Methods For Creating XSEDE Compatible Clusters

Jeremy Fischer
Indiana University
2709 E. Tenth Street
Bloomington, IN 47408
jeremy@iu.edu

Richard Knepper
Indiana University
2709 E. Tenth Street
Bloomington, IN 47408
rknepper@iu.edu

Matthew Standish
Indiana University
2709 E. Tenth Street
Bloomington, IN 47408
mstandis@iu.edu

Craig A. Stewart
Indiana University
2709 E. Tenth Street
Bloomington, IN 47408
stewart@iu.edu

Resa Alvord
Cornell Center for Advanced Computing
Frank H.T. Rhodes Hall
Hoy Road
Ithaca, NY 14853-3801
rda1@cornell.edu

David Lifka
Cornell Center for Advanced Computing
Frank H.T. Rhodes Hall
Hoy Road
Ithaca, NY 14853-3801
lifka@cac.cornell.edu

Barbara Hallock
Indiana University
2709 E. Tenth Street
Bloomington, IN 47408
bahalloc@iu.edu

Victor Hazlewood
National Institute for Computational Sciences
University of Tennessee
Oak Ridge National Laboratory
PO Box 2008, BLDG 5100
Oak Ridge, TN 37831-6173
vhazlewo@utk.edu

ABSTRACT

The Extreme Science and Engineering Discovery Environment has created a suite of software that is collectively known as the basic XSEDE-compatible cluster build. It has been distributed as a Rocks roll for some time. It is now available as individual RPM packages, so that it can be downloaded and installed in portions as appropriate on existing and working clusters. In this paper, we explain the concept of the XSEDE-compatible cluster and explain how to install individual components as RPMs through use of Puppet and the XSEDE compatible cluster YUM repository.

Categories and Subject Descriptors

C.4 [**Computer Systems Operations**]: Performance of Systems - Reliability, availability, and serviceability;

C.4 [**Computer Systems Operations**]: C.5.5.5 Servers

General Terms

Management, Documentation, Performance, Design, Standardization, Compatibility

Keywords

XSEDE, Rocks, rolls, cluster, Linux, RedHat, CentOS, HPC, puppet, rpm, yum, campus, bridging, research,

1. INTRODUCTION

In response to needs expressed by the US research community, the Extreme Science and Engineering Discovery Environment (XSEDE) Campus Bridging group has created the concept of a "basic XSEDE-compatible cluster." The XSEDE Compatible Basic Cluster (XCBC) is a computational cluster build that uses open source tools exclusively to create a cluster that operates from the user's perspective in a way the same as or analogous to a cluster in XSEDE. This cluster build won't magically make a GPU appear in your local cluster just because there are lots of GPUs in XSEDE clusters, but when a person wants to compile a C program with the gcc compiler, the ScaLAPAK mathematical library, or other common open source tool, that tool will be in the same place on the basic XSEDE-compatible cluster that compiler will be in the same place and work the same way as in a cluster created with the XCBC as on XSEDE.

The XCBC build has been available for some time as a "Rocks Roll" [1][2][3]. The current contents of the XCBC are described in detail in the Knowledge Base document "What software is installed on a "bare-bones" XSEDE-compatible Rocks cluster?" [4] The packages included in this build are specific versions of scientific, mathematical, and visualization applications recommended by XSEDE for best compatibility with other XSEDE clusters. [5]

The motivations for creating the XCBC software distribution include:

- From the standpoint of supporting researchers and cluster administrators on campuses, the XCBC build helps these staff automate those cluster creation and administration processes that are straightforward to automate. This allows staff on campuses throughout the US – who often have more work to do than time to do it – to focus on serving their local users with particular attention to local needs.

XSEDE '14, July 13 - 18 2014, Atlanta, GA, USA

ACM 978-1-4503-2893-7/14/07…$15.00.
http://dx.doi.org/10.1145/2616498.2616578

- From the standpoint of XSEDE in its goals to support the national research community, the distribution of these tools will lead over time to the creation of more clusters that are set up in a relatively consistent way – consistent with each other and consistent with the open source software setup of the least esoteric of the XSEDE clusters. (The XCBC is based heavily on the open source software installed on the TACC Ranger and Stampede systems). This consistency will make it easier to re-use training materials created for and by XSEDE systems.
- Over the long run, the inclusion of integration tools – such as Globus Online, Execution Management Services (EMS), Global Federated File System (GFFS), etc. – will make it easier to integrate clusters on campuses and XSEDE into a well-integrated national cyberinfrastructure. The option of including tools on local clusters will make it easier for the US open research community to align its cyberinfrastructure around XSEDE.

