The Design and Implementation of a Virtual Cluster Management System

Hidemoto Nakada[1], Takeshi Yokoi[1], Tadashi Ebara[1,2], Yusuke Tanimura[1], Hirotaka Ogawa[1], and Satoshi Sekiguchi[1]

[1] National Institute of Advanced Industrial Science and Technology (AIST), Tukuba, Japan,
[2] SURIGIKEN Co., Ltd Tokyo, Japan

Abstract. To fully utilize resources in computer center, virtualization techniques are getting popular and several systems are proposed for this purpose. However, they just provide set of virtualized nodes, not the 'virtual clusters'; i.e., they are not able to install and configure middlewares and tools that makes 'set of nodes' into 'cluster'. Another problem is that they just virtualize nodes, leaving storage resources and networks, which are equivalently essential for clusters, un-virtualized. We propose a virtual cluster management system which virtualizes compute resources, as well as disk storage and network, and install and setup softwares that are essential for cluster operation, using Rocks, a cluster provisioning system. We virtualize storage with iSCSI and network with tagged VLAN.

1 Introduction

For organizations that have to manage large number of computer resources, such as computer centers or data centers, to gain utilization and to reduce management cost are the key issues. One possible way to address these issues is *cluster virtualization* technique, which abstracts resources in the organization, constructs "virtual clusters", and provides them to the users. The virtualized resources can be mapped on to the real resource arbitrary, allowing the resource administrators to change the mapping dynamically for efficient usage of the real resources.

One important issue on virtual cluster management is the fact that a 'virtual cluster' is much more than a 'set of virtualized nodes'. To be a real cluster, a lot of things are required, such as username space management, shared file systems, monitoring software, and batch queuing system. Other important issues are storage and network virtualization. Usually, storage is closely tied to the physical entity that resides. Without storage virtualization, a cluster cannot be fully virtualized. Network virtualization is required from the security aspect. It is essential to isolate several virtual clusters on a real cluster.

We propose a virtual cluster management system that addresses the issues shown above. It provides fully installed and configured virtual cluster, using NPACI Rocks. It virtualizes storages and networks with iSCSI and VLAN for complete virtualization and separation. We implement a prototype of the system and measured time spent for installing a virtual clusters.

2 NPACI Rocks

In this section, we describe the overview of the NPACI Rocks [1, 2] cluster installation system, which was used in our system. Rocks is developed in SDSC (San Diego Supercomputer Center) as a part of NPACI (National Partnership for Advanced Computational Infrastructure) effort. Rocks is a cluster management tool to reduce the management cost of clusters. It installs specified software packages on the computation nodes of clusters automatically. It uses CentOS (Community ENTerprise Operating System) which is based on Redhat Enterprise Linux.

Rocks assumes a cluster to have one "frontend node" and several "computation nodes" connected with private network. of which address space is managed by the frontend. The frontend has another network interface connected to public network, and acts as a NAT router for compute nodes. Rocks installs and sets up softwares for cluster operation, such as 411 for configuration file management and Ganglia [3] for cluster monitoring, as well as ordinary software packages for each compute nodes.

Rolls and Appliances Rocks manages softwares by "meta-packages" called *Rolls*. Each Roll stores RPM packages, dependency description written in XML, and post-install deployment processes. A cluster manager can add new capabilities to a cluster just specifying Rolls on the installation. A number of Rolls, mainly on high-performance computing and scientific computing, are made available by Rocks team, software developers and vendors

Rocks can manage several configurations for compute nodes. The configurations are called *Appliance types*. For example, a cluster can have Web servers and database nodes in it. To enable this, cluster manager defines "Web server appliance" and "database node appliance" and allocate nodes for each appliance type.

Appliance types and Rolls are orthogonal. An appliance type might be defined by several Rolls, and a Roll might define several appliances.

Cluster installation with Rocks Steps taken for cluster installation with Rocks are as follows. First, the cluster manager installs a frontend from CD-ROM specifying Rolls to be used for the cluster. Then, he / she start up each computation node with network boot, specifying appliance type. The frontend node acts as an installation server and provides packages for compute nodes. The node number will be automatically allocated in the order of starting up.

3 The Design of the System

3.1 Scenario for virtual cluster usage

The players in this system are cluster providers, service providers, and users. The cluster providers manage their real clusters using this system and provide virtual clusters to the service providers. The service providers 'rent' virtual clusters from the cluster providers and provide services to the users.

