Workshop Report:
Campus Bridging: Reducing Obstacles on the Path to Big Answers

Presented in conjunction with IEEE Cluster 2015
on September 8, 2015 in Chicago, IL

Workshop Organizers: Barbara Hallock, Yashema Mack[1], Resa Reynolds[2]

Editors: Barbara Hallock, Richard Knepper, Winona Snapp-Childs, Craig A. Stewart

Contributing Writers: Dustin Atkins[3], Jim Bottum[4], Eric Coulter[5], Ian Foster[6], Robert Gardner[7], Andrew Grimshaw[8], Barbara Hallock[9], Richard Knepper[10], Lee Liming[11], Miron Livny[12], Charles Peck[13], Rob Quick[14], Steve Tuecke[15], Craig A. Stewart[16]

[1] ymack@utk.edu / https://www.nics.tennessee.edu/ / National Institute for Computational Sciences / Knoxville, TN
[2] rda1@cornell.edu / https://www.cac.cornell.edu/ / Cornell University / Ithaca, NY 14850
[3] datkin2@clemson.edu / https://aci-ref.org/ / Clemson University / Clemson, SC
[4] jb@clemson.edu / https://aci-ref.org/ / Clemson University / Clemson, SC
[5] jecoulte@iu.edu / https://pti.iu.edu/ / Indiana University / Bloomington, IN 47401
[6] foster@uchicago.edu / https://globus.org / University of Chicago / Chicago, IL
[7] rwg@uchicago.edu / https://globus.org/ University of Chicago / Chicago, IL
[8] grimshaw@virginia.edu / https://genesis2.cs.virginia.edu / University of Chicago / Chicago, IL
[9] bahalloc@iu.edu / https://pti.iu.edu/ / Indiana University / Bloomington, IN 47401
[10] rknepper@iu.edu / https://pti.iu.edu/ / Indiana University / Bloomington, IN 47401
[11] lliming@uchicago.edu / https://globus.org / University of Chicago / Chicago, IL
[12] miron@cs.wisc.edu / https://www.cs.wisc.edu/ / University of Wisconsin /
[13] charliep@cs.earlham.edu / https://littlefe.org / Earlham College / Earlham, MA
[14] rquick@iu.edu / https://opensciencegrid.org/ / Indiana University / Bloomington, IN 47401
[15] tuecke@uchicago.edu / https://globus.org / University of Chicago / Chicago, IL
[16] stewart@iu.edu / http://pti.iu.edu/ / Indiana University / Bloomington, IN 47401

Table of Contents

1. Executive Summary

In September 2015, XSEDE Campus Bridging staff hosted a workshop titled "Campus Bridging: Reducing Obstacles on the Path to Big Answers." The workshop was held in conjunction with IEEE Cluster 2015 in Chicago, IL, and featured talks from a number of subject matter experts who have been involved with organized campus bridging efforts at academic computing centers around the United States.

The workshop was organized by Barbara Hallock of Indiana University, Yashema Mack of the National Institute for Computational Sciences (NICS), and Resa Reynolds of Cornell University, and was held on Tuesday, September 8.

A significant portion of the workshop was dedicated to substantive discussion about what strategies campus bridging professionals in various organizations are successfully employing in order to "bridge" the gap between researchers and computational resources. A wide group of professionals at both national and regional organizations who are themselves working on campus bridging came together to discuss their particular approach to the issue; what strategies were successful; and to discuss what approaches they could take singly and in concert moving forward.

With the first XSEDE grant coming to a close, the workshop also provided the opportunity for XSEDE Campus Bridging staff and management to gather information that would inform their priorities in drafting the proposal for the organization to follow XSEDE. One of the most significant points articulated during the workshop was that campus bridging in general had not excited the computer and computational science communities because it had not been presented in the form of a computer science problem. Campus bridging was formulated as a set of problems commonly faced by people working on campuses, deploying or using campus cyberinfrastructure. The commonality in defining campus bridging was the combination of who felt this set of problems and that they were felt on campuses as people looked in a particular direction – from their campus toward the national cyberinfrastructure.

2. Introduction

As a part of this workshop, seven talks were given. In addition to the talks, two papers were published in the conference proceedings: "*Building Bridges from the Campus to XSEDE*" [1], and "*The XSEDE-compatible basic cluster – A tool for cluster implementation and management in research and training* [2].

In Order of Presentation:

- *Campus Bridging through Facilitation: the ACI-REF Project* (Jim Bottum and Dustin Atkins)
- *HTC Campus Queue* (Rob Quick)
- *The XSEDE-compatible basic cluster – A tool for cluster implementation and management in research and training* (Jeremy Fischer, Eric Coulter, Richard Knepper, Charles Peck and Craig Stewart)
- *Building Bridges from the Campus to XSEDE* (Lee Liming, Ian Foster and Steve Tuecke)
- *CI Connect: a Service for Building Multi-Institutional Campus Cluster Environments* (Robert Gardner)
- *Simple Secure Resource Sharing with the XSEDE Global Federated File System (GFFS) and Execution Management Services (EMS)* (Andrew Grimshaw)
- *Submit Locally and Run Globally* (Miron Livny)

2.1. Motivation and Background.

As the initial XSEDE NSF award would be drawing to a close within the next year, it was necessary to define a vision for Campus Bridging in the national cyberinfrastructure community as it would exist going forward. In addition, the need for this workshop was motivated by the distributed nature of Campus Bridging efforts and experts; technical conferences are one of the better opportunities for staff from disparate organizations to come together for discussion and planning.

Thus, it was important to identify the right group of speakers to bring together for the workshop. In addition to practitioners currently engaging in campus bridging activities across various spheres, the workshop featured three participants who were members of the original NSF Advisory Committee for CyberInfrastructure (ACCI) Task Force on Campus Bridging [3].

Additionally, co-location with a technical conference focused on Cluster computing provided the potential for new exposure to professionals working in the domain, as well as the opportunity to solicit input from deeply engaged practitioners of HPC. The venue also provided an opportunity to clarify exactly what kind of work XSEDE Campus Bridging staff are engaged in with those professionals who are likely to benefit most from it.

2.2. Preparation for the Workshop.

A team of Campus Bridging professionals was identified to organize the workshop. Barbara Hallock of Indiana University served as main organizer, with assistance by Yashema Mack of the National Institute for Computational Sciences and Resa Reynolds of Cornell

University. The organizers, with input from XSEDE Campus Bridging management, identified a list of people who have been actively engaged in the XSEDE Campus Bridging efforts and invited a number of them to submit abstracts for the workshop.

The speakers prepared slides and presentations to be delivered onsite at the workshop in Chicago, Illinois. In two cases, they also produced papers, which were published in the proceedings of IEEE Cluster 2015, with which the workshop was co-located.

The Indiana University Pervasive Technology Institute (PTI) provided assistance with advertising the workshop in advance of the conference, as well as hosting the conference web page [4], which the organizers prepared and updated in advance of the workshop with the agenda and the confirmed speakers. After the workshop, PTI provided assistance in updating the web page to include PDFs of the slide decks used by each speaker at the workshop, as well as a photograph taken during a presentation by Jim Bottum of Clemson University.

3. Context and Definitions

3.1 Defining Campus Bridging.

Beginning in 2009 and culminating in a report produced in 2011, the National Science Foundation convened its ACCI Task Force on Campus Bridging. The report generated at the conclusion of that Task Force's work formed much of the basis upon which XSEDE Campus Bridging as a group would define its priorities within XSEDE. Three members of that Task Force – Jim Bottum, Miron Livny, and Craig A. Stewart – were present to participate in the workshop. In that report, campus bridging is defined thusly:

"the goal of campus bridging is to enable the seamlessly integrated use among: a scientist's or engineer's personal cyberinfrastructure; cyberinfrastructure on the scientist's campus; cyberinfrastructure at other campuses; and cyberinfrastructure at the regional, national, and international levels; so that they all function as if they were proximate to the scientist. When working within the context of a Virtual Organization (VO), the goal of campus bridging is to make the 'virtual' aspect of the organization irrelevant (or helpful) to the work of the VO" [3].

The ultimate goal of campus bridging can be described thusly: to create an experience of interacting with cyberinfrastructure, no matter where it is located, as easily and seamlessly as if it were simply a peripheral attached to the researcher's own laptop.

3.2 Defining Cyberinfrastructure.

In order to truly appreciate the scope of what campus bridging aims to do, it is necessary to arrive at a working definition for cyberinfrastructure. For the purposes of this workshop, we utilized the same definition from which the ACCI Task Force worked:

"Cyberinfrastructure consists of computational systems, data and information management, advanced instruments, visualization environments, and people, all linked together by software and advanced networks to improve scholarly productivity and enable knowledge breakthroughs and discoveries not otherwise possible" [5].

Thus, campus bridging is not concerned merely with one particular facet of technology, but rather aims for a multidisciplinary approach to the problem of improving access to computational resources; additionally, human factors are equally important to, if not more important than, technical ones in the campus bridging equation.

Campus Bridging in context within XSEDE. The eXtreme Science and Engineering Discovery Environment (XSEDE) is a sociotechnical system which provides access, software, and support for the use of top-tier academic supercomputing centers around the country [6]. Funded by NSF, XSEDE's mission is to provide large-scale advanced cyberinfrastructure to academic projects. The XSEDE network consists of 16 supercomputers and data visualization resources around the country [7], and a diverse VO to support its operations. XSEDE Campus Bridging exists under the organizational umbrella of XSEDE Training, Education, and Outreach Services (TEOS) [8], which greatly informed its

priorities as an organizational unit. The XSEDE Campus Bridging team is charged with providing improved access to the national cyberinfrastructure in aggregate, facilitating access to resources on campuses, in regional partnerships, and to XSEDE Service Providers.

4. Existing XSEDE Campus Bridging Initiatives

4.1 The XSEDE-Compatible Basic Cluster
Based on a Rocks Cluster Management [9] and CentOS foundation, the XCBC [10] consists of everything an administrator would need to take a room full of bare metal hardware and turn it into a fully functioning cluster, plus a Rocks roll of various Open Source Software (OSS) packages in various scientific domains that are in use on XSEDE SPs. The XCBC was developed as a solution for administrators who need to take a room full of existing hardware and turn it into a cluster, or who need to do a complete rebuild on an existing cluster. Eric Coulter presented about XCBC and XNIT at the workshop.

4.2 The XSEDE National Interoperability Toolkit
In addition to XCBC, XSEDE Campus Bridging curates and maintains a YUM repository of OSS packages that is collectively known as the XNIT [11]. The purpose of the XNIT is to allow administrators who already have a cluster up and running, but would like to extend its capabilities, to leverage XSEDE resources (including an intensive vetting process that all packages must go through before being deployed on XSEDE SPs) and decrease the amount of administrative overhead required to keep all the various packages installed on the system up-to-date. In addition, the packaging team at Cornell University has made extensive progress toward the eventual goal of making every RPM in the XNIT relocatable. Additionally, in its efforts to expand and improve the XNIT, XSEDE Campus Bridging solicits requests for OSS packages to be added to the repository on an ongoing basis; one package that was added as a result of community request was the SLURM scheduler [12].

4.3 Globus
The Globus [13] suite of tools had, as of the workshop, been installed on all XSEDE Tier I Resources for over a year, and both the number of users and the number of terabytes (TB) of data moved to and from those resources has grown with each quarter. One chief benefit of Globus is its intuitive user interface, which allows users to "set it and forget it" – they simply drag and drop files in the Globus client to start the file transfer and walk away; Globus sends them an e-mail when the transfer is finished and they can resume working with the data.

4.4 Technical Writing and Education, Outreach, and Training (EOT)
As a unit within the TEOS, XSEDE Campus Bridging by necessity has given high priority to these types of activities. As of the workshop, XSEDE Campus Bridging staff members were in the process of developing a webinar highlighting and demonstrating XCBC and XNIT. Considerable effort had already been devoted by those staff to documenting the usage and contents of XCBC and XNIT, as well as outreach efforts singly and in concert with the larger TEOS team. Members of the XSEDE Campus Bridging team have also completed numerous outreach events such as talks and posters at a number of conferences.

