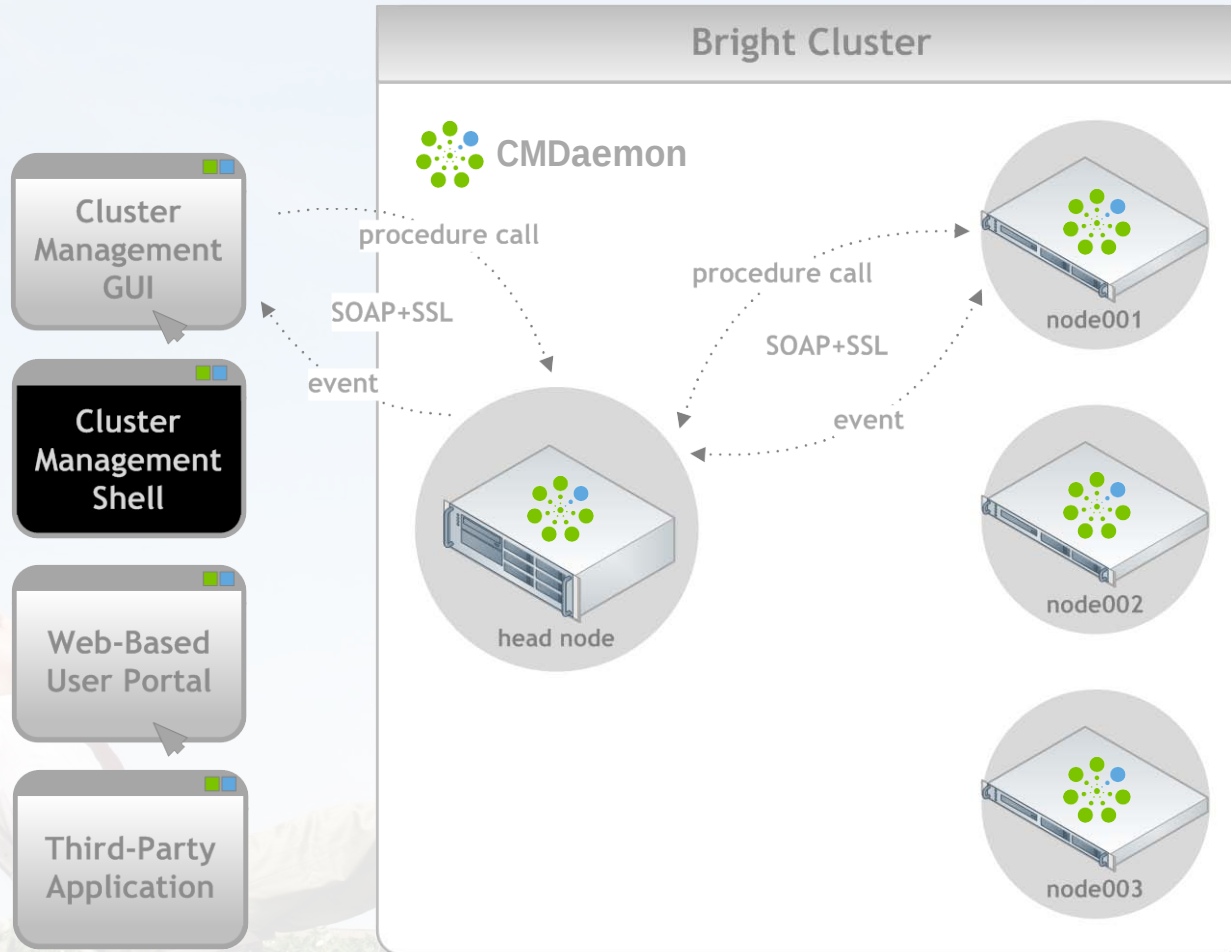


Cloud Bursting with SLURM and Bright Cluster Manager

Martijn de Vries
CTO



Architecture



Management Interfaces

Graphical User Interface (GUI)

- Offers administrator full cluster control
- Standalone desktop application
- Manages multiple clusters simultaneously
- Runs on Linux & Windows
- Built on top of Mozilla XUL engine



Cluster Management Shell (CMSH)

- All GUI functionality also available through Cluster Management Shell
- Interactive and scriptable in batch mode