We see this project as an Education, Outreach, and Training (EOT) effort because the goals – and much of the really hard work – are primarily in the EOT area. Our work with campus champions and information technology professionals at small colleges and universities – and Historically Black Colleges and Universities (HBCUS) in particular – has confirmed that it is typical that there are inadequate staff resources to administer and support local cyberinfrastructure resources. Our interactions with faculty at such schools have confirmed consistently that one of the most difficult types of time to get – officially as an allocation of a faculty member's effort – is time for curriculum development.

The outcome of our work is that faculty at small colleges with limited time for curriculum development can create a XCBC and make use of curriculum tutorials made about XSEDE resources, thus saving themselves the trouble of re-creating new materials specialized for local resources. Also, if faculty members happen also to be their own cluster administrators – as is often the case – the cluster administration becomes easier and less time consuming. There is relatively little that is being done in the XCBC project that is development of new technology. This project is primarily involved in packaging and distributing existing technology developed by XSEDE or others. The work is packaging the software in a way that is usable and easily used by the national open research community generally. (More information on the user needs that drove the creation of the XCBC is available in the XSEDE Campus Bridging Use Cases document. [6].

The Rocks-based XCBC distribution is wonderful for people building a cluster for the first time, or for people so unhappy with their current cluster configuration that they want to start over from scratch. In addition, the Rocks distribution of the XCBC build has had some important successes already. However, this approach may not be the best one in the case of a cluster that is already well set up and administered. Nationwide, there are more universities, colleges, departments, labs, and individuals with adequately managed clusters who want to add this functionality than there are such entities that want to build a cluster from scratch. XSEDE is thus following the lead of the Open Science Grid (OSG) in creating a mechanism for downloading and adding specific packages and functionality to an already working cluster to make it function in a way compatible with XSEDE clusters (in addition to, rather than instead of, the cluster's current capabilities) [7]. OSG pioneered the approach of distributing such software modules as RPMs (RedHat Package Manager). a similar model of packaging and distributing software was created. In addition to a cluster building mechanism, a means of creating an easily adoptable methodology was to be created. These packages can be downloaded and installed individually as needed to establish or maintain an XSEDE-compatible cluster.

Making RPMs easily usable can be done with a pair of tools called YUM and Puppet. YUM, an acronym for Yellowdog Updater Modified, is a set of tools for creating repositories of RPMs, and perhaps the most widely used tool developed from the Yellowdog Linux distribution. Puppet is an open-source configuration management tool that facilitates deployment, configuration, and management of servers.

In the rest of this paper, we describe the existing Rocks-based XCBC build, the XSEDE YUM repository with a collection of RPMs that enable an administrator to add the XCBC tools to an existing cluster, and the use of the Puppet configuration manager. Figure 1 is a flowchart that depicts which tool to use, and when, and then explain the use of these tools in the remainder of this paper.

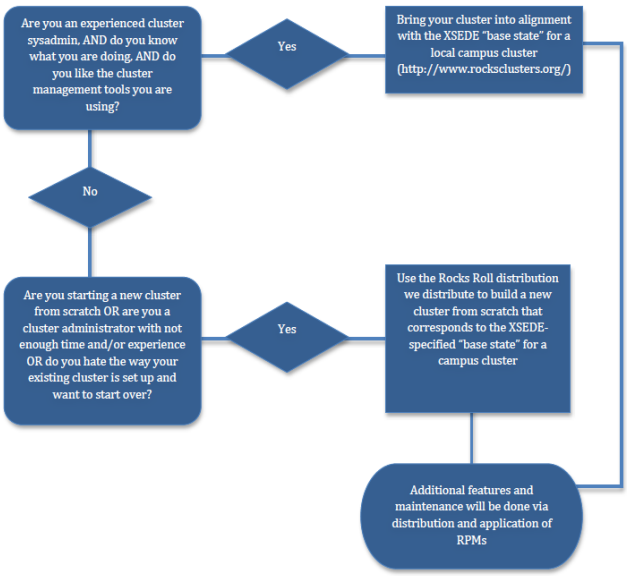

Figure 1. Campus Bridging Service Selection Flowchart [1]

2. ROCKS ROLLS

Rocks is an open source cluster distribution solution that simplifies the processes of deploying, managing, upgrading, and scaling high-performance parallel computing clusters. Rocks is designed to help scientists with little or no cluster experience build supercomputers that are compatible with systems used by national computing centers and international grids. [8]

The Rocks group has been addressing the problem of creating easily deployed and managed clusters since the turn of this century. [9] They have created a system to install a common Linux operating system (CentOS) and a means to manage computational nodes from a central (frontend) node. [3] This method creates a fairly simple to deploy basic cluster. Using an internal database, Rocks can manage many compute nodes. This management allows an administrator to easily add, remove, and upgrade software across nodes and to maintain a uniform

environment. This allows an institution with limited resources to easily create and maintain a cluster.