The virtual clusters consist of one frontend node and several worker nodes. The frontend node has public network interface as well as private one to talk with workers, and works as the gateway for worker nodes.

Here, we show the scenario of the usage of the system.

- The cluster providers installs real cluster with the system.
- A service provider requests for a virtual cluster, specifying time slot, number of nodes, required memory size, required softwares for it. The software might be supplied by the service provider.
- The cluster provider constructs a virtual cluster on the real cluster following the specification and provides access to it for the service provider.
- The service provider provides service to the user using the software and the virtual cluster.

One possible usage of a virtual cluster is 'computation farm'. Assume that a company has huge computation to be done within a month. The computation can be some scientific one or computer graphics rendering. The company contracts with a cluster provider to be provided one virtual cluster with 100 nodes, job scheduling system and proper application programs installed. On the reservation start time, the virtual cluster is setup and accepts connection from the users in the company.

Another possible usage will be 'cluster for classes'. Using this system, computer centers in universities or colleges can easily setup an isolated virtual cluster for each class.

3.2 Requirements for virtual clusters

Here are the requirements for virtual clusters.

Nodes configuration The virtual clusters have to be indistinguishable with the real one for the service providers and the users. It means the virtual clusters have to have the same node configuration with the typical real clusters; with one frontend nodes with public network access and several worker nodes with private network access only.

Cluster Operation Software Installation To make a set of node a “cluster”, cluster operation software have to be set up, such as configuration management software NIS, or cluster monitoring software Ganglia [3], and job scheduling system like Grid Engine, TORQUE, or Condor.
The softwares have to be setup head node and worker nodes in consistent way. For example, for the Grid Engine, master node has to be configured with worker node list, and the worker nodes have to be configured with the master node name.

Computer Virtualization For cluster virtualization we have to virtualize the most significant resource in the cluster, computers.

Storage Virtualization Storage is also an important resource for clusters. There are two kinds of storages in clusters; local storage for each node and shared storage that can be accessed from all the nodes. The size of storage should be able to be setup upon requests from service providers. Management cost for storage also have to be concerned.

Typically, virtual machine file systems are allocated on the host computer file systems. This method is easy to implement, but the file system size cannot exceed the host computer file system size. And the host computer file system cannot be shared by other virtual machine on other node. This configuration will increase the storage management cost, since the storage spans all the virtual computer hosting nodes. Centralized storage will be better for management.

Network Virtualization Since we are assuming that several virtual clusters share one real cluster, network isolation has to be considered to ensure secure communication within each cluster. Usual bridged network, which might be used for naive implementation, will cause all the virtual cluster shares single address space. We have to provide network isolation among virtual clusters, provided the cluster managers tend to assume intra cluster network is secure.

3.3 Overview of the design

Software installation and configuration We employed NPACI Rocks for software installation and configuration for virtual clusters. This is because of the Rocks' wide installation base and a variety of software meta-packages (Rolls).

Computer Virtualization We used VMWare Server [4] for computer virtualization. The VMWare Server is a product supplied by the VMWare inc. and can be used for free, under the license agreement. The VMWare Server provides full virtualization including BIOS and devices. Almost all the operating system can run on it without modification.

Storage Virtualization As a storage virtualization method, SAN (Storage Area Network) is proposed. SAN typically uses high speed network technologies such as Fiber Channel or Infiniband, which are generally too expensive to be widely deployed.

We employed *iSCSI* [5]; one of the implementation of IP based SAN, which can be implemented inexpensively. In iSCSI, the SCSI protocol packets encapsulated in IP packets are transferred over the IP network. It does not require any special hardware for implementation. In iSCSI, the computer that provides storage is called *target*, and the one that uses it is called *initiator*. To use iSCSI storage as the virtual machine file system, the virtual machine monitor has to have the initiator capability in it. Unfortunately, the VMWare Server does not provide it. [3] We worked around this as follows. We installed initiator capability on the host computer, instead of in the virtual machine monitor. When the

[3] VMWare Server ESX does support this capability.

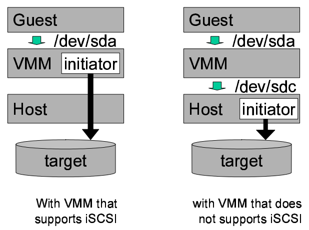


Fig. 1. iSCSI Configurations.