RESOURCES

My Clusters

Demo Cluster

Switches

- switch01
- switch02
- switch03
- switch04
- switch05

Networks

- externalnet
- ipminet
- mpinet
- slavenet
- storagenet

Power Distribution Units

- apc01
- apc02
- apc03
- apc04

Software Images

- default-image

Node Categories

- slave

Head Nodes

- demohead1
- demohead2

Racks

Chassis

Virtual SMP Nodes

Slave Nodes

- node001
- node002
- node003
- node004
- node005
- node006
- node007
- node008
- node009

Demo Cluster

Overview

Settings

Failover

Rackview

Health

Parallel shell

License

Notes

Uptime: 45 days 3 hours 7 minutes

Nodes: 503 ↑ 7 ↓ 2 ⊖

GPU Units: 38 ↑ 0 ↓ 0 ⊖

Devices: 64 ↑ 0 ↓ 0 ⊖

Jobs: 45 running 67 waiting

Phase load: 783 A

CPU Cores:

3.93 K out of 4 K

GPUs:

13 out of 38

Memory:

7.32 TB out of 7.45 TB

Users:

13 out of 38

CPU Usage:

48% u 29% s 13% o 10% i

Occupation rate:

83.2 %

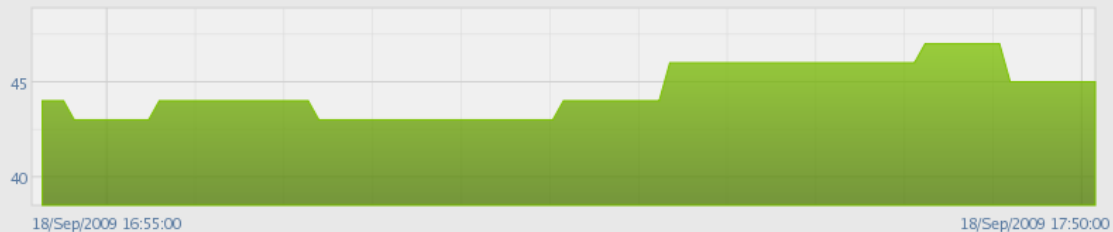
Disk Usage

Mountpoint	Used	Size	Use %
/	15.83 GB	37.25 GB	
/boot	14.31 MB	99.18 MB	
/home	832.6 GB	9.91 TB	

Workload Management

Queue	Running	Queued	Error	Completed	Avg. Duration	Est. delay
short.q	32	43	0	482	7 hours, 27 minutes	9 hours, 5 minutes
medium.q	5	11	0	41	2 days, 15 hours	4 days, 16 hours
long.q	8	13	0	91	8 days, 9 hours	15 days, 13 hours

Metric: RunningJobs[all.q]



EVENT VIEWER

All Events

▼	Ack	Time	▲	Cluster	▼	Source	▼	Message	▼
ⓘ		18/Sep/2009 17:05:53		Demo Cluster		demohead1		Service ntpd was restarted on demohead1	
ⓘ		18/Sep/2009 17:05:47		Demo Cluster		demohead1		Service named was restarted on demohead1	
ⓘ		18/Sep/2009 17:05:45		Demo Cluster		demohead1		Service postfix was restarted on demohead1	
ⓘ		18/Sep/2009 17:05:45		Demo Cluster		demohead1		Service dhcpcd was restarted on demohead1	
ⓘ		18/Sep/2009 17:05:45		Demo Cluster		demohead1		Service maui was restarted on demohead1	

Bright Cluster Manager

FileMonitoringViewHelp

RESOURCES

My Clusters

Seismic Houston

- Switches
 - switch01
 - switch02
 - switch03
 - switch04
 - switch05
- Networks
 - externalnet
 - ipminet
 - mpinet
 - slavenet
 - storagenet
- Power Distribution Units
 - apc01
 - apc02
 - apc03
 - apc04
- Software Images
 - default-image
- Node Categories
 - slave
- Head Nodes
 - demohead1
 - demohead2
- Racks
- Chassis
- Virtual SMP Nodes
- Slave Nodes
- Other Devices
- Node Groups
- Users & Groups
- Workload Management
- Monitoring Configuration
- Authorisation
- Authentication