While Rocks uses a common Linux variant and packages, the Rocks method is a bit more rigid in design. This allows for a predictable, repeatable result in creating clusters. The development of Rocks is active but may often be slow to react to newer versions of software and operating systems. For instance, at the time of writing this paper, the current OS in Rocks 6.1 is CentOS 6.3. The version of CentOS in Rocks 6.1 is two revisions back from the current CentOS release of 6.5. Rocks expects to create a new version based on 6.5 during 2014, but this is an example of how Rocks may not be the best choice for institutions with higher IT staffing levels and cluster experience. Since the initial revision and acceptance of this paper, Rocks has released a new version of the OS, version 6.3, utilizing CentOS 6.5.

That aside, using the XSEDE roll during the Rocks cluster install will give a cluster the packages necessary to have an XSEDE-compatible cluster. Once you are up and running, to maintain the package levels, you can enable the XSEDE Yum repository and then follow the Rocks instructions or use the preferred method and create an update roll to add to your distribution. [10] The down side to Rocks upgrades is that neither method will most likely seem easy to the novice administrator. The long term result is that while clusters are relatively easy to bring online and expand, upgrading and other more in-depth maintenance may be daunting to less experienced users. This may lead to clusters not being maintained, not kept secure, and not kept upgraded with the latest XSEDE-compatible cluster software. These problems aside, Rocks may be the best solution for getting an XSEDE compatible cluster up for institutions that may have to depend on graduate students, faculty, or shared IT staff for installing and maintaining an XSEDE-compatible cluster.

3. USING YUM

Yellowdog updater modified (Yum) is a package manager that was developed by Duke University to improve the installation of RPMs. [11] Yum helped solved the problem of dependency issues for RedHat based installations by checking not only for updates but also for any updates that are a dependency for the initial package being updated. Using a series of repository configuration files, it can check as many or as few repositories as the administrator would prefer.

Yum itself does not provide any cluster capabilities. It merely provides a mechanism for easily maintaining packages. To utilize Yum in creating an XSEDE-compatible cluster, an administrator would need to initially set up the repository configuration. There are two ways to do this: (1) install the XSEDE repo RPM from http://cb-repo.iu.xsede.org/xsederepo or (2) install the yum-plugin-priorities package and then create the file /etc/yum.repos.d/xsede.repo with the lines specified in http://cb-repo.iu.xsede.org/xsederepo/readme.xsederepo.

As new packages are created, when "yum update" is called, it will find any new packages in the repositories your server is using and will try to resolve any dependencies for those packages and then provide the administrator with a full list of packages to be updated. Yum still requires an administrator to run update checks periodically. There are tools available (or admins can write their own scripts and cron jobs) to either automate Yum updates or to notify administrators of the availability of package updates.

Updating packages automatically may be a dangerous proposition in a production environment, causing unexpected issues to arise. Creating a notification script so that packages may be reviewed and tested on non-production nodes or systems might be the more prudent action. There are several tools available that do this such as yum-updatesd developed by Duke and available from CentOS and other distribution packagers.

4. USING PUPPET

Puppet is an open-source configuration management tool created by Puppet Labs. Puppet provides methods to deploy, configure, manage and maintain servers and can be deployed on a variety of Linux based operating systems including RedHat/CentOS variants, Debian and Ubuntu, as well as Mac OS X and Windows. Once you install puppet on all nodes and the puppet-master package on the frontend node and do the necessary configuration, you can use Apache and Puppet to maintain packages and configurations from the frontend node.

Using Puppet alone doesn't give any direct high performance computing capabilities. An administrator can use Puppet to install XSEDE-compatible software and bring an XSEDE-compatible cluster online fairly easily. Utilizing Puppet and specific recipes/patterns in conjunction with the XSEDE Yum repository can ensure that you have an up to date XSEDE-compatible cluster at all times.

Puppet is extremely flexible and allows an almost endless array of management tasks to be performed. You can find Puppet recipes for building/installing new servers, managing servers/users, maintaining content management systems such as Drupal, and the list goes on and on. To that effort, creating downloadable recipes for those already using Puppet to easily deploy an XSEDE-compatible cluster would be yet another way of ensuring a compatible research infrastructure. Beyond that, developing a base set of recipes and documentation for deploying an XSEDE-compatible cluster using Puppet to add XSEDE software to an existing cluster would be a logical next step for Campus Bridging efforts within XSEDE. Taking that idea further might be to create a Kickstart image and basic Puppet recipes to allow people to create an XSEDE compatible cluster from scratch outside of Rocks.