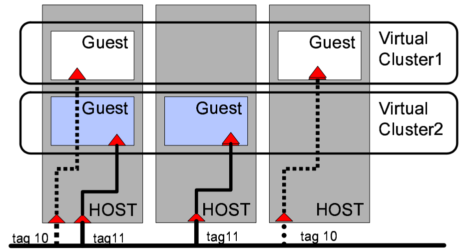


Fig. 2. Network isolation with VLAN.

host computer attaches to a target, the volume appears as a storage device (like /dev/sdc) on the host. We used the device as the file system for the VMWare Server guest. From the VMWare Server point of view, the device is just a physical disk, while it is iSCSI disk on the remote target machine in reality. (fig. 1)

Network Virtualization For network virtualization, there are two methods available; VLAN and VPN. The latter is originally meant for constructing a virtual LAN spanning on physically distributed offices. It encapsulates network packets in wrapping packets and route them to the destination. Although VPN can provide highly secure environment, the overhead imposed by packet encapsulation is substantial.

We employed tag based VLAN for network virtualization. The system allocates unique VLAN ID (tag) for each virtual cluster. It dynamically creates a network interface with the VLAN ID on the host and bridge the connection form the guest to the interface. Fig. 2 shows two virtual clusters (1 and 2) configured on three real nodes. The virtual cluster 1 is allocated VLAN ID 10 and 2 is allocated 11. The left computer has two interfaces with both ID, since it has to host two virtual machines for both virtual clusters. Note that there is no configuration is required on the guest, since the tagging is performed on the host.

4 Implementation

4.1 Overview

We use the Rocks for the virtual cluster provisioning. For that, we have to install 'virtual frontend' on the virtual cluster, and then install 'virtual compute nodes' from it. Note that the real cluster itself is also managed by Rocks, reducing the management cost of the real cluster.

The system consists of 4 types physical nodes (fig. 3A).

- **Cluster Manager Node**
  The cluster manager module, that manages all the physical nodes and virtual clusters, sits on this node. It has network interfaces for public and private

networks. It provides Web interface and Web Service interface for requesting and monitoring the virtual cluster reservation. There is only one node of this kind on the whole physical cluster.

This node also acts as the Rocks frontend for physical cluster installation. The following three type's nodes are installed via this node using Rocks.

- **Gateway Node**
  The virtual frontend nodes will be hosted on this kind of nodes. These nodes also have two network interfaces for private and public network.
- **Physical Compute Node**
  The virtual compute nodes will be hosted on this kind of nodes. These nodes just have one interface connected to the private network. The difference between thess and the Gateway nodes are just the number of network interface.
- **Storage Node**
  This kind of nodes have large disk space and provide them via iSCSI to other nodes. They have private network access only.

  They use LVM (Logical Volume Manager) for storage management. When the cluster manager requests a volume for a Storage Node, it creates a new logical volume and publishes it as an iSCSI volume. LVM hides the size or boundary of the physical disk and enables efficient management of the disk space.

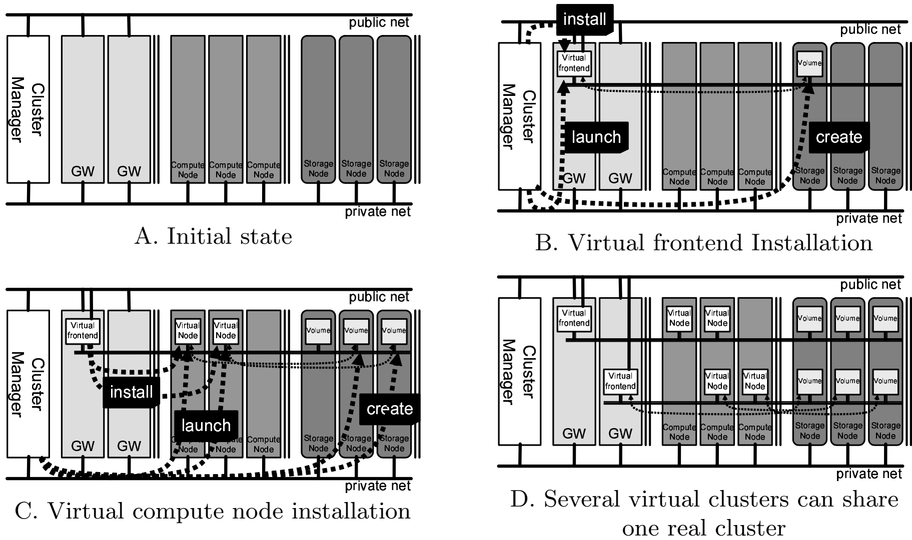
4.2 The Cluster Manager Implementation

The Cluster Manager consists of several python scripts that shares one database on the cluster manager node. Reservation requests from service providers will be processed by set of CGI scripts that just update the database. The scripts which make the reservations happen will be invoked from the *cron* daemon. The scripts look up the database table and find what should be done now and do it. This 'database centric design' makes the cluster manager robust, eliminating daemons that might crushes during execution.