Seismic Houston

OverviewSettingsFailoverRackviewHealthParallel shellLicenseNotes

U	Rack 1	Rack 2	Rack 3	Rack 4	Rack 5	Rack 6
01	demohead1	032	057	097098		231
02		033	058	099100		233234
03		034	059	101102		235236
04		035	060	103104		237238
05	demohead2	036	061	105106		239240
06		037	062	107108		241242
07		038	063	109110		243244
08		039	064	111112		245246
09		040	065	113114		247248
10		041		115116		
11	001	042	066	117118	169170	249250
12	002	043	067	119120	171172	251252
13	003	044	068	121122	173174	253254
14	004	045	069	123124	175176	255256
15	005	046	070	125126	177178	257258
16	006	047	071	127128	179180	259260
17	007	048	072	129130	181182	261262
18	008	049	073	131132	183184	263264
19	009		074		185186	265266
20	010		075		187188	267268
21	011		076		189190	269270
22	012		077		191192	271272
23	013		078		193194	273274
24	014		079		195196	275276
25	015	050	080	133134	197198	277278
26	016	051	081	135136	199200	279280
27	017	052	082	137138	201202	281282
28	018	053	083	139140	203204	283284
29	019	054	084	141142	205206	285286
30	020	055	085	143144	207208	287288
31		056		145146		

View:

Refresh

Setup

Temp CPU0 OC68.74 C

Temp CPU1 OC68.74 C

EVENT VIEWER

All Events

	Ack	Time	Cluster	Source	Message
i		18/Sep/2009 17:05:53	Demo Cluster	demohead1	Service ntpd was restarted on demohead1
i		18/Sep/2009 17:05:47	Demo Cluster	demohead1	Service named was restarted on demohead1
i		18/Sep/2009 17:05:45	Demo Cluster	demohead1	Service postfix was restarted on demohead1
i		18/Sep/2009 17:05:45	Demo Cluster	demohead1	Service dhcpd was restarted on demohead1
i		18/Sep/2009 17:05:45	Demo Cluster	demohead1	Service maui was restarted on demohead1

Ready

Workload Manager Integration

Integration with workload manager:

- All popular workload managers supported
- SLURM default choice during installation
- Automatic installation

Points of integration:

- Automatic node and queue configuration
- Automatic high availability configuration
- Monitoring workload management metrics
- Health checking
- Job monitoring and control

Workload Management

English(US)



- Welcome
- License
- Kernel Modules
- Hardware Info
- Nodes
- Network Topology
- Additional Networks
- Networks
- Nameservers
- Network Interfaces
- Subnet Managers
- Installation Source
- **Workload Management**
- Disk Layout
- Time Configuration
- Cluster Access
- Authentication
- Console
- Summary

A workload management system is highly recommended to run compute jobs. Please choose the workload management system that should be configured. To prevent a workload management system from being set up, select 'None'. The number of slots per node should ideally be equal to the number of CPU cores available on each node. On small clusters, the head node may also be used for compute jobs.

Workload management system **Slurm(v2.2.4)**

Number of slots/node

Use head node for compute jobs ☐ Yes ☒ No



The Simple Linux Utility for Resource Management (SLURM) is an open source, fault-tolerant, and highly scalable cluster management and job scheduling system for large and small Linux clusters. The slurm controller daemon will be configured to run on the head node and the slurm daemons will be configured to run on all the nodes. If the master node is required to run jobs, then the slurmd will also run on the head node. MySQL will be used to store job accounting information.

[Cancel](#)[Go Back](#)[Continue](#)

Cloud Bursting

Scenario I



Mixing Local and Cloud Resources

Cloud does not work well for all HPC workloads

- Sensitive data/computations
- Problems getting huge amounts of data in/out
- Workload may depend on low latency / high bandwidth
- Workload may depend on non-standard compute resources
- Workload may depend on advanced shared storage (e.g. Lustre)

Not everyone will replace HPC cluster with EC2 account

- Allow local cluster to be extended with cloud resources to give best of both worlds
- Allow workload suitable for cloud to be off-loaded
- Allow traditional HPC users to try out and migrate to cloud

Cloud Bursting

Scenario II



RESOURCES

My Clusters

ScaleMP Demo Cluster

- Switches
- Networks
- Power Distribution Units
- Software Images
- Node Categories
- Head Nodes
- Racks
- Chassis
- Virtual SMP Nodes
- Nodes
- Cloud Nodes**
 - Amazon EC2 - Demo
 - Amazon EC2 - Production
- GPU Units
- Other Devices
- Node Groups
- Users & Groups
- Workload Management
- Monitoring Configuration
- Authorization
- Authentication



Cloud Nodes



ScaleMP Demo Cluster

Overview

Tasks

Cloud Accounts

Amazon EC2 - Production



Provider: Amazon EC2
Username: production@brightcomputing.com
Certificate: /root/.ec2/production.cert

