Nuts and Bolts: Lessons Learned in Creating a User-Friendly FOSS Cluster Configuration Tool

Presenters:
Barbara Hallock, Indiana University, bahalloc@iu.edu
Resa Reynolds, Cornell University, rda1@cornell.edu
XSEDE Capabilities and Resource Integration

Presented 10/16/2017 at the 2017 Internet2 Technology Exchange in San Francisco, CA

Introduction

- XSEDE Capabilities and Resource Integration (XCRI) - formerly Campus Bridging – has for some time provided a Free Open-Source Software (FOSS) cluster configuration solution known as the XSEDE-Compatible Basic Cluster (more on that later)
- We also curate a YUM repository called the XSEDE National Integration Toolkit (XNIT) that contains useful FOSS for scientific applications

Introduction 2

- We also provide consulting to system administrators at universities around the US with regards to the selection and setup of these tools
  - Eric Coulter, XCRI Engineer, is currently on site at Slippery Rock University of Pennsylvania setting up a cluster in conjunction with Dell and SRU Administrators

Introduction 3

- The idea is to provide a way for under-resourced institutions to take (usually) decommissioned hardware and turn it into a usable cluster, although we have sites who purchase new hardware as well.

History of XCBC

- This configuration suite has been called the XSEDE-Compatible Basic Cluster, and has been through a few iterations:
  - The original XCBC was based on Rocks, a FOSS cluster configuration package being developed at the San Diego Supercomputer Center.
  - The new XCBC is based on OpenHPC.

History of XCBC 2

- Alongside the XCBC, the XCRI group has also curated a set of relocatable RPM packages in a repository we call the XSEDE National Integration Toolkit (XNIT)
- Due to the fact that the original XCBC did not allow admins to enable the XNIT repo, there was considerable overlap between the original XCBC and XNIT package lists.

Why choose a new XCBC?

- Overlap of effort between XCBC and XNIT – many packages duplicated, but not all
- End of Life issues with the underlying Operating System – CentOS 6
- Growing interest in SLURM from stakeholders – Rocks XCBC not compatible

Requirements for New XCBC

- XNIT compatibility
  - The new XCBC base was selected partially with extensibility in mind. This would allow XCBC sites to benefit from the XNIT, rather than being stuck using one or the other. Additionally, this means that the new XCBC is able to utilize packages released by other sources, opening up the ecosystem significantly

Requirements for New XCBC 2

- Open Source
  - Due to cost constraints associated with licensed software, the new XCBC had to be Free Open Source Software (FOSS). This allows for XCRI to provide the service at no cost to recipient institutions.

Requirements for New XCBC 3

- Under active development
  - Given the issues experienced with Rocks when CentOS 6 went EoL, XCRI prioritized the selection of an underlying set of packages that are being actively developed. This is not just a security necessity, but also provides administrators and users of the new XCBC with a much more robust community of practice from which to seek answers.

Requirements for New XCBC 4

- Able to run on a wide variety of hardware
  - The new XCBC needed to be compatible with a wide variety of hardware in a range of warranty statuses, so that it is able to run on the widest range of resources possible in order to minimize hardware costs for the recipient institutions, many of whom lack significant funding for cluster computing activities.

Requirements for New XCBC 5

- Better documentation and usability
  - In order to increase the range of users to which cluster computing is useful and usable, priority was placed on choosing a solution that was easy to deploy and maintain, with a robust community of practice contributing documentation to the main project.

Requirements for New XCBC 6

- Easy to install, allows for a variety of node types
  - A prime reason that Rocks was chosen in the initial XCBC toolkit was the ease of installation, and the presence of pre-existing profiles for different types of nodes, which allows the cluster to conform to a wide variety of environments (different hardware, local identity providers, storage nodes, etc.)

"Stretch Goal" for New XCBC

- A "stretch goal" for the new XCBC was that it would be possible to create an XCBC out of virtual machines using the Jetstream service (https://jetstream-cloud.org) with a single click of a button, thus streamlining the install process significantly over any pre-existing cluster configuration software.

The Chosen Solution

- XNIT Compatibility
- FOSS
- Under active development
- Able to run on a wide variety of hardware
- Better documentation and usability
- Ease of Installation, variety of node types
- Stretch goal

OpenHPC

- OpenHPC is a collaborative, community effort that initiated from a desire to aggregate a number of common ingredients required to deploy and manage High Performance Computing (HPC) Linux clusters including provisioning tools, resource management, I/O clients, development tools, and a variety of scientific libraries. Packages provided by OpenHPC have been pre-built with HPC integration in mind with a goal to provide re-usable building blocks for the HPC community.... The community includes representation from a variety of sources including software vendors, equipment manufacturers, research institutions, supercomputing sites, and others.
- (https://openhpc.community)

The Chosen Solution: OpenHPC

- XNIT Compatibility...✓
- FOSS...✓
- Under active development...✓
- Able to run on a wide variety of hardware...✓
- Better documentation and usability...✓
- Ease of Installation, variety of node types...✓*
- Stretch goal...X