4.5 Support and Consulting
In addition to more general outreach efforts, XSEDE Campus Bridging offers support and consulting via a number of options – by e-mail to help@xsede.org 24/7/365; by phone 1-

866-907-2383 8 a.m. – 5 p.m. Eastern time, Mondays through Fridays except holidays; and in person by arrangement onsite at campuses who have reached out to request assistance using XCBC or XNIT.

All consulting is provided at no charge to the recipient institutions, and if applicable, consultant travel expenses are paid for by XSEDE. Campus Bridging engineers work with their point of contact at the recipient institution to determine what services are required and what preparation needs to be done on XSEDE Campus Bridging's part and what needs to be done by the recipient institution. Often, this is limited to a few relatively simple tasks such as recommending lodgings, arranging parking passes, and making sure that any new hardware is plugged in correctly and configured to boot in a certain manner. This ensures a smooth install process and allows more of the visit to be devoted to education and training efforts, if the recipient institution requests those.

4.6 Federated Login Mechanism

One of the major hurdles to seamless use of geographically disparate CI resources is that they sometimes fall under different login domains. On XSEDE resources, this issue has been resolved with the use of an XSEDE Portal username, which ties together a user's accounts on each system they have allocations for. However, this is not necessarily a viable solution for universities wishing to allow collaboration between their users and those from other institutions. XSEDE Campus Bridging has engaged in ongoing efforts to leverage the effort and lessons provided by XSEDE's efforts to resolve this problem to identify some sort of federated identity management mechanism that could be configured to allow resource sharing without requiring accounts to be created for the guest users at each institution.

4.7 Campus Queue

Though the bulk of XSEDE Campus Bridging activities are centered on HPC, there has also been effort allocated to High-Throughput Computing (HTC) approaches to campus bridging. The campus queue project is a joint effort between XSEDE Campus Bridging and the Open Science Grid (OSG) [14] to develop a pool of shared resources between the two. Rob Quick presented on the status of the Campus Queue at the workshop; development efforts are ongoing as of the publication of this report.

4.8 GenesisII

The GenesisII software, developed at the University of Virginia [15], comprised a significant part of XSEDE Campus Bridging activities in the first half of XSEDE, in a beta program designed to improve the existing software and documentation base and vet it for deployment on XSEDE resources. XSEDE Campus Bridging staff members coordinated efforts between the XSEDE Security team, the GenesisII development team, and a number of friendly users selected from a call for proposals to ensure that GenesisII was up to the rigorous standards set forth by XSEDE for approval before a given package may be deployed on one or more XSEDE resources. Andrew Grimshaw, lead developer of GenesisII, was actively involved in the beta process, and presented a demo of the software at the workshop.

5. Exemplars of Success in Campus Bridging Efforts

5.1 Globus is helping more people move more data to and from XSEDE with each passing quarter.
The Globus transfer software has been deployed to all XSEDE Service Providers and is actively in use. This has been the most notable impact Campus Bridging activities have had on the XSEDE ecosystem; users are transferring hundreds of terabytes per quarter to and from XSEDE resources using Globus.

5.2 XCBC and XNIT are helping under-resourced institutions bring HPC to their users.
Thanks to the XCBC and XNIT projects, a number of small higher education institutions have been able to make use of donated and reclaimed hardware to provide HPC clusters to their students and faculty. As of the writing of this report, there are an estimated 9 universities with XCBCs and 48 universities pulling regularly from XNIT, based on IP traffic analysis. In addition, a number of universities have expressed strong interest in arranging site visits as of the publication of this report, and site visits will continue into XSEDE2.0.

5.3 Facilitation model of ACI-REF is already showing a number of successful applications.
ACI-REF representatives discussed a number of use cases in their talk wherein the facilitation-based approach had already garnered results at universities around the United States. XSEDE Campus Bridging and ACI-REF have continued to collaborate since the workshop, with a number of new joint efforts in various planning stages.

6. Findings

Finding 1
One of the biggest challenges to the advance of Campus Bridging activities is the "Campus Bridging" name itself. There was strong agreement that this was the case among those present.

The concept of campus bridging is not well understood outside of those who work in the specific domain, and those who do campus bridging often have slightly different conceptions of what it means, if indeed they are even aware of the concept (many activities which fall under the heading of "campus bridging" can also be characterized under other academic domains such as usability analysis, grid computing research, or science and technology innovation).

This presents a great challenge to the success of campus bridging initiatives. Sometimes, the name even leads those unfamiliar with the discipline to presume that XSEDE Campus Bridging's purview is much more narrow and less widely applicable than it actually is, which causes them to discard Campus Bridging resources as being irrelevant to their needs without ever determining what those services might even be.

Finding 2
One of the great successes of XSEDE Campus Bridging has been its toolkits. Globus is one such toolkit that has seen wide adoption and success across XSEDE, and is enabling the transfer of huge amounts of data to and from XSEDE resources. Additionally, the XCBC toolkit has been useful in bringing HPC to under-resourced universities which might not otherwise have access to it; of particular note, a number of these institutions are Minority-Serving Institutions (MSIs), which means that these tools are helping to increase the pool of professionals and academics who are familiar with cluster computing and HPC. By enabling easier access to these types of resources, Campus Bridging initiatives allow both early researchers and campus IT staff to become familiar with cyberinfrastructure tools and the broader community of cyberinfrastructure users. This modular approach to producing toolsets for diverse user populations could be a valuable paradigm to employ moving forward.

Finding 3
There lies a huge potential for expansion into the long tail of science, but this will necessitate a different set of approaches than traditional HPC markets. In particular, disciplines in Humanities and Social Sciences may have huge amounts of data that could benefit from a High-Performance approach – but they don't even realize it. These academic fields also tend to employ professionals who are not technically fluent with command line-based computing, which is a significant barrier to entry, that cannot be solved by the traditional approach to HPC. Not only does serving these communities necessitate a change in the services we deliver, but also to the ways we think and speak about them.

Finding 4

An increase in facilitation-based approaches to campus bridging is already closing pitfalls in the process of getting results, but more effort is required. The ACI-REF organization, though relatively new, is already seeing great success in facilitation-based approaches to campus bridging, which they discussed at the workshop. Facilitation-based strategies engage with researchers and campus IT staff directly around given problem areas and provide support and guidance in making use of all of the available resources. For example the ACI-REF program has funded a targeted facilitator responsible for identifying and assisting with the use of tools for the use of Geographic Information Services (GIS) in ACI-REF-affiliated institutions. Though there had already been some collaboration between XSEDE Campus Bridging and ACI-REF leading up to the workshop, the discussions reinforced the importance of this approach.

Finding 5

One of the most important challenges currently facing campus bridging practitioners is how to handle authentication. Solving this problem in a portable, reliable manner requires a great deal of thought and care, and remains a high priority for XSEDE Campus Bridging. Discussion among workshop participants underscored the importance of an authentication solution, and work to solve this problem will ramp up in XSEDE2.0. InCommon [16] is the mechanism XSEDE utilizes to do federated identity management; it was identified as a valuable avenue of research by a number of participants in the workshop.

7. Campus Bridging Moving Forward

One of the speakers in this workshop make the very perceptive observation that campus bridging in general had not excited the computer and computational science communities because it had not been presented, as of the date of this workshop, in the form of a computer science problem. This observation is absolutely correct, and it goes back to the fact that campus bridging was not formulated as a computer science problem. It was formulated as a set of problems that had as their commonality that they were problems faced by people working on campuses, deploying or using campus cyberinfrastructure. The commonality in defining campus bridging was the combination of who felt this set of problems and that they were felt on campuses as people looked in a particular direction – from their campus toward the national cyberinfrastructure. The final report of the ACCI campus bridging taskforce arrived at a set of findings, as follows:

Finding 1
The diversity in the US cyberinfrastructure environment creates tremendous opportunities for US science and engineering research, but adds new types of complexity and new challenges in campus bridging. The cyberinfrastructure environment in the US is now much more complex and varied than the long-useful Branscomb Pyramid. As regards computational facilities, this is largely due to continued improvements in processing power per unit of money and changes in CPU architecture, continued development of volunteer computing systems, and evolution of commercial Infrastructure/Platform/Software as a Service (cloud) facilities. Data management and access facilities and user communities are also increasingly complex, and not necessarily well described by a pyramid.

Finding 2
*The reward system as perceived by individual faculty researchers in science and engineering does not support the development of a coordinated national cyberinfrastructure. It encourages a highly diffuse, uncoordinated cyberinfrastructure that makes sharing and collective investment difficult and does not optimize the effectiveness of cyberinfrastructure support for research and development in science and engineering in the United States. In particular, the current reward structure does not align rewards to faculty with a focus on collaboration in ways
that support NSF's stated views on Virtual Organizations as an essential organizational structure in scientific and engineering research.*

Finding 3
The current state of cyberinfrastructure software and current levels of expert support for use of cyberinfrastructure create barriers in use of the many and varied campus and national cyberinfrastructure facilities. These barriers prevent the US open science and engineering research community from using the existing, open US cyberinfrastructure as effectively and efficiently as possible.

Finding 4
The existing, aggregate, national cyberinfrastructure is not adequate to meet current or future needs of the US open science and engineering research community.

Finding 5

A healthy national cyberinfrastructure ecosystem is essential to US science and engineering research and to US global competitiveness in science and technology. Federal R&D funding overall is not sufficient to meet those needs, and the NSF share of this funding is not sufficient to meet even the needs of basic research in those disciplines that the NSF supports.

Finding 6

Data volumes produced by most new research instrumentation, including that installed at the campus lab level, cannot be supported by most current campus, regional, and national networking facilities. There is a critical need to restructure and upgrade local campus networks to meet these demands.

Based on these findings that same task force made a number of recommendations. These recommendations, with comments about the nature of the problem identified and actions taken since the report was finalized are presented inline below:

Recommendations to the National Science Foundation

Strategic Recommendation to the NSF #1

As part of a strategy of coherence between the NSF and campus cyberinfrastructure and reducing reimplementation of multiple authentication systems, the NSF should encourage the use of the InCommon Federation global federated system by using it in the services it deploys and supports, unless there are specific technical or risk management barriers.

Comments on progress to date: This is essentially a recommendation about national trust relationships among cyberinfrastructure users and cyberinfrastructure providers, and the NSF has indeed pursued this recommendation with diligence. InCommon is now widely accepted and widely implemented as a mechanism for authentication, and it is used widely in large scale distributed computing facilities. As a result, it is much more common today than it was in 2011 that a researcher can use their home institution login credentials to access elements of the national cyberinfrastructure. This is a significant improvement over the situation at the time of the completion of the ACCI Campus Bridging Task Force Report. At the time of the completion of this report, this was not a computer science problem: the computer science research and development was done, and a robust solution to the problem of sensible approaches to authentication (as opposed to a separate set of credentials for each system on which one had rights) had been solved.

Strategic Recommendation to the NSF #2

The NSF must lead the community in establishing a blueprint for a National Cyberinfrastructure. Components of this leadership should include the following strategic approaches to funding cyberinfrastructure:

- When funding cyberinfrastructure projects that are intended to function as infrastructure, the NSF should use the review criteria and approaches that are generally used for research infrastructure rather than the criteria used for scientific

discovery awards. Such awards should be made in ways that complement existing infrastructure and align with best practices, appropriate international standards, and the NSF vision and plans for CIF21.

The NSF has implemented this recommendation to a very great extent, but it was never a computer science problem; it was essentially an infrastructure funding and infrastructure evaluation problem

- The NSF should establish a national cyberinfrastructure software roadmap. Through the Software Infrastructure for Sustained Innovation (SI2) or other programs, the NSF should seek to systematically fund the creation and ongoing development and support of a suite of critical cyberinfrastructure software that identifies and establishes this roadmap, including cyberinfrastructure software for authentication and access control; computing cluster management; data movement; data sharing; data, metadata, and provenance management; distributed computation / cycle scavenging; parallel computing libraries; network performance analysis / debugging; VO collaboration; and scientific visualization. Funding for personnel should be a strong portion of such a strategy.