Defined instances: 0
Active instances: 0
Total charges: \$0



Amazon EC2 - Demo



Provider: Amazon EC2
Username: demo@brightcomputing.com
Certificate: /root/.ec2/demo.cert

Defined instances: 8
Active instances: 5
Total charges: \$139



Add a new cloud account

Bright Cluster Manager

File

Monitoring

View

Tools

Help

RESOURCES

My Clusters

ScaleMP Demo Cluster

Switches

Networks

Power Distribution Units

Software Images

Node Categories

Head Nodes

Racks

Chassis

Virtual SMP Nodes

Nodes

Cloud Nodes

Amazon EC2 - Demo

Amazon EC2 - Production

GPU Units

Other Devices

Node Groups

Users & Groups

Workload Management

Monitoring Configuration

Authorization

Authentication

Amazon EC2 - Production

ScaleMP Demo Cluster

Overview

Storage

Status

Tasks

Network Setup

Modified

Hostname

Type

Category

Persistent storage

Node Creation Wizard

First node:

node001

Last node:

node016

Category:

default

Type:

m1.xlarge

Storage:

EBS

40G

Cancel

Previous

Next

Open

Add

Clone

Remove

Create Nodes

Revert

Save

virtual node network overview refreshed, virtual nodes status refreshed

RESOURCES

My Clusters

ScaleMP Demo Cluster

- Switches
- Networks
- Power Distribution Units
- Software Images
- Node Categories
- Head Nodes
- Racks
- Chassis
- Virtual SMP Nodes
- Nodes
- Cloud Nodes

Amazon EC2 - Demo

Amazon EC2 - Production

- vnode001
- vnode002
- vnode003
- vnode004
- vnode005
- vnode006
- vnode007
- vnode008
- vnode009
- vnode010
- vnode011
- vnode012
- vnode013
- vnode014
- vnode015
- vnode016

- GPU Units
- Other Devices
- Node Groups

Users & Groups

Workload Management



Amazon EC2 - Production



ScaleMP Demo Cluster

Overview

Storage

Status

Tasks

Network Setup

Modified	Hostname	Type	Category	Persistent storage
	vnode001	m1.xlarge	default	EBS (40G)
	vnode002	m1.xlarge	default	EBS (40G)
	vnode003	m1.xlarge	default	EBS (40G)
	vnode004	m1.xlarge	default	EBS (40G)
	vnode005	m1.xlarge	default	EBS (40G)
	vnode006	m1.xlarge	default	EBS (40G)
	vnode007	m1.xlarge	default	EBS (40G)
	vnode008	m1.xlarge	default	EBS (40G)
	vnode009	m1.xlarge	default	EBS (40G)
	vnode010	m1.xlarge	default	EBS (40G)
	vnode011	m1.xlarge	default	EBS (40G)
	vnode012	m1.xlarge	default	EBS (40G)
	vnode013	m1.xlarge	default	EBS (40G)
	vnode014	m1.xlarge	default	EBS (40G)
	vnode015	m1.xlarge	default	EBS (40G)
	vnode016	m1.xlarge	default	EBS (40G)

Open

Add

Clone

Remove

Create Nodes

Revert

Save

RESOURCES

My Clusters

ScaleMP Demo Cluster

- Switches
- Networks
- Power Distribution Units
- Software Images
- Node Categories
- Head Nodes
- Racks
- Chassis
- Virtual SMP Nodes
- Nodes

Cloud Nodes

- Amazon EC2 - Demo
- Amazon EC2 - Production

vnode001

vnode002

vnode003

vnode004

vnode005

vnode006

vnode007

vnode008

vnode009

vnode010

vnode011

vnode012

vnode013

vnode014

vnode015

vnode016

GPU Units

Other Devices

Node Groups

Users & Groups

Workload Management

vnode001

ScaleMP Demo Cluster

Overview

Tasks

Settings

Cloud Settings

System Information

Services

Process Management

Network Setup

Type: m1.xlarge

15 GB memory
8 EC2 Compute Units (4 virtual cores with 2 EC2 Compute Units each)
1,690 GB instance storage
64-bit platform
I/O Performance: High
API name: m1.xlarge

Storage: vol-3678316 (EBS) 40GB /dev/sda ☒ rootfs -Storage: ephemeral0 auto /dev/sdb ☐ rootfs - +

Region: default (us-east-1)

Sub region: auto

Expert options:

Instance ID:

SSH connect string:

Revert

Save

RESOURCES

My Clusters

ScaleMP Demo Cluster

- Switches
- Networks
- Power Distribution Units
- Software Images
- Node Categories
- Head Nodes
- Racks
- Chassis
- Virtual SMP Nodes
- Nodes
- Cloud Nodes

Amazon EC2 - Demo

Amazon EC2 - Production

- vnode001
- vnode002
- vnode003
- vnode004
- vnode005
- vnode006
- vnode007
- vnode008
- vnode009
- vnode010
- vnode011
- vnode012
- vnode013
- vnode014
- vnode015
- vnode016

- GPU Units
- Other Devices
- Node Groups

Users & Groups

Workload Management



Production



ScaleMP Demo Cluster

Overview

Storage

Status

Tasks

Network Setup

Power:

On

Off

Reset

Operating System:

Shutdown

Restart

Software image:



Update nodes

Reinstall nodes

Workload:

Drain

Undrain

Set category:

Set

Misc:

Open

Close

Provisioning Log

RESOURCES

My Clusters

ScaleMP Demo Cluster

- Switches
- Networks
- Power Distribution Units
- Software Images
- Node Categories
- Head Nodes
- Racks
- Chassis
- Virtual SMP Nodes
- Nodes

Cloud Nodes

Amazon EC2 - Demo

Amazon EC2 - Production

- vnode001
- vnode002
- vnode003
- vnode004
- vnode005
- vnode006
- vnode007
- vnode008
- vnode009
- vnode010
- vnode011
- vnode012
- vnode013
- vnode014
- vnode015
- vnode016

- GPU Units
- Other Devices
- Node Groups
- Users & Groups

Workload Management



Workload Management



ScaleMP Demo Cluster

Jobs

Queues

Nodes

Modified	Job ID	Scheduler	User	Queue	Status
	modflow	slurm	kate	long.cloud	running
	modflow	slurm	kate	long.cloud	running
	modflow	slurm	kate	short.mixed	running
	modflow	slurm	kate	short.mixed	running
	modflow	slurm	kate	short.mixed	running
	myparallelapp	slurm	jodi	medium.local	running
	serial1	slurm	alice	serial.cloud	running
	serial1	slurm	alice	serial.cloud	running
	serial1	slurm	alice	serial.mixed	running
	serial1	slurm	alice	serial.mixed	running
	xhpl	slurm	matthew	short.local	running
	xhpl	slurm	matthew	short.local	running
	xhpl	slurm	matthew	short.local	running
	modflow	slurm	kate	long.cloud	queued
	modflow	slurm	kate	short.mixed	queued
	serial1	slurm	alice	serial.mixed	queued