5. CASE STUDIES

5.1 Marcus Alfred, Howard University

Howard University "is one of only 48 U.S. private, Doctoral/Research-Extensive universities, comprising 12 schools and colleges with 10,500 students enjoying academic pursuits in more than 120 areas of study leading to undergraduate, graduate, and professional degrees. The University continues to attract the nation's top students and produces more on-campus African-American Ph.Ds. than any other university in the world" [12]

Dr. Marcus Alfred is an Associate Professor of Physics at Howard University. "Howard University is an active research university, but without a centralized HPC (high-performance computing) Center," says Marcus Alfred, professor of physics, Howard University. As a researcher working in computational nuclear physics, Dr. Alfred manages his own computing cluster out of necessity. He was so enthused about the capabilities of the basic XSEDE-compatible compute cluster that he restarted his cluster setup from scratch using the XCBC build from the Rocks Roll. As he put it, "the time of faculty members is precious, and the ease of an XSEDE Rocks Roll makes this unbeatable as a help for us to manage our clusters."

5.2 DAUIN HPC Initiative

The Department of Control and Computer Engineering (DAUIN) is an organization that "conducts research and teaching" and "consists of over 60 teachers and researchers, nearly 100 graduate students…and about 20 technicians and administrative staff." [13] Beginning in 2008, the DAUIN HPC initiative sought to create usable HPC resource in a time of economic stagnation. Their goals were terascale processing power, multipurpose computing using open source software, making it a shared resource, all while working in a modest budget of 13,500 Euros. [14]

The OS choice for the Casper_3 project was Rocks. This choice was made because it was based on CentOS, integrates OS and management layers, easy to install, manage, and add nodes, and helped maximize the benefits of HPC on a small resource. Rocks also gave them pre-configured software for an HPC environment. [14] All of these add up to create a cluster environment that would be easier to support than some alternative means.

This sort of small cluster capability is exactly what the XSEDE compatible cluster project strives for. While DAUIN is part of the domain of the Partnership for Advanced Computing in Europe (PRACE), a sister organization to XSEDE, the problems are the same whether in the United States or abroad. There is limited funding for hardware, especially for smaller organizations, and the human resources for supporting HPC projects continues to be limited, as well. Organizations such as DAUIN might also benefit from the software provided in an XSEDE compatible cluster.

5.3 ATLAS Tier 3 Analysis Cluster

Another approach to creating an XSEDE compatible cluster would be to embrace puppet. Puppet would require a bit more planning on both the part of Campus Bridging staff as well as the individuals installing the cluster. Rocks excels at being a cluster solution for people that need a cluster and do not have a large base of system administration knowledge or personnel to work with. By and large, you can follow the instructions and have a running cluster fairly quickly.

Puppet, on the other hand, is built as a tool for system administrators. It does require a little more knowledge on the front end but also allows for a much finer tuning and construction set from the start. It allows for more complete customization. Some Rocks installations with multiple frontends also use Puppet to control and maintain them.

Deploying and maintaining a cluster can be a very time consuming task requiring very specialized knowledge. With some help on the planning and preparation side, a turnkey deployment can be achieved with Puppet. Hendrix, et al, using a cluster definition consisting of complete Puppet configurations, kickstart installation scripts, and post installation scripts created a means of using a bootable USB stick. [15] Once booted and operating system is installed, puppet takes over and installs the software and configurations the cluster designer included in the definition. Adding or replacing nodes becomes a matter of booting with the USB stick for system nodes. Maintaining or making changes becomes a matter of creating a puppet job to change whatever parameters might need adjusting.

6. CONCLUSIONS

We have found the Rocks Roll distribution of the XCBC build to be useful, but it is useful only in two circumstances: someone is setting up a cluster from scratch; or someone is so unhappy with their cluster setup that they are willing to start over from scratch using the Rocks Roll. We've demonstrated – with Professor Marcus Alfred of Howard University – that the second case actually exists. But these cases are also not usual. Much more often, a lab, group, or campus will have a cluster that is already well to very well set up, but the administrators and users will see value in adding software to create compatibility and/or interoperability with XSEDE clusters. For this purpose, the combination of Puppet and a YUM repository of RPMs is a very practical choice – much better than starting over from scratch. We have demonstrated the value of this approach as well.