The NSF has chosen not to create a national software roadmap per se, at least to date. It has chosen to fund through SI2 and other programs a small suite of truly critical software projects that influence the

- The NSF should continue to invest in campus cyberinfrastructure through programs such as the Major Research Infrastructure (MRI) program, and do so in ways that achieve goals set in the Cyberinfrastructure Vision for 21st Century Discovery and a national cyberinfrastructure software roadmap.

The MRI program continues to invest heavily in campus cyberinfrastructure, and now has the novel and so far highly successful attribute that one can propose local cyberinfrastructure, or at larger funding levels, one can propose campus cyberinfrastructure that is made available in part to the national research community such as the XStream computational resource now operated by Stanford University (NSF Grant No. ACI-1429830).

Strategic Recommendation to the NSF #3
The NSF should create a new program funding high-speed (currently 10 Gbps) connections from campuses to the nearest landing point for a national network backbone. The design of these connections must include support for dynamic network provisioning services and must be engineered to support rapid movement of large scientific data sets.

The CC* program and its predecessors (e.g. CC-NIE) were created in response to this recommendation. This was and always has been a financial and organizational strategy problem, not a computer science problem. The successful NSF funding programs and the ongoing decrease in the cost of network access have not made this a solved problem yet.. but substantial progress has been made since the writing of the ACCI campus bridging taskforce report.

Strategic Recommendation to the NSF #4
The NSF should fund national facilities for at least short-term storage and management of data to support collaboration, scientific workflows, and remote visualization; management tools should include support for provenance and metadata. As a complement to these facilities and in coordination with the work in Recommendation #3, the NSF should also fund the development of services for bulk movement of scientific data and for high-speed access to distributed data stores. Additionally, efforts in this area should be closely coordinated with emerging campus-level data management investments.

Success in this area has been mixed, with one such effort developed by XSEDE being discontinued for lack of users (for example, see XSEDE-Wide File System [17, 18]). The Wrangler data analytics and storage system is a current effort to address this and other data management problems and is meeting with some early success. Again, the storage issues per se are financial rather than computer science problems. The data movement problem identified in this recommendation has been, as described in this workshop, largely solved through Globus Online services.

Strategic Recommendation to the NSF #5
The NSF should continue research, development, and delivery of new networking technologies. Research priorities funded by the NSF should include data intensive networks, sensor nets, networking in support of cyberphysical systems, geographically distributed file systems, and technologies to support long distance and international networking.

The NSF is indeed funding the development of new networking technologies and these are areas of active computer science research.

Strategic Recommendation to the NSF #6
The NSF should fund activities that support the evolution and maturation of cyberinfrastructure through careful analyses of needs (in advance of creating new cyberinfrastructure facilities) and outcomes (during and after the use of cyberinfrastructure facilities). The NSF should establish and fund processes for collecting disciplinary community requirements and planning long-term cyberinfrastructure software roadmaps to support disciplinary community research objectives. The NSF should likewise fund studies of cyberinfrastructure experiences to identify attributes leading to impact, and recommend a set of metrics for the development, deployment, and operation of cyberinfrastructure, including a set of guidelines for how the community should judge cyberinfrastructure technologies in terms of their technology readiness. All NSF-funded cyberinfrastructure implementations should include analysis of effectiveness including formal user surveys. All studies of cyberinfrastructure needs and outcomes, including ongoing studies of existing cyberinfrastructure facilities, should be published in the open, refereed, scholarly literature.

The NSF has made some progress in this area, including funding the ACI-REF project that synthesizes and disseminates information about cyberinfrastructure systems.

When one boils all of the above down, one finds that there are now a discernable set of computer science problems that remain unaddressed from the original campus bridging task force final report. We hope that this workshop report will spur interest in these problems, which based on the above commentary we identify as follows:

- Cyberinfrastructure software for authentication and access control; computing cluster management; ~~data movement~~; data sharing; data, metadata, and provenance management; distributed computation / cycle scavenging; parallel computing libraries; ~~network performance analysis~~ / debugging; VO collaboration; and scientific visualization. *(We note that the workshop itself suggests that data movement is now, at least mostly, a solved problem. Network performance analysis is now largely a solved problem as well with widely used tools such as perfSONAR and Periscope. As described in the material presented in this workshop, cycle scavenging in terms of high throughput computing is a largely solved problem. Cycle scavenging in terms of parallel computing is attracting the attention of some computer scientists but remains an active area of research.)*
- Research, development, and delivery of new networking technologies. Research priorities funded by the NSF should include data intensive networks, sensor nets, networking in support of cyberphysical systems, geographically distributed file systems, and technologies to support long distance and international networking.
- Collecting disciplinary community requirements and planning long-term. cyberinfrastructure software roadmaps to support disciplinary community research objectives. The NSF should likewise fund studies of cyberinfrastructure experiences to identify attributes leading to impact, and recommend a set of metrics for the development, deployment, and operation of cyberinfrastructure, including a set of guidelines for how the community should judge cyberinfrastructure technologies in terms of their technology readiness. All NSF-funded cyberinfrastructure implementations should include analysis of effectiveness including formal user surveys. All studies of cyberinfrastructure needs and outcomes, including ongoing studies of existing cyberinfrastructure facilities, should be published in the open, refereed, scholarly literature.

The above do not constitute a cohesive area of computer science research. They are a collection of now distinct computer science challenges that have in common the fact that people interacting with local and national cyberinfrastructure. Campus bridging likely remains a useful term going forward, but now as always before a significant portion of the tractable problems in what was originally termed campus bridging are really issues of organizational strategy for campuses and financial prioritization. The above computer and computational science problems may usefully be referred to in some contexts as campus bridging. In other contexts, it may be useful to create new terms. For example, XSEDE has now taken the set of issues related to interoperability of campus cyberinfrastructure systems with national cyberinfrastructure, and termed that set of problems as *"XSEDE Community Infrastructure."* XSEDE states that, *"The mission of the XSEDE Community*

Infrastructure (XCI) team is to facilitate interaction, sharing and compatibility of all relevant software and related services across the national CI community building on and improving on the foundational efforts of XSEDE." [19].

XCI activities include creating a software repository of services and tools for the national research community to facilitate connecting resources, software, and services into the broader cyberinfrastructure ecosystem. Similarly XCI produces a suite of downloadable RPMS that include a wide variety of middleware and application tools called the XSEDE National Integration Toolkit (XNIT). This toolset makes it easier for local campus CI administrators to manage software that is compatible with Open Science Grid and XSEDE software tools. XSEDE's departure from the use of *campus bridging* as a term to use of terms that talk specifically about one area of the computational or computer science challenges that were identified by the campus bridging task force report but are more easily talked about and understood using other terms. We expect that in the future we will see more and more often researchers and technology adopters picking up terms relevant to one subset of what was included within the area called *campus bridging* in the 2011 ACCI task force report, but which is now more easily and appropriately referred to in some more specific way related to the topic of the yet outstanding computer and computational science problems.

8. Final Notes

The editors have included a list of acronyms and their definitions as Appendix 1. Appendix 2 contains a full listing of workshop participants. Both of the papers submitted in preparation for the workshop are included as part of Appendix 3, and the slides for each talk are presented in Appendix 4.

The major findings of the workshop were as follows:

1. One of the biggest challenges to the advance of campus bridging activities is the campus bridging name itself.
2. One of the great successes of XSEDE Campus Bridging has been its toolkits.
3. There lies a huge potential for expansion into the long tail of science, but this will necessitate a different approach from traditional HPC markets.
4. A facilitation-based approach to campus bridging is already closing pitfalls in the process of getting results, but more effort is required.
5. One of the most important challenges currently facing campus bridging practitioners is how to handle authentication.

9. Acknowledgments

The organizers of the workshop (Barbara Hallock, Yashema Mack, and Resa Reynolds) would like to thank the individuals who gave presentations or otherwise participated in the workshop:

10. References

1. Liming, L., Foster, I., and Tuecke, S. *Building Bridges from the Campus to XSEDE* In: Proceedings of IEEE Cluster 2015 (Chicago, IL, 8-11 Sept., 2015).

2. Fischer, J., Coulter, E., Knepper, R., Peck, C., and Stewart, C. *The XSEDE-compatible basic cluster – A tool for cluster implementation and management in research and training* In: Proceedings of IEEE Cluster 2015 (Chicago, IL, 8-11 Sept, 2015).

3. ACCI Task Force On Campus Bridging. National Science Foundation Advisory Committee for Cyberinfrastructure Task Force on Campus Bridging Final Report (2011). doi=http://hdl.handle.net/2022/13210.

4. Campus Bridging: Reducing Obstacles on the Path to Big Answers. https://pti.iu.edu/workshops/campusbridging-ieee/index.php.

5. S Stewart, C.A., S. Simms, B. Plale, M. Link, D. Hancock and G. Fox. What is Cyberinfrastructure? In: Proceedings of SIGUCCS 2010. (Norfolk, VA, 24-27 Oct, 2010). Available from: http://portal.acm.org/citation.cfm?doid=1878335.1878347.

6. XSEDE | Home. https://xsede.org/.

7. What we do. https://www.xsede.org/what-we-do/.

8. Training, Education, and Outreach Services. https://www.xsede.org/education-and-outreach/.

9. Rocks website. www.rocksclusters.org/.

10. XSEDE Campus Bridging. https://www.xsede.org/campus-bridging/.

11. The XSEDE National Interoperability Toolkit. http://cb-repo.iu.xsede.org/xsederepo/.

12. Simple Linux Utility for Resource Management. http://slurm.schedmd.com/

13. Globus. http://globus.org/.

14. Open Science Grid. http://opensciencegrid.org/.

15. GenesisII Wiki. http://genesis2.virginia.edu/wiki/.

16. InCommon: Security, Privacy, and Trust for the Research and Education Community. https://www.incommon.org/.

17. What is the XSEDE-Wide File System (XWFS)? https://portal.xsede.org/knowledge-base/-/kb/document/bdcp

18. Chris Jordan. 2012. Evaluation of parallel and distributed file system technologies for XSEDE. In *Proceedings of the 1st Conference of the Extreme Science and Engineering Discovery Environment:*

Bridging from the eXtreme to the campus and beyond (XSEDE '12). ACM, New York, NY, USA, , Article 9 , 1 pages. DOI=http://dx.doi.org/10.1145/2335755.2335799

19. XSEDE Community Infrastructure. https://confluence.xsede.org/display/XT/WBS+2.3.1+XSEDE+Community+Infrastructure+(XCI),+Office+of+the+Director

Appendix 1: Acronyms

Table 1. Acronyms and complete terms presented alphabetically

Acronym	Complete term
ACCI	NSF Advisory Committee for CyberInfrastructure
ACI-REF	Advanced CyberInfrastructure - Research and Education Facilitators
CI	Cyberinfrastructure
EMS	Execution Management Services
EOT	Education, Outreach, and Training
GFFS	The Global Federated File System
HPC	High-Performance Computing
HTC	High-Throughput Computing
IEEE	Institute of Electrical and Electronics Engineers
MSI	Minority-Serving Institution
NICS	The National Institute for Computational Sciences
NSF	The National Science Foundation
OSG	Open Science Grid
OSS	Open Source Software
PTI	The Indiana University Pervasive Technology Institute
RPM	RPM Package Manager (formerly RedHat Package Manager)
SLURM	Simple Linux Utility for Resource Management
SPs	Service Providers
TB	Terabytes
TEOS	XSEDE Training, Education, and Outreach Services
VO	Virtual Organization
XCBC	The XSEDE-Compatible Basic Cluster
XNIT	The XSEDE National Interoperability Toolkit
XSEDE	The eXtreme Science and Engineering Discovery Environment
YUM	Yellowdog Updater, Modified

Appendix 2: Participants

Table 2. Workshop organizers and participants

Name	Role	Institution	E-mail address
Barbara Hallock	Organizer	Indiana University	bahalloc@iu.edu
Yashema Mack	Organizer	NICS	ymack@utk.edu
Resa Reynolds	Organizer	Cornell University	rda1@cornell.edu
Dustin Atkins	Participant	Clemson University/ACI-REF	datkin2@clemson.edu
Jim Bottum	Participant	Clemson University/ACI-REF	jb@clemson.edu
Eric Coulter	Participant	Indiana University/XSEDE	jecoulte@iu.edu
Ian Foster	Participant	University of Chicago/Globus	foster@uchicago.edu
Robert Gardner	Participant	University of Chicago/Globus	rwg@uchicago.edu
Andrew Grimshaw	Participant	University of Virginia/XSEDE	grimshaw@virginia.edu
Barbara Hallock	Participant	Indiana University/XSEDE	bahalloc@iu.edu
Lee Liming	Participant	University of Chicago/Globus	lliming@uchicago.edu
Miron Livny	Participant	University of Wisconsin	miron@cs.wisc.edu
Rob Quick	Participant	Indiana University/OSG	rquick@iu.edu
Resa Reynolds	Participant	Cornell University/XSEDE	rda1@cornell.edu
Craig Stewart	Participant	Indiana University/XSEDE	stewart@iu.edu
Steve Tuecke	Participant	University of Chicago/Globus	tuecke@uchicago.edu

Appendix 3: Presentations and Papers

The following pages contain, in order of presentation, the slides from each of the presentations at the workshop followed by the full-text versions of both papers that were produced for this workshop.