4.3 Virtual Cluster Installation Steps

Virtual cluster installation has following steps: 1) creation of a VLAN on the private network, 2) installation of virtual frontend on the gateway node, 3) installation of virtual compute nodes from the virtual frontend.

VLAN configuration The central manager allocates the real compute nodes and storage nodes to be used for the virtual cluster, and then allocate VLAN ID for it. Note that VLAN ID space is limited, so it should be treated as reusable resources. The central manager orders the real compute nodes and storage nodes to create a tagged interface with the allocated VLAN ID.



A. Initial state

B. Virtual frontend Installation

C. Virtual compute node installation

D. Several virtual clusters can share one real cluster

Fig. 3. Virtual Cluster Installation

Viratual Frontend Launch and Installation A virtual frontend will be installed on a gateway node. For automatic frontend installation, special CDROM images are required, that includes all the information required on installation, such as IP addresses, cluster name, list of Rolls and the places to download the Rolls.

The cluster manager creates a special CDROM image on the gateway node that is responsible for hosting the virtual frontend. Then it starts up VMWare guest, that will become the virtual frontend, specifying the CDROM image in the configuration file. The virtual frontend boots up from the CDROM image and installs itself as specified in the image. For the file system for the virtual frontend, an iSCSI volume on a storage node will be used (fig. 3B). We will describe this on the next item in detail. Rolls to be installed on the virtual frontend have to be provided by the cluster manager node. The virtual frontend downloads the rolls and installs them. using the public network.

The virtual frontend have to be configured with the user (service provider) specific information, such as ssh public keys, users, hostnames, and cluster allocation settings, such as virtual node MAC address and IP addresses, appliance types for nodes, These informations are stored in a special roll that will be installed on the virtual frontend, On the installation, the roll updates Rocks database and overwrites some files on the cluster so that the user specified information takes effect.

Virtual Compute Node Launch and Installation Virtual compute nodes are installed on the real compute nodes (fig. 3C). A real node launches a virtual node as a guest, specifying the VLAN network as the bridging interface. The

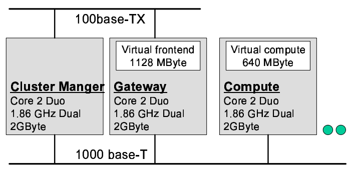


Fig. 4. Environment for measurement.

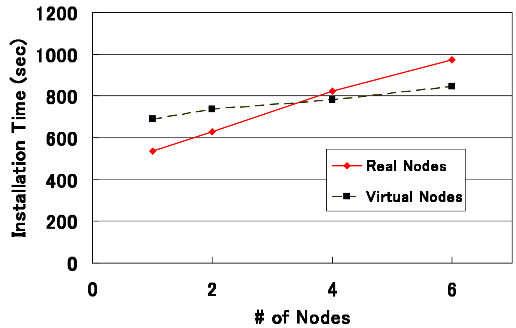


Fig. 5. Time spent to install nodes.

virtual node automatically boots up and is installed following Rocks installation process, downloading everything from the virtual frontend node.

The virtual compute node attaches iSCSI volume as the file system, as described in 3.3. In advance of the virtual node startup, the cluster manager requests allocation of an iSCSI volume and obtains an ID for the volume. Then the cluster manager requests to the real compute nodes to start up the virtual node specifying the ID. The real compute nodes attaches the volume and setup the device name for the volume in the virtual machine configuration file.

Virtual Clusters share One Real Cluster Fig. 3D shows two virtual clusters sharing one real cluster. Note that one compute node is shared by two clusters. While two virtual clusters in the figure have its own gateway nodes, it is not mandatory. Several virtual clusters can share one gateway node as far as the node can provide enough memory and bandwidth.

4.4 Measurements on Virtual Cluster Installation

We measured required time to install a virtual cluster. Experimental setup is shown in fig. 4.

The time spent to setup a virtual frontend is shown in tab. 1. The 'Rocks distribution build' shown in the table shows the time spent for rebuilding the distribution package files in the cluster manager, so that it can be used for virtual frontend installation. The most time spent operation is the installation itself, taking more than 27 min. One of the reasons is that it used 100 base-TX network, not the 1000 base-T network, for downloading all the packages from the cluster manager. We might be able to speed up this phase replacing the network with 1000 base-T.

