcher does configuration for default MPI use

make selection
Step 3

Install OSCAR Server (cluster head node) specific packages on cluster head node

May take a few minutes

Wait for button…
Install server packages

--- Updating /etc/profile
Back up /etc/profile
Adding path entries to /etc/profile
--- Updating /etc/exports
Back up /etc/exports
Checking for /home export
Adding /home export
--- Updating rsyncd.conf
Back up rsyncd header stub
Adding hosts allow to file
Updated rsyncd.conf file
--- Refreshing services

Starting NFS services:
Starting NFS quotas:
Starting NFS daemon:
Starting NFS mountd:
Starting sshd:
--- Fixing root "dot" files
Making any necessary PATH fixes to (/root/.bashrc)
Making any necessary PATH fixes to (/root/.tcshrc)
Making any necessary PATH fixes to (/root/.cshrc)
--- Finished server_prep script
--- Step 3: Successfully installed OSCAR server

success
Step 4

Specify and build system image for client (compute) nodes
**Build image configure**

- name your image
- list of packages
- package file location
- disk partition file location
- static or dynamic
- halt, reboot, beep
Building image

showing progress
Building image finished

success
Step 5

Define client nodes
Define client nodes

specify image name (from step 4 – or other saved image)

client IP domain name

client base name (oscarnodeXXX)

node count

starting index to append to base

padding to client names (3 = oscarnode009)

starting IP address

Subnet Mask

Default Gateway
Define client nodes

success
Step 6

in one operation – setup networking for all cluster client nodes

for first time in installation process we will “touch” the client nodes
Setup network – initial window

machines named as specified in prior step 5

IP address as specified in prior step 5
Setup network – scanning network

found first MAC address and assigned to machine
Setup network – initial window

found and assigned all MAC addresses
Reboot Clients

reboot on own – “post install action” from step 4

or

manually reboot
Step 7

only after ALL clients have rebooted

runs “post install” scripts for packages that have them

cleanup and reinitialize where needed
Complete setup

```
root@rh9vm:/opt/oscar

Shutting down ntpd: [FAILED]
ntpd: Synchronizing with time server: [ OK ]
Starting ntpd: [ OK ]

----- oscarnode2.oscardomain-----
Shutting down ntpd: [FAILED]
ntpd: Synchronizing with time server: [ OK ]
Starting ntpd: [ OK ]

--> About to run /opt/oscar/packages/loghost/scripts/post_install for loghost
*********** oscar_cluster *************

----- oscarnode1.oscardomain-----
Setting loghost to 192.168.152.227
Shutting down kernel logger: [ OK ]
Shutting down system logger: [ OK ]
Starting system logger: [ OK ]
Starting kernel logger: [ OK ]

----- oscarnode2.oscardomain-----
Setting loghost to 192.168.152.227
Shutting down kernel logger: [ OK ]
Shutting down system logger: [ OK ]
Starting system logger: [ OK ]
Starting kernel logger: [ OK ]
Cluster setup complete!

--> Step 7: Successfully completed the cluster install
```

success
Step 8

test suite provided to ensure that key cluster components are functioning properly
Test cluster setup

All Passed!!!
OSCAR
Cluster Maintenance
Add / Delete Nodes
Add OSCAR Clients

increase the number of compute nodes in the cluster
Add OSCAR Clients

Operates in similar manner to steps 5, 6, and 7 in OSCAR installation

Action behind the scenes differs though…
Delete OSCAR Clients

decrease the number of compute nodes in the cluster
Delete OSCAR Clients

ready to select client(s) to delete
Delete OSCAR Clients

client selected to delete
Delete OSCAR Clients

success

In order to delete OSCAR clients from your cluster, select the nodes you wish to delete and press the Delete Clients button.
Quit OSCAR Wizard

Your OSCAR cluster is now ready to use
Thin OSCAR

Sherbrooke University
Sherbrooke, Quebec, Canada

The Development Team

Benoit des Lignieris
Michel Barrette
Michel Dagenais
Francis Giraldeau
Thin OSCAR implementation

- Root RAM system
  - uses ram disks (/dev/ramXX)
  - compressed RAM disk image transferred by network at each boot
  - minimal system in ram (~20MB)
HA-OSCAR

The Development Team

Louisiana Tech University
Chokchai Leangsuksun
Lixin Sher
Hertong Song

Ericsson Research, Canada
Ibrahim Haddad

Oak Ridge National Laboratory
Stephen L. Scott
Conventional OSCAR Architecture
HA-OSCAR in active/hot-standby mode
HA-OSCAR in active/active mode
More OSCAR Information

Open Cluster Group
www.OpenClusterGroup.org/

OSCAR Home Page
oscar.sourceforge.net/

OSCAR Development site
sourceforge.net/projects/oscar/

Mailing Lists
oscar-users@lists.sourceforge.net
oscar-devel@lists.sourceforge.net
Questions