Campus Bridging Through Facilitation: The ACI-REF Project

Campus Bridging: Reducing Obstacles on the Path to Big Answers

Jim Bottum

Principal Investigator – ACI-REF Project

CIO & Vice Provost – Clemson University

Presidential Fellow – Internet2

Background and Context

National – HPC as Demand Driver
- Labs, Centers, PACI, TeraGrid, XSEDE, OSG

Campus Computing Demand Growing in Parallel
- MRIs, CRIs, Start-Up Packages
- Condo and Co-lo Approaches
- Big Data Driving New Communities

University based research computing operations thin on people ESPECIALLY user-facing people

Result
- Training and education gap between resources and researchers – high barrier to entry without human assistance
  - …and the barriers become higher as we bring in new communities

Takeaway: We Need People!

Campus Bridging Takes People

The need for a new workforce – a new flavor of **mixed science and technology professional** – is emerging. These individuals have expertise in a particular domain science area, as well as considerable expertise in computer science and mathematics. Also needed in this interdisciplinary mix are professionals who are **trained to understand and address the human factors** dimensions of working across disciplines, cultures, and institutions using technology-mediated collaborative tools."

-2003 NSF Blue Ribbon Advisory Panel on Cyberinfrastructure

- Research enablement takes concentrated, committed effort by campuses and organizations to the right people.
- Typical research problems today are not technology or infrastructure problems – it's navigating the complexity, which takes people.
- Nationwide gap on campuses of these professionals, while demand grows.

An Approach

Answered Need: The ACI-REF Project (People)

**Goal:** Seed investments in user-facing people – **facilitators** – at campuses to:

- Assist researchers in taking advantage of advanced computing resource investments, especially at the local campus level; and

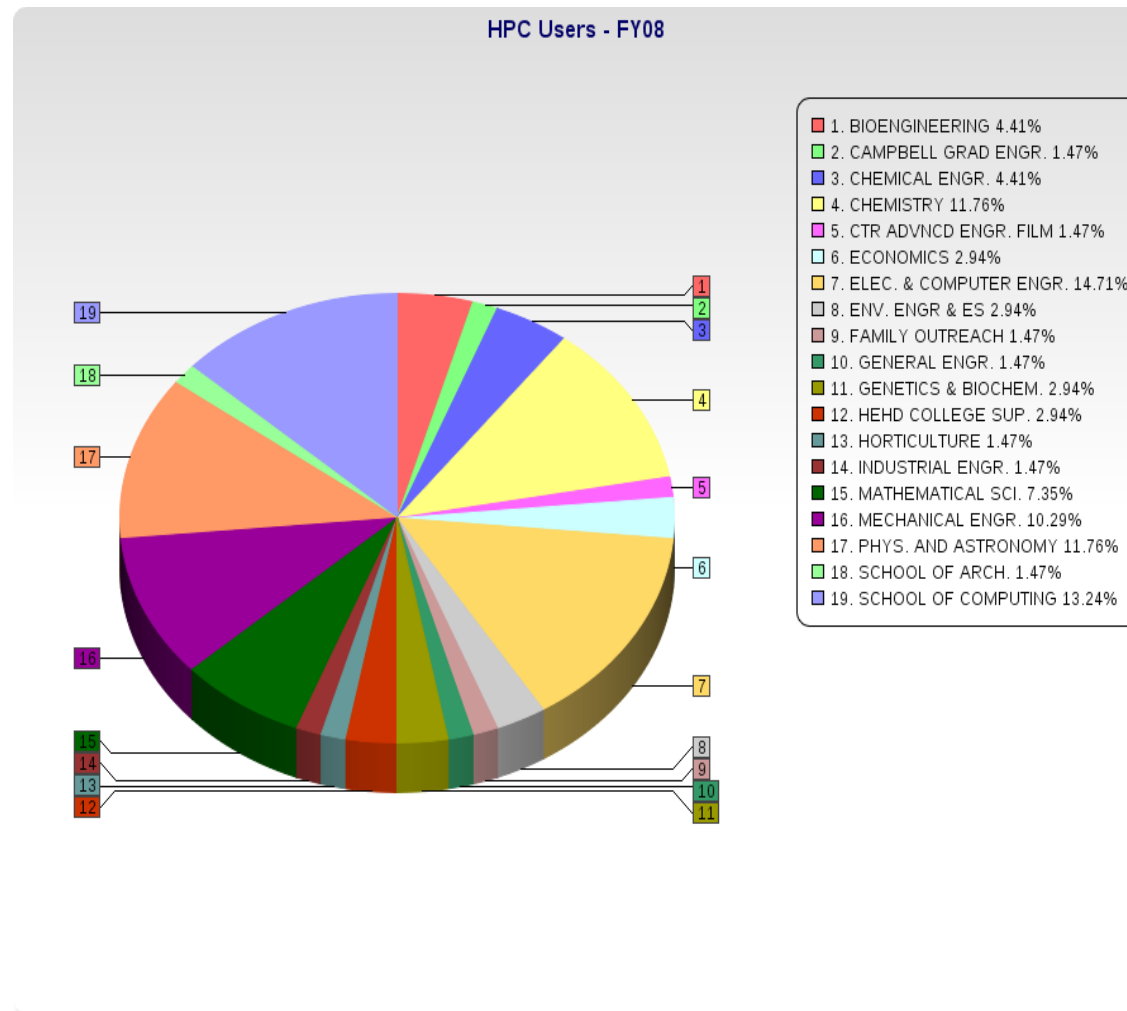- Build inter-institutional collaborative networks of knowledge to share expertise across campuses.

ACI-REF Project Facilitators

ACI Resources

Researchers

One Campus' Experience

Clemson

May 2010 – first Clemson "facilitator" funded

Departments with Faculty Trained on Palmetto 54 Clemson Academic Departments

May 2010: NSF Outreach & Infrastructure Improvement Grant Funded
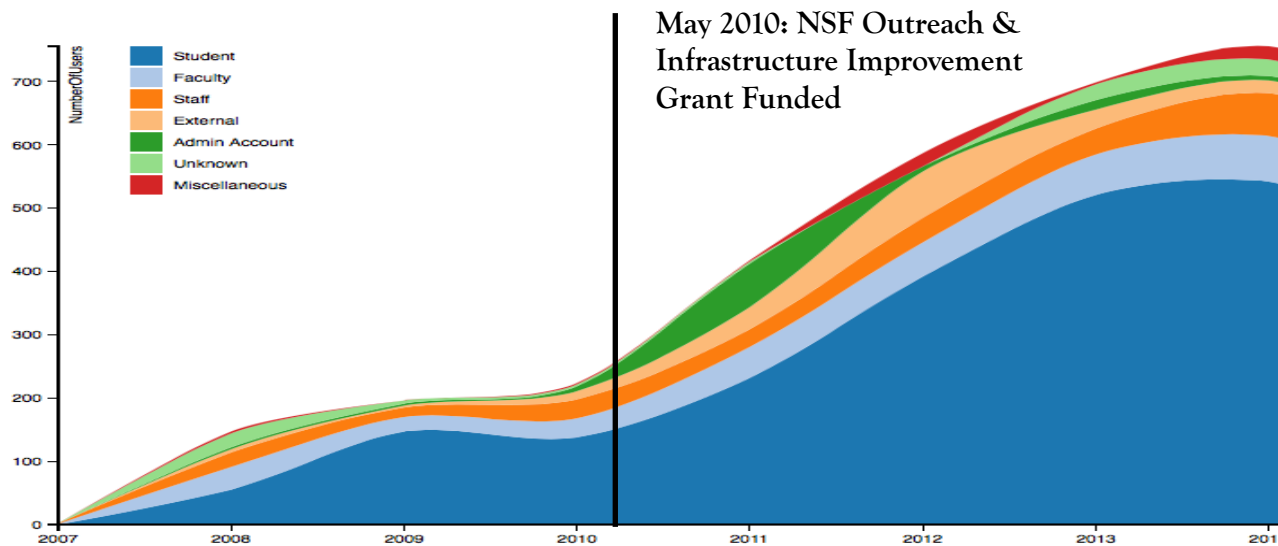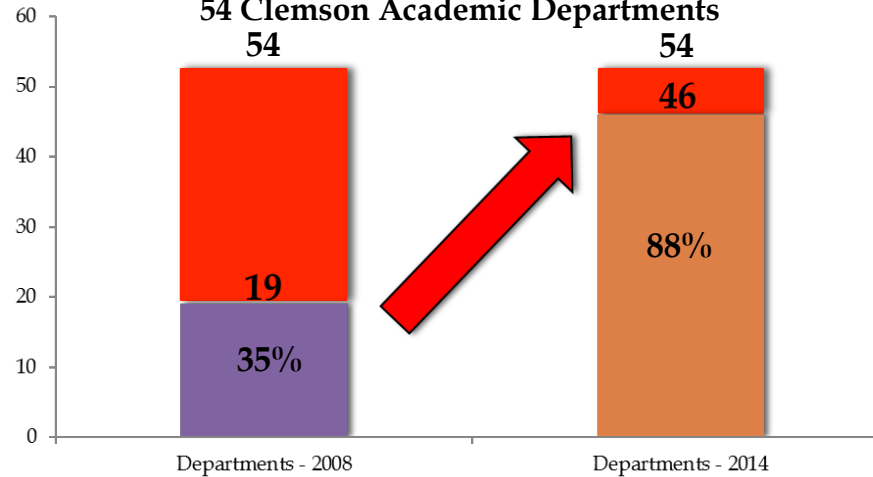
Campus Community Growth