Fig. 5 shows time spent for installing virtual compute nodes from the virtual frontend. For reference we also showed the real compute node installation time on the same figure. Note that the file systems for the virtual compute nodes are on the host file systems, not on the iSCSI storage for this experiment. We can

Table 1. Time spent for installing virtual frontend (sec.)

Operation	time
ISO image modification	42
Restore roll creation	2
Rocks distribution build	155
Installation	1623
Total	1822

see the installation time for the virtual compute nodes are almost same as the time spent for the real one.

5 Related Work

Virtual Workspace [6, 7] is one of the sub projects of the Globus project [8], which is aiming to provide virtualized execution environment for job execution. It defines interface for managing virtual execution environment and execution in it based on WSRF. This project is focusing on the individual job execution, while we are targeting on providing complete virtual clusters that have complete configuration and longer life time.

In [9], the authors showed a virtual cluster system with Xen and OSCAR [10], which is another cluster provisioning system similar to Rocks. They just showed that a virtual cluster can be installed with OSCAR. Fully automatic installation of the virtual cluster and storage and network virtualizations are not addressed.

VFrame from Cisco System inc. [11] enables IP network and storage virtualization with Infiniband based SAN technology. While this technology is highly sophisticated, for example it can provide I/O QoS, it is based on proprietary hardware and too expensive to be installed deployed in a small computer center such as computer center in universities.

6 Conclusion

We proposed a virtual cluster management system that can dynamically construct complete clusters with head nodes and cluster operation software installed. The system virtualizes network and storage, as well as computers, reducing management cost of the virtual clusters. The system is still in early development stage and having several issues to be addressed, including:

- **Hiding installation cost from service providers**
  The virtual cluster installation takes about 45min. in total. We have to hide this cost from the service providers using some tricks such as background installation.

- **Working with Xen**
  The proposed system is using VMWare Server for computer virtualization, and cannot work with Xen [12], one of the most widely used virtual machine monitor. This is because the anaconda installer for CentOS 4, which was used for Rocks, is not aware of Xen. In the spring of 2007, CentOS 5, that presumably aware of Xen, will be released. We will address this issue after the release.
- **Advanced virtual storage management**
  Currently, we provide an iSCSI volume mounted on the virtual frontend as a shared storage for the virtual compute nodes via NFS. In this configuration, all the access to the shared disk will go to one iSCSI target, restricting the total bandwidth for the shared storage. Moreover, all the access will go through the virtual frontend making the node the bottleneck. We are planning to address the issue using cluster file systems such as GFS, PVFS2, Lustre, or Gfarm [13] to evenly distribute storage access on several iSCSI targets. The cluster file system will also useful to provide high speed temporal storage, that is required by specific kind of scientific computation, by striping the file access to several targets.

References

1. Philip M. Papadopoulos, Mason J. Katz, and Greg Bruno. Npaci rocks: Tools and techniques for easily deploying manageable linux clusters. In *Cluster 2001: IEEE International Conference on Cluster Computing*, 2001.
2. Rocks. http://www.rocksclusters.org/.
3. Ganglia. http://ganglia.sourceforge.net/.
4. Vmware. http://www.vmware.com.
5. iSCSI Specification. http://www.ietf.org/rfc/rfc3720.txt.
6. Virtual workspace. http://workspace.globus.org/.
7. Kate Keahey, Ian Foster, Tim Freeman, and Xuehai Zhang. Virtual workspaces: Achieving quality of service and quality of life in the grid. *Scientific Programming Journal*, 2006.
8. Globus project. http://www.globus.org.
9. Geoffroy Vallée and Stephen Scott. Xen-oscar for cluster virtualization. In *Workshop on XEN in HPC Cluster and Grid Computing Environments (XHPC '06)*, 2006.
10. OSCAR: open source cluster application resources. http://oscar.openclustergroup.org/.
11. Cisco VFrame Server Fabric Virtualization Software. http://cisco.com/en/US/products/ps6429/products_data_sheet0900aecd8029fc58.html.
12. Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauery, Ian Pratt, and Andrew Warfield. Xen and the art of virtualization. In *SOSP 2003*, 2003.
13. Osamu Tatebe, Youhei Morita, Satoshi Matsuoka, Noriyuki Soda, and Satoshi Sekiguchi. Grid datafarm architecture for petascale data intensive computing. In *Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2002)*, pages 102–110, 2002.