The NSF budget cannot meet the collective cyberinfrastructure needs of all open research activities in the US, any more than the NSF budget could have funded all of the networking needs of open US researchers in the 1990s. Leadership and funding of NSFNET aligned networking technology throughout the US to create the current modern Internet. In a way that is analogous (although technically different), we believe that XSEDE and the XCBC" build can align open US computational cyberinfrastructure. There is a technical difference: network standards create a stronger need for standards compliance in order to achieve interoperability than is needed for interoperability of computational clusters. But the concept of aligning effort and achieving economies of scale applies to the case of XSEDE and clusters throughout the US generally. While no sensible person would diminish the importance of cloud computing, there are many benefits to the possession and management of local and locally owned compute clusters as well.

The XCBC concept, the software build, and the distribution tools we discuss here make it easier for faculty members, IT experts, and campus champions across the US to adopt technology that is consistent with and enables interoperability with XSEDE. The ongoing efforts by the XSEDE Campus Bridging group will continue to help integrate these resources with the larger XSEDE community. Tools like the Genesis II software that will allow data to be shared amongst XSEDE resources and EMS (Execution Management Services) that allow jobs to be shared over XSEDE resources will further make the digital divide smaller, allowing researchers to easily share data and perform research on said data. [16] These tools will be particularly valuable to smaller schools with relatively more overworked and underfunded IT facilities and staff. For that reason, this is an important XSEDE effort but one that falls firmly in the category of education, outreach, and training activities.

7. ACKNOWLEDGMENTS

This document was developed with support from National Science Foundation (NSF) grant OCI-1053575. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF.

8. REFERENCES

[1] Stewart, C.A., R. Knepper, J.W. Ferguson, F. Bachmann, I. Foster, A. Grimshaw, V. Hazlewood and D. Lifka. What is campus bridging and what is XSEDE doing about it? 2012. Presentation. Presented at: XSEDE12 (Chicago, IL, 16-20 Jul 2012). http://hdl.handle.net/2022/14599

[2] XSEDE Software Repository. http://software.xsede.org/cb/centos6/x86_64/

[3] Rocks – Open Source Toolkit for Real and Virtual Servers. http://www.rocksclusters.org/wordpress/

[4] Fischer, Jeremy L. and Brown, Paul. 2013. What software is installed on a "bare-bones" XSEDE-compatible Rocks cluster? *Indiana University Knowledge Base* https://kb.iu.edu/data/bdww.html

[5] Fischer, Jeremy L. and Brown, Paul. 2013. What is the XSEDE Yum Repository and how do I use it? *Indiana University Knowledge Base* https://kb.iu.edu/data/bdwx.html.

[6] Stewart, Craig A.; Knepper, Richard; Grimshaw, Andrew; Foster, Ian; Bachmann, Felix; Lifka, David; Riedel, Morris; Tueke, Steven. 2012. XSEDE Campus Bridging Use Cases.

[7] Hazlewood, Victor, Knepper, Richard, Lee, Steven, Lifka, David, Navarro, JP, Stewart, Craig A. 2013. XSEDE Campus Bridging – Cluster software distribution strategy and tactics. http://hdl.handle.net/2022/15459

[8] Fischer, Jeremy L. and Brown, Paul. 2013. What is the XSEDE Rocks roll, and how do I use it? *Indiana University Knowledge Base* https://kb.iu.edu/data/bdwx.html.

[9] About Rocks. http://www.rocksclusters.org/wordpress/?page_id=57

[10] The Rocks Group. Base User Guide Rocks 6.1. Chapter 8: Advanced Tasks – 8.10 System Updates. http://central6.rocksclusters.org/roll-documentation/base/6.1/update.html

[11] RedHat Knowledge Base. "What is yum and how to use it?" https://access.redhat.com/site/solutions/9934

[12] Howard University. "Brief History of Howard University." http://www.howard.edu/explore/history.htm

[13] DAUIN Department of Control and Computer Engineering. "General Information." http://www.dauin.polito.it/it/informazioni_generali

[14] Croce, F. Della, Nepote, N.,Piccolo, E. "A Terascale Cost-Effective Open Solution for Academic Computing: Early Experience of the Dauin HPC Initiative." http://www.dauin-hpc.polito.it/papers/aica2011.pdf

[15] Hendrix, Val, Benjamin, Doug, Yao, Yushu. "Scientific Cluster Deployment and Recovery – Using puppet to simplify cluster management." 2012 *J. Phys.: Conf. Ser.* **396** 042027 doi:10.1088/1742-6596/396/4/042027

[16] XSEDE. "Campus Bridging." https://www.xsede.org/campus-bridging