2007/2008 – The Usual Suspects

Campus Community Growth

2014 – Non-Traditional Communities

1. ADVANCED PLANT TECHNOLOGY 0.13%
2. AGRICULTURAL RESEARCH 0.06%
3. ART 0.06%
4. AUTOMOTIVE ENGINEERING 0.75%
5. BBS NONDEGREE 0.06%
6. BIOENGINEERING 2.32%
7. BIOLOGICAL SCIENCES 1.72%
8. CAMPBELL GRAD ENGR PROGRAM 0.19%
9. Center for Optical Materials Science and Engineering Technologies 0.06%
10. CHARLES H. HOUSTON CENTER 0.13%
11. CHEMICAL ENGINEERING 3.69%
12. CHEMISTRY 3.07%
13. CIVIL ENGINEERING 4.76%
14. CTR ADVNCD ENGR FIBER FILMS 0.13%
15. CU GENOMICS & COMP BIO LAB 0.25%
16. CU GENOMICS INSTITUTE 0.38%
17. ECONOMICS 3.69%
18. ELEC. & COMPUTER ENGR. 16.03%
19. ENVIRONMENTAL ENGR & EARTH SCI 2.76%
20. EXPERIENTIAL EDUCATION 0.06%
21. FAMILY OUTREACH 0.06%
22. FOOD, NUTRITION, AND PACKAGING 0.06%
23. GENERAL ENGINEERING 0.94%
24. GENETICS & BIOCHEMISTRY 1.63%
25. GRAPHIC COMMUNICATIONS 0.06%
26. HEHD COLLEGE SUPPORT SERVICES 0.13%
27. HORTICULTURE 0.13%
28. HUMAN RESOURCES 0.06%
29. INDUSTRIAL ENGINEERING 1.94%
30. INST BIOLOGICAL INTERFACES ENG 0.06%
31. INTERNATIONAL VISITORS 0.69%
32. MANAGEMENT 0.25%
33. MARKETING 0.06%
34. MATERIALS SCIENCE & ENGRG 1.00%
35. MATHEMATICAL SCIENCES 5.95%
36. MBA PROGRAM 0.19%
37. MECHANICAL ENGINEERING 8.95%
38. NATIONAL INSTITUTE FOR PARKS 0.06%
39. PARKS,REC & TOURISM MGT 0.13%
40. PHYSICS AND ASTRONOMY 6.07%
41. PLANNING, DEV. & PRESERVATION 0.13%
42. POLITICAL SCIENCES 0.06%
43. PRE-BUSINESS 0.06%
44. PSYCHOLOGY 0.25%
45. PUBLIC HEALTH SCIENCES 0.13%
46. RESEARCH SAFETY 0.06%
47. SCHOOL OF ACCOUNTANCY&FINANCE 0.13%
48. SCHOOL OF AG, FOR, ENV SCIENCE 1.06%
49. SCHOOL OF ARCHITECTURE 0.06%
50. SCHOOL OF COMPUTING 28.24%
51. SCHOOL OF NURSING 0.65%
52. SOCIOLOGY AND ANTHROPOLOGY 0.69%
53. SPECIAL STUDENT 0.19%
54. MAINTENANCE & OPERATIONS 0.06%

Non-Traditional Impacts

Case Study: Hadoop In Action

Kevin McKenzie, Clemson Chief Information Security Officer, and his team used the Clemson Hadoop platform as part of a recent security incident response. His team needed to evaluate multi-year volumes of log data from the Clemson network to validate the extent of an incident they were investigating. The team loaded log data into the Hadoop cluster to gain a higher performance of log analysis.

Using Hadoop proved very beneficial, as it eliminated the estimated weeks (if not months) to accomplish on current local systems and the analysis was completed in less than a couple of hours, allowing the team to more quickly determine the extent of the issue.

Departments Receiving Hadoop Training

Bioengineering
Chemical Engineering
Chemistry
Civil Engineering
Economics
Elec. & Computer Engr.
Environmental Engr & Earth Sci
Experiential Education
General Engineering
Genetics & Biochemistry
Industrial Engineering
International Programs
Law Enforcement & Safety

Management
Mathematical Sciences
Mechanical Engineering
Medicaid IT Services
Physics And Astronomy
Public Health Sciences
Research Safety
School of Ag. For. Env Science
School of Computing
Univ Facilities Support Svcs
VP Finance & Operations
Information Security & Privacy

ACI-REF Formation

- Award for NSF-sponsored workshops held in 2012 helped define the needs of the broader community

- Goal: **Advance our nation's research & scholarly achievements through the transformation of campus computational capabilities and enhanced coupling to the national infrastructure.**
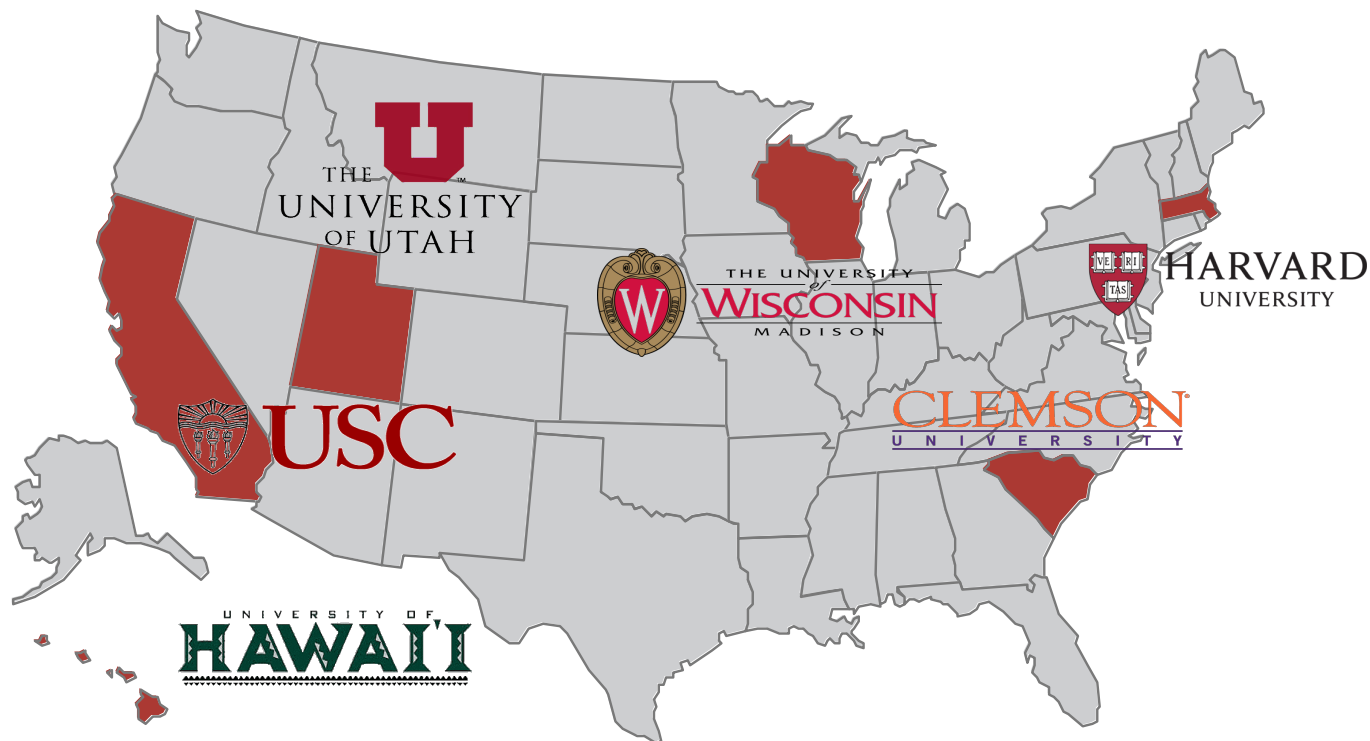
NSF-Funded Project – ACI-REF

$5.3M NSF Award supports the project leadership team and 2 Facilitators for each of the 6 partner sites for 2 years.

PI: Jim Bottum, **Clemson**

Project Leadership:
- James Cuff, **Harvard** (PI Chair)
- Maureen Dougherty, **USC**
- Gwen Jacobs, **Hawaii**
- Paul Wilson, **Wisconsin**
- Tom Cheatham, **Utah**
- Barr von Oehsen, **Clemson**

Facilitator Lead: Bob Freeman, **Harvard**

Chief Scientist: Miron Livny, **Wisconsin**

Progress & Accomplishments

March 2015 – Early Results

- Participating campuses saw:
  - 9% ↑ Growth in Departments Served (157 to 171 across campuses)
  - 15% ↑ Growth in Number of Advanced Computing Users (1,462 to 1,674 across campuses)
  - 800+ Individual Consultations with ACI-REFs
  - 1000+ Training Attendees in Sessions by ACI-REFs
  - 74 Training Sessions Given by ACI-REFs
  - Breadth of support increased through expertise sharing
    - Facilitators participating in XSEDE, CloudLab, OSG, Software Carpentry training sessions to further their skills
  - Development of **replicable best practices**
    - Training, office hours, cross-institutional knowledge base
  - Anecdotal: "Love letters" from faculty and researchers
  - Facilitators are functioning as a distributed group

Progress – 2015

August 2015 – ACI-REF Project 1.5 Year Mark

- Participating campuses saw:
  - Estimated **17%** Growth in Departments Served
  - Estimated **16%** Growth Number of Advanced Computing Users
  - **821** Individual Consultations with ACI-REFs **since March 2015**
  - **1288** Training Attendees in Sessions by ACI-REFs **since March 2015**
  - **75** Training Sessions Given by ACI-REFs **since March 2015**
  - **317M+** Core Hours Delivered by ACI-REF Campuses **since March 2015**
  - At least **65** Non-Traditional Departments Using ACI Across Campuses
- Other Notables:
  - **Multi-site Support Network** – for example, ACI-REF at USC able to tap into Clemson GIS ACI-REF to help a faculty member with their research
  - **Best Practices Manual** – ACI-REFs have developed a "best practices for facilitation" manual to aid in the onboarding of new facilitators and to formalize the practice
  - **Office Hours** – most ACI-REF campuses are now holding set advertised hours each week where researchers can stop by and get help with any problems or ask questions

The Value of Facilitation at Scale

- Individual campuses can (and are) investing in facilitators within their IT organizations.
  - Some campuses are writing ACI-REF into their campus CI plans as an aspirational goal to "join"
- What's the value of creating a community of facilitators across the nation?
  - A single facilitator cannot be a 'jack of all trades' lest they turn into a master of none.
  - Deploying facilitators at a national-scale ensures a community that shares in distributed, specialized expertise across the nation.
- Original proposal was called "*Condo of Condos*" – because the community (people-sharing) model is much like that of condo computing:
  - Campuses 'buy in' at the 1-facilitator level, but can leverage a community of facilitators to serve their researchers' needs

Example Successes – Clemson & OSG

The Systems Genetics Lab at Clemson University (F. Alex Feltus, PI; Will Poehlman, PhD student) requires high performance computing (HPC) to build and interpret biomolecule interaction networks (node-edge graphs) to discover gene sets underlying complex traits in plants and animals. Dr. Feltus regularly interacts with Clemson ACI-REFs on his research needs.

"Experiment completion time is highly queue dependent of course, but as an example, we recently broke a single experiment into 12,000 jobs (1GByte RAM each) and launched on OSG which took 19.5 hours to complete. In contrast, the same experiment would have taken 14-21 days to complete on the Palmetto cluster, which is also part of OSG, given the PI's resource allocations."
 – **F. Alex Feltus, Clemson University**

Clemson contributed **over 300K core hours** last month to the OSG pool – in line with the spirit of sharing in the ACI-REF project.

Example Successes – Utah

High-Energy Theoretical Physics – Chris Kelso, University of Utah

"I work in high energy theoretical particle physics. Specifically, I investigate physics beyond the Standard Model with a focus on dark matter implications. My research often requires scans of models that have very large numbers of parameters. This work could not be completed without the computing resources provided at CHPC. Almost as valuable as the use of the CHPC machines was the extremely helpful assistance I received from **Wim R. Cardoen**. *Many of the codes I often use are serial, open source code that has been developed by many physics experts. To try and convert these codes to parallel would be a monumental task. Wim worked very hard to help me to find a solution that allowed this serial code to still utilize the numerous processors available on the CHPC machines. Without this, my projects would take months to finish, rather than a few days."*

– Chris Kelso, University of Utah PostDoc, on Utah ACI-REF Wim Cardoen

Example Successes – Harvard

HPC Assistance in Biology Software and Workflow – Zack Lewis, Harvard University

"I am a sixth year graduate student in the Department of Organismic and Evolutionary Biology. I started a transcriptomics project with little experience in coding and no experience in high powered computing (HPC). Without **Bob Freeman's** *work through ACI-REF I do not think I would have been able to complete my bioinformatics project. I was not aware of ACI-REF at the time I started my HPC bioinformatics work. To my good fortune I happened to connect with Bob Freeman at the weekly Research Computing office hours. Bob has accompanied me nearly every step of the way along my 6 month journey into HPC. Bob's help has taken the form of instruction on coding, monitoring active jobs, writing and adapting scripts for my project, as well as connecting me with researchers working on similar problems or at similar stages in learning transcriptomics. In particular, building connections with other researchers at Harvard through ACI-REF has been one of the most useful experiences. I now often work through my HPC issues with graduate student and postdoc peers that I have connected with through Bob."* **– Zack Lewis, Harvard University PhD Candidate, on Harvard ACI-REF Bob Freeman**
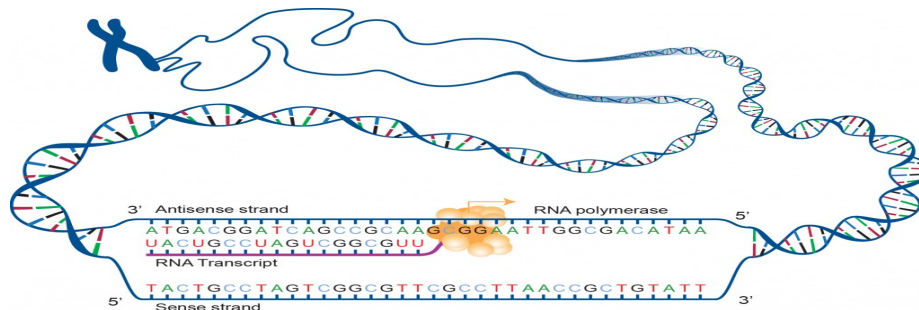
Project Status & Future Directions

Phase I Award
- Successful Year 1
- Second Year
  - Evaluation in progress, planning for Phase II award extension

ACI-REF Consortium – Demand-Driven
- Many requests to join from campuses across the nation
- Also being written into campus CI plans
- First ACI-REF Summer School held @ OU in 2015
- Developing mechanism for adding partners committed to a community that:
  - Values facilitation and outreach/engagement
  - Focuses on people helping people and collaboration
  - Recognizes we are limited in what we can do and who we can help as a single campus
  - Sustainability – creation & adoption of a new career path for facilitators

Significant Partnerships
- **Open Science Grid** – technical integration project commissioned between OSG & ACI-REF at Spring 2015 OSG Executive Committee Meeting
- **XSEDE** – ongoing discussions with XSEDE Campus Bridging team
- **CloudLab** – ACI-REFs attended 'train the trainer' workshops to help researchers take advantage of CloudLab resources

QUESTIONS?

The Campus HTC Queue or Give me a condor's quill!

Rob Quick
Indiana University
Research Technologies - High Throughput Computing
Open Science Grid - Operations and Communication Officer

What can we provide with local resources and local effort defined by local stakeholders that allow them to access local, national, and international computational resources?

- Defined by local needs

- Little to no dependency on other infrastructure services

- Allows for experimentation with new technology

- Does not rediscover the wheel - though a new tread pattern may be needed

RESEARCH TECHNOLOGIES

INDIANA UNIVERSITY

University Information Technology Services
Pervasive Technology Institute

"Give me a condor's quilll! Give me Vesuvious crater for an inkstand!" - Herman Melville

What is our Leviathan?

What are our Leviathans?

* Indiana University - Big Red II, Karst, Mason

* Open Science Grid - 127 Clusters on 5 Continents

* XSEDE - 8 High End HPC Clusters

* Other compute resources (Cloud, International HTC Providers, Other Technologies/Systems)

IU HTC Cluster - Karst

- 4096 Cores

- 32 GB RAM/Node

- PBS/Torque

- SSH connection from submission point
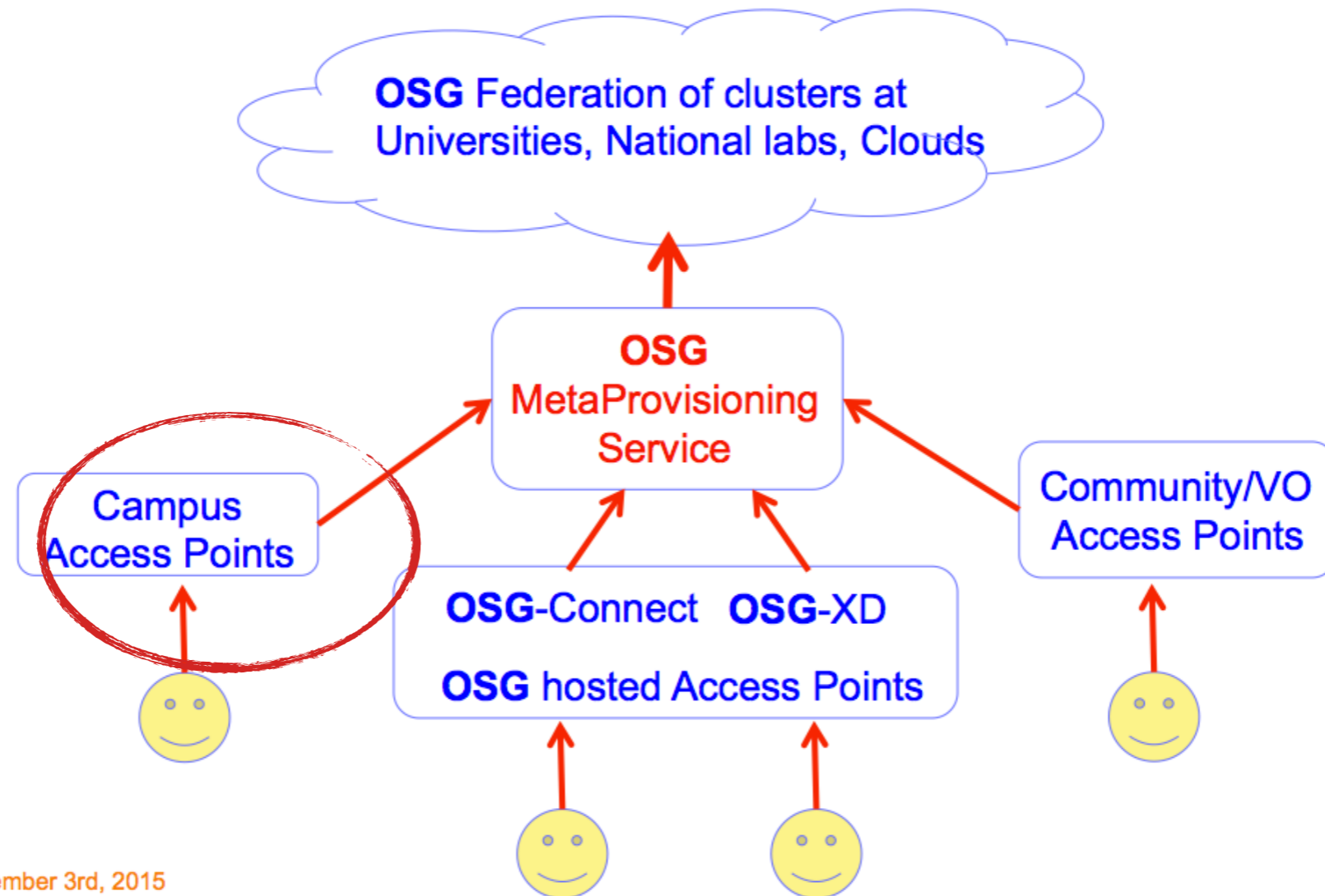
Open Science Grid

Slide -FKW

Duplicate already existing xd-login mechanism.
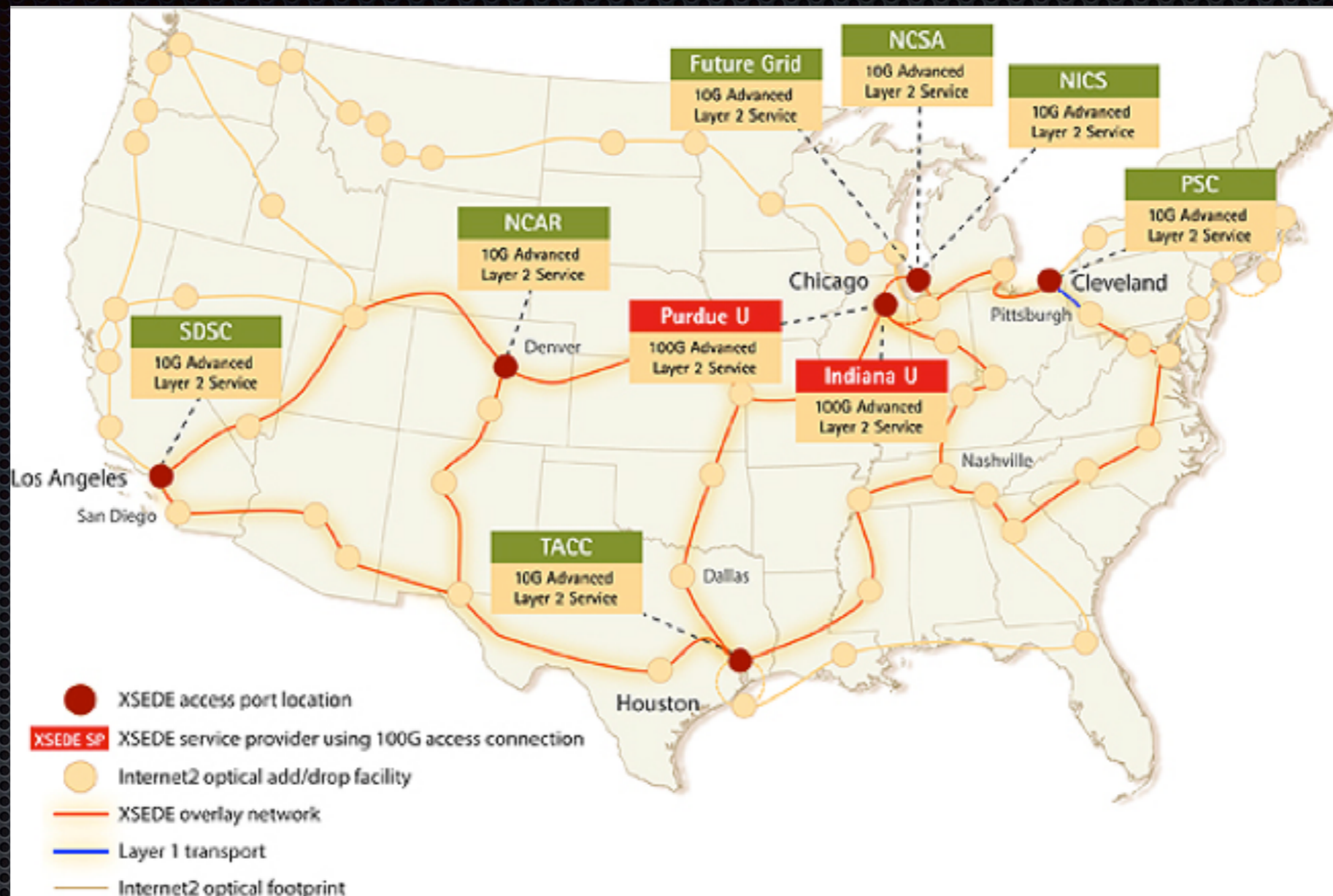
XSEDE Execution Management Service

Uncertain after viewing
https://youtu.be/6geHQxwpQUY

Our harpoon**s** already exists!

* HTCondor

* BOSCO - BLAH Over SSH Condor Overlay

  * BLAH - Batch Local Ascii Helper

  * Supports LSF, PBS/Torque, SGE, HTCondor and SLURM

* XSEDE EMS

* Services and knowledge provided OSG and OSG-Connect

However they need sharpening

* Prioritization rules

* Flocking to the OSG Virtual Organization

* Connection to Execution Management Service (XSEDE)

* Extension to other computing resources

What we have so far…

* Hardware for the submission node

* Committed funding and effort for initial IU instance

* Authentication scheme - IU Credentials

* Contact points for Karst, OSG-XD, and EMS

* Test cases of SPLInter and Galaxy-BLAST

Our expected timeline…

- June - Procure and Install Submission Hardware

- September - Install HTCondor/BOSCO and other necessary components

- October - Initial testing of SPLInter and Galaxy-BLAST submission

- November - Transition SPLInter to new service

- January - Experiment with EMS and/or other alternatives to connect to XSEDE resource providers

- March - Determine future funding model and partner institutes

Some added benefits…

- More horsepower for SPLInter and Galaxy-BLAST

    - BLAST needs some amount of processing power to recompile results

- More horsepower available for OSG xd-login node

    - More pressure applied to the OSG Factory leading to more cycles averrable to OSG

- A test space for other IU Researchers who may be considering an HTC solution (local or national)

    - IU Network Science Institute, extension of SPLInter, other Galaxy based applications

- Connections - Local researchers to national/international resources and nationally funded providers connection to local projects

"Queequeg was a native of Kovoko, an island far away to the West and South. It is not down in any map; true places never are."

Herman Melville

Disclaimer: No whales were harmed in the creation of this slide deck.

FREE MOBY

powermobydick.com

XCBC and XNIT: tools for cluster implementation and management in research and training

Jeremy Fischer – jeremy@iu.edu

Rich Knepper – rknepper@iu.edu

Eric Coulter – jecoulte@iu.edu

Charles Peck – charliep@cs.earlham.edu

Craig Stewart – stewart@iu.edu

XSEDE

Extreme Science and Engineering
Discovery Environment

Overview

- Big Picture – Campus Bridging
- XCBC – XSEDE Compatible Basic Cluster
  - LittleFe example XCBC
- XNIT – XSEDE National Integration Toolkit
  - Limulus HPC200 example XNIT
- Success Stories

Campus Bridging Goals

The goal of campus bridging in general is to create a sense of "virtual proximity." Any resource should feel as if it's just a peripheral to your laptop or workstation.

The goal is to make it convenient and intuitive to simultaneously use your personal computing systems, departmental and campus systems (at your campus and others), and national resources like XSEDE . . . all (almost) transparently and easily.

Campus Bridging Initiatives

- Setting up campus resources is a challenge:
  - Disseminate best practices from XSEDE and Service Provider Centers

- Getting started on XSEDE resources is tough:
  - Provide campuses with an environment that is like what their users will see when using XSEDE
  - Make it easier to move jobs and data in between campuses, centers, XSEDE, etc…

The XSEDE-Compatible Basic Cluster – XC*BC

- What is the XSEDE Compatible Basic Cluster?

- Why is the XCBC project important to Campus Bridging?

The XSEDE-Compatible Basic Cluster – XC*BC

- The LittleFe based cluster
  - Created to teach parallel computing techniques
  - Bootable Cluster CD (BCCD) part of this project
  - Over 60 LittleFe units built in workshops
- Practical applications
  - Real world deployments
  - Teaching clusters
  - Piloting projects

The XSEDE-Compatible Basic Cluster – XC*BC

- Looking at the LittleFe

- Differences in LittleFe projects design versus XCBC adaptation

- Why we deviated from the LittleFe design
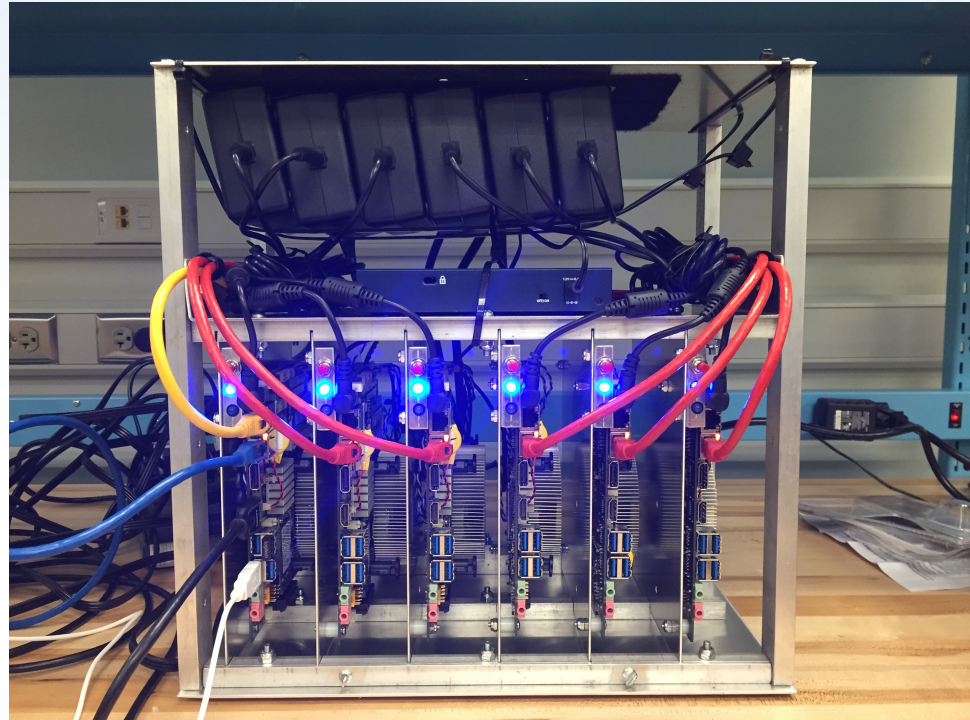
The XSEDE-Compatible Basic Cluster – XC*BC

Rocks Minimum Requirements

Frontend Node

- Disk Capacity: 30 GB
- Memory Capacity: 1 GB
- Ethernet: 2 physical ports (e.g., "eth0" and "eth1")
- BIOS Boot Order: CD, Hard Disk

Compute Node

- Disk Capacity: 30 GB
- Memory Capacity: 1 GB
- Ethernet: 1 physical port (e.g., "eth0")
- BIOS Boot Order: CD, PXE (Network Boot), Hard Disk

The XSEDE-Compatible Basic Cluster – XC*BC

The XSEDE-Compatible Basic Cluster – XC*BC

What's next?

What comes in the "box"?

- Torque/Maui scheduling system

- OpenMPI parallel libraries

- GCC compiler

- Well-rounded selection of scientific software

What do I need to do?

- Set up scheduling!

- GPU nodes?

- Set up Globus?

- Connect to a Grid?

- Ensure custom codes make use of the include OpenMPI and other libraries

- Add users

- Try out some jobs!

The XSEDE-Compatible Basic Cluster – XC*BC

- Future possibilities for XC*BC

  - Kickstart (assuming staying RPM-based) into Werewulf

  - Xcat

  - Slurm

  - Ansible management of Rocks

  - Default Science Gatways

  - Everything as modules

The XSEDE-Compatible Basic Cluster – XC*BC

Additional resources:

- LittleFe site - https://littlefe.net/

- Bootable Cluster CD Project - http://www.bccd.net/

- Rocks Clusters site – http://www.rocksclusters.org/

- XSEDE Campus Bridging – https://www.xsede.org/campus-bridging

The XSEDE National Integration Toolkit – XN*IT

XN*IT

The XSEDE National Integration Toolkit – XN*IT

- Why XNIT?
- XN*IT vs XC*BC

The XSEDE National Integration Toolkit – XN*IT

- Taking an existing campus cluster and enabling the XNIT lets you have an XSEDE-like resource

- All you really need is rpm compatibility, the desire for scientific software packages, and the administration knowledge you've already been using for your cluster...

The XSEDE National Integration Toolkit – XN*IT

- Demo case: The Limulus HPC200
  - Why is it here? What does it do?
  - Can't you just do the same thing with the LittleFe? Or even a stack of old PCs?

The XSEDE National Integration Toolkit – XN*IT

- DVD and SSD
- RAID Array
- Front Panel (PWR, USB, Video)
- Motherboard Stack Behind Cooling Fans

- The HPC200 features more in depth…

The XSEDE National Integration Toolkit – XN*IT

Main Motherboard

HDD Drive Cage

Front Panel

Ethernet Switch and Power Relays

Main Power Supply

Motherboard Stack
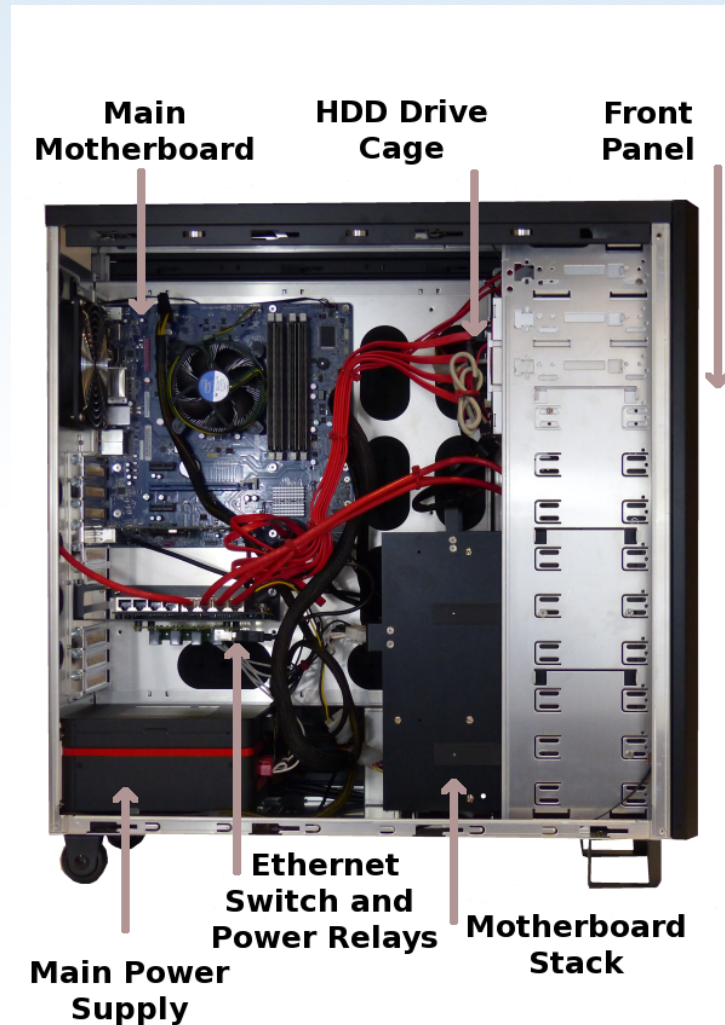
- The HPC200 features more in depth…

The XSEDE National Integration Toolkit – XN*IT

- A campus cluster may already have a broad selection of software…

- …but making it XSEDE-like really is as easy as setting up the Yum repository of the XNIT

- Lots of the pre-installed software is in environment modules. Future XNIT packages will be available as modules, too.

Comparing the LittleFe and HPC200

Rmax and Rpeak

- LittleFe:
  - Cost: $3600
  - 2.8ghz * 2 cores * 16 ins/cycle * 1 cpu/node = 89.6 GFLOPs per node * 6 Nodes = 537.6 GFLOPS Rpeak
  - Actual measured with HPL = 403.2 GFLOPS
  - Efficiency = 403.2/537.6 = 75%

Comparing the LittleFe and HPC200

Rmax and Rpeak

- HPC200:
  - Cost: $5995
  - 3.1ghz * 4 cores * 16 ins/cycle * 1 cpu/node = 198.4 GFLOPs per node * 4 Nodes = 793.6 GFLOPS Rpeak
  - Actual measured with HPL = 658.8 GFLOPS
  - Efficiency = 658.8/793.6 = 83%

Success Stories

- XCBC Installations
  – Howard University
  – Michigan State University
  – Marshall University
  – Southern Illinois University (upcoming)
- XNIT Installations
  – Hawaii University
  – Montana State University

Our reach will forever

exceed our grasp, but,

in stretching our horizon,

we forever improve our world.

XSEDE

Extreme Science and Engineering
Discovery Environment
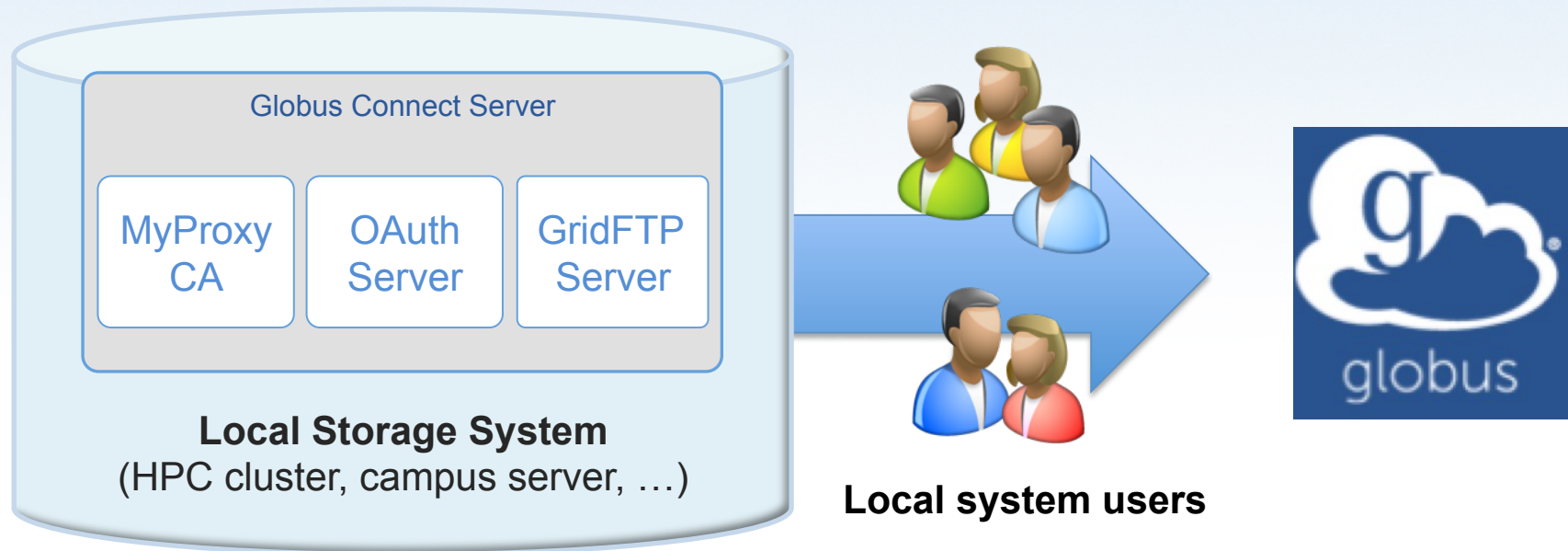
Data Transfer for both XN*IT & XC*BC systems

Data Transfer for both XN*IT & XC*BC

Transfer data easily, quickly and reliably to other locations…

Research Computing HPC Cluster

Lab Server

Campus Home Filesystem

Personal Laptop

Desktop Workstation

XSEDE Resource

Public Cloud
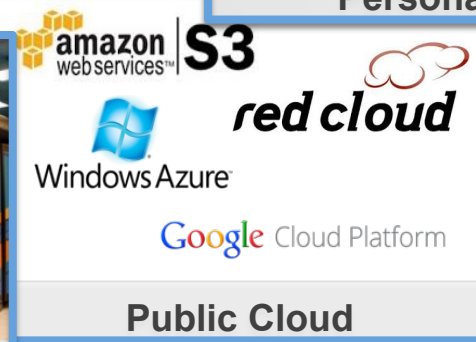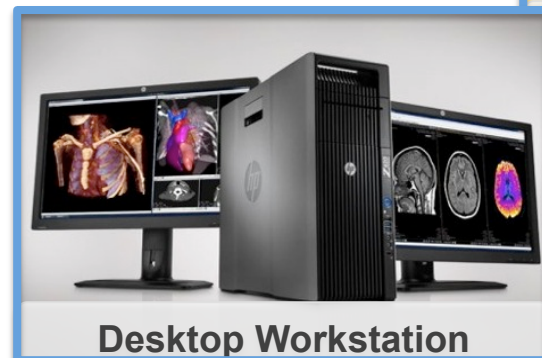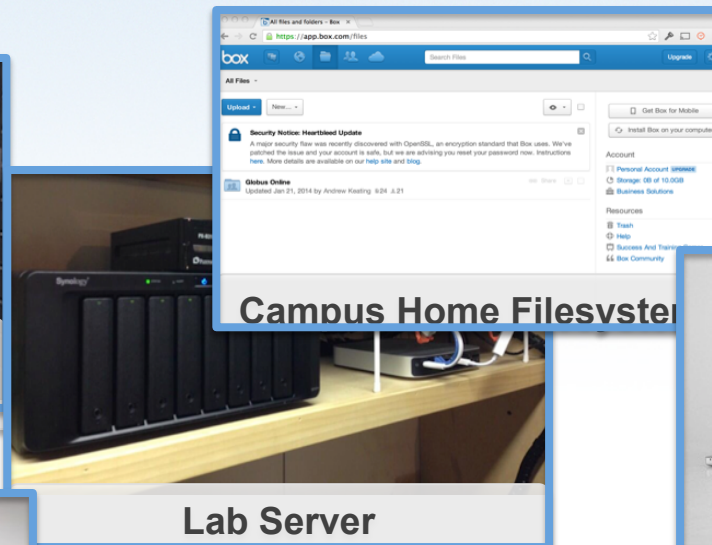
Data Transfer for both XN*IT & XC*BC

Enable your storage system with Globus Connect Server

Globus Connect Server

| MyProxy CA | OAuth Server | GridFTP Server |

Local Storage System
(HPC cluster, campus server, …)

Local system users

globus

- Create endpoint in minutes; no complex software install
- All users with local accounts can transfer and share files
- Easily integrates with campus security systems
- Delivered as native Linux packages: RPMs and DEBs

XSEDE

Typical deployment

Science

DMZ

+

Globus

Border Router

perfS●NAR

Enterprise Border Router/Firewall

WAN

10G

10GE

10GE

perfS●NAR

Clean, High-bandwidth WAN path

Site / Campus access to Science DMZ resources

10GE

Site / Campus LAN

Science DMZ Switch/Router

10GE

Per-service security policy control points

perfS●NAR

High performance Data Transfer Node with high-speed storage

XSEDE

Globus Web Transfers

After

Before

Data Transfer for both XN*IT & XC*BC

- Signup: globus.org/signup
- Enable your resource: globus.org/globus-connect-server
- Need help? support.globus.org
- Subscribe to help make Globus self-sustaining
  globus.org/provider-plans
- Follow Globus: @globusonline

Building bridges from the campus to XSEDE

Ian Foster, Lee Liming, Steve Tuecke

foster@uchicago.edu

Argonne
NATIONAL LABORATORY

XSEDE campus bridging use cases (http://hdl.handle.net/2142/43882)

1) XSEDE systems and services should allow **use of campus identity credentials** by supporting federated identity and authorization mechanisms

2) XSEDE should help campus system administrators **create XSEDE-like environments on campus systems** to smooth their users' transitions between campus and national systems

3) XSEDE services should **provide "remote desktop" access** so that users on campuses can remotely view graphical displays generated by XSEDE systems

4) XSEDE should enable users on campuses to **conduct integrated data analysis** that includes data on both campus and XSEDE systems

5) XSEDE should enable users on campuses to **initiate automated workflows** (scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services

XSEDE campus bridging use cases (http://hdl.handle.net/2142/43882)

1) XSEDE systems and services should allow **use of campus identity credentials** by supporting federated identity and authorization mechanisms

2) XSEDE should help campus system administrators create XSEDE-like environments on campus systems to smooth their users' transitions between campus and national systems

3) XSEDE services should provide "remote desktop" access that users on campuses can remotely view graphical display generated by XSEDE systems

4) XSEDE should enable users on campuses to **conduct integrated data analysis** that includes data on both campus and XSEDE systems

5) XSEDE should enable users on campuses to **initiate automated workflows** (scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services

XSEDE campus bridging use cases (http://hdl.handle.net/2142/43882)

1) XSEDE systems and services should allow **use of campus identity credentials** by supporting federated identity and authorization mechanisms

2) XSEDE should help campus system administrators **create XSEDE-like environments on campus systems** to smooth their users' transitions between campus and national systems

3) XSEDE services should **provide "remote desktop" access** that users on campuses can remotely view graphical displays generated by XSEDE systems

4) XSEDE should enable users on campuses to **conduct integrated data analysis** that includes data on both campus and XSEDE systems

5) XSEDE should enable users on campuses to **initiate automated workflows** (scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services

XSEDE today

- Many resources with different credential and authorization requirements, e.g.:
    - Different service providers (SPs)
    - XSEDE User Portal
    - Globus services
    - Web services
- Service providers support single sign-on via X.509 credentials (with MyProxy)
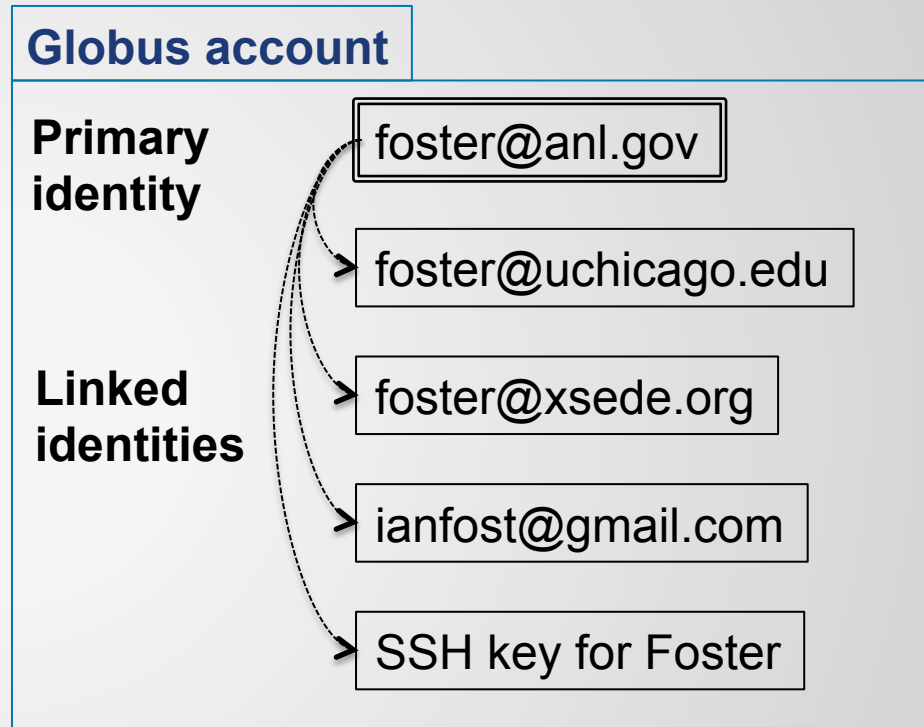- Multiple group management systems

"Use campus identity credentials to access XSEDE services"

- User can use campus credential (e.g., "foster@uchicago.edu", "foster@anl.gov") to authenticate to XSEDE User Portal (XUP)
- Authenticated user can then access various XSEDE services without further authentication
  - E.g., HPC resources, XRAS allocation service, Globus services, …
  - May require authorization by providers, and additional verification of identity
- XSEDE can translate token obtained at XUP login into other credentials understood by other service