Show

Remove

Hold

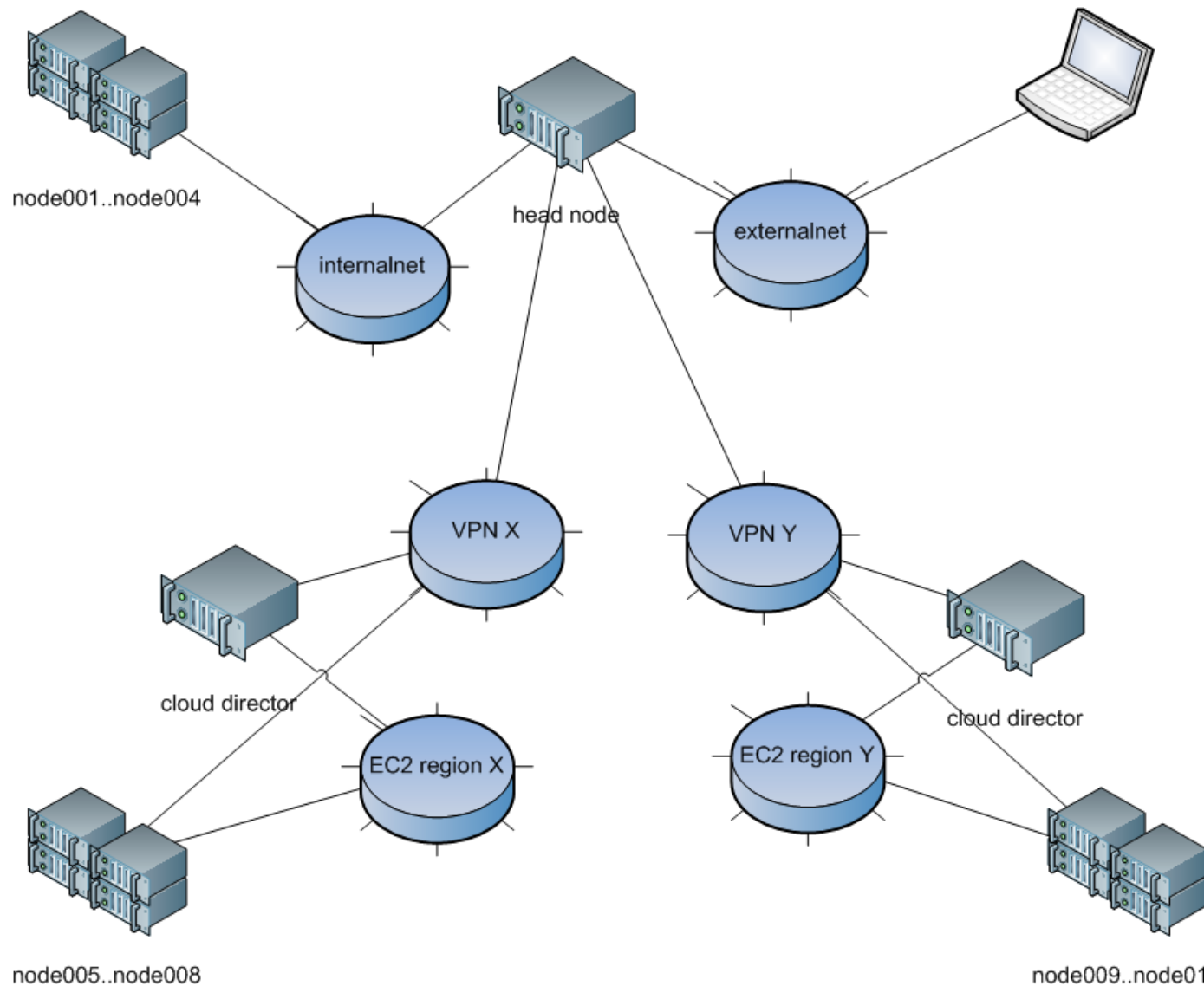
Release

Suspend

Resume

Refresh

Cloud Network Map



Uniformity

Cloud nodes behave the same way as local nodes

- Same method of provisioning
- Same software image and user environment
- Same workload management set-up
- Same management interface that allows to control cluster
- Same monitoring & health checking

Everything can talk to everything

- Accomplished using VPN, routing, network mapping
- VPN set-up automated and does not require firewall set-up (requires just **outgoing** access on 1194/udp)
- Single global DNS namespace

Running Cloud Nodes

Cloud Director has a number of responsibilities:

- Gateway between local and cloud nodes
- Provision software image to cloud nodes
- Serve shared storage for cloud nodes
- Mirror network services for the cloud nodes (e.g. LDAP, DNS)

Cloud node booting process

- Instances are created with 1GB EBS and n GB ephemeral/EBS disk
- Bright Node Installer AMI goes on EBS disk
- Node Installer continues with normal procedure to bring up node
- Software image gets provisioned onto second disk

SLURM & Bright Cloud Bursting

- Common setup: one SLURM partition per cloud region
- Example:

```
[root@sc11-demo ~]# sinfo
```

PARTITION	AVAIL	TIMELIMIT	NODES	STATE	NODELIST
defq*	up	infinite	1	idle	node001
california	up	infinite	4	idle	cnode[001-004]
oregon	up	infinite	4	idle	cnode[005-008]
- Jobs that may run in the cloud should be submitted to one of the cloud partitions
- SLURM will schedule jobs onto cloud nodes the same way as on local nodes
- Current situation:
 - /cm/shared mirrored and exported by cloud director
 - /home mounted over VPN
- Works great, but /home is too slow

Data Locality Problem

- Jobs usually require input data and produce output data
- Input and/or output data may require significant transfer time
- Resources charged by the hour, so input/output data should be transferred while resources are not yet allocated
- Solution to data locality problem should ideally be hidden from users as much as possible

Data Aware Workload Management

- SLURM needs to be made aware of job data dependencies
- Jobs should not be scheduled until data is present on cloud-director
- As part of job script, copy input data in special input directory, copy output directory into output directory
- Workload management environment takes care of transferring input and output directories
- Option A) let SLURM take care of copying data (e.g. using job dependencies)
- Option B) transfer data using separate daemon and set SLURM job attributes to allow/disallow job start

Questions?