The Missing Pieces

- FOSS automation tools
  - OpenHPC provides all of the necessary pieces to build an HPC system, but does not prescribe an installation method – all bits are provided as RPMs, which is excellent for a seasond administrator or team, but makes life more difficult for the new user. It also allows for infinitely configurable nodes, which does not help the new user/admin.
  - FOSS configuration management tools fit the bill perfectly, allowing XCRI to offer pre-configured sets of nodes and a well-supported framework for configuring a cluster headnode from bare metal, using the pieces offered by OpenHPC

Ansible

- FOSS configuration management
  - Out of a variety of options, Ansible seemed the best fit for an HPC system
  - User-activated, push-only, so it does not interfere with the resource manager (SLURM) or cluster management software (Warewulf)
  - Used to ease configuration of the headnode and creation of images for different node types
  - Bonus: Easily extends to a variety of environments (TO THE CLOUD!)

The Chosen Solution: OpenHPC+Ansible

- XNIT Compatibility...✓
- FOSS...✓
- Under active development...✓
- Able to run on a wide variety of hardware...✓
- Better documentation and usability...✓
- Ease of Installation, variety of node types ✓
- Stretch goal...✓

Putting the Pieces Together

- OpenHPC provides all the necessary software to build and manage a basic cluster, but without scientific software or pre-defined node templates
- Ansible brings the pieces from OpenHPC and the XNIT together, and allows for the creation of templates for node types, or installation of non-packaged software

The First Implementation

- Developed the toolkit internally for several months, testing on available hardware and VMs, until our next site visit
- No matter how much testing you do, there will be bugs in production!
- No matter how much testing you do, there will be bugs in production!
- No matter how much testing you do, there will be bugs in production!

The First Implementation: Brandeis

- The ideal situation is to deploy with a Friendly User, which we were lucky enough to have!
- Francesco Pontiggia, lone admin and research support at Brandeis University's Division of Science HPC Center
- Needed to update an old cluster while keeping as much of it running as possible

The First Implementation: Brandeis

- Repeat this to yourself three times: THERE WILL BE HARDWARE ISSUES
  - Sometimes, it takes all day just to install the base OS on your chosen headnode, and that's ok.
- Informal whiteboarding is a *good* idea.
  - The initial build led to massive changes in the toolkit, which would not have been possible without Francesco and our whiteboard notes.

The First Implementation: Brandeis

- Identity Management is *HARD*
  - The most troubling node type to design was the login node.
- GPU installation on Linux is also hard!
  - Especially when dealing with older hardware, requiring precise versions of drivers and CUDA libraries – this makes gpu-node template building very difficult, when even the driver installation scripts have different flags across versions.

The First Implementation: Brandeis

- Ended with a working cluster, and three basic node templates: Compute, Login, and GPU
- Better yet, we had a defined way of creating new node templates, and easily customizing them to fit the needs of a particular site
- The XCBC is an evolving project – each new site brings improvements and additions to the available configuration
- Starting off with a friendly user build/collaboration was essential!

Attaining the Stretch Goal

- Subsequent to the Brandeis visit, a number of the practical lessons from the install informed the process of automating a cluster build on Jetstream
- Once again, the OpenHPC+Ansible recipe was successful, with additional simplifications / complications due to the cloud environment
- The toolkit for Virtual Jetstream clusters is available, but currently lacking in documentation – if you are interested, please let us know! (We can help you build one quickly, but going it alone will be hard)

XCBC Aggregate Totals

Category	Total
Number of XCBC sites	11
Number of Nodes	457
Most Nodes	145
Fewest Nodes	4
Number of Cores	5989
Max Rpeak (Tflops)	200
Min Rpeak (Tflops)	0.26
Old XCBC Sites	8
New XCBC sites	3

For more information...

- https://www.xsede.org/ecosystem/resource-integration XSEDE Capabilities and Resource Integration
- https://portal.xsede.org/knowledge-base/-/kb/document/bdwx What is the XNIT and how do I use it?
- https://portal.xsede.org/knowledge-base/-/kb/document/aoje What packages are included in the XCBC and XNIT suites?
- https://portal.xsede.org/knowledge-base/-/kb/document/aojf What optional software does XCRI provide for HPC cluster administrators?
- https://github.com/XSEDE/ XSEDE GitHub repository
- https://openhpc.community OpenHPC home page
- https://jetstream-cloud.org Jetstream home page

Q+A / Contact info

- help@xsede.org
- Just mention XCRI in the subject and it will get to us

Acknowledgments

- This work was supported, in part, by awards from the National Science Foundation (Jetstream: ACI-1445604; XSEDE: OCI-1053575, OCI-0948142, OCI-1059812), and by the IU Pervasive Technology Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or Pervasive Technology Institute.
- The presenters wish to acknowledge Eric Coulter (jecoulte@iu.edu) and Richard Knepper (rich.knepper@cornell.edu) for their assistance in preparing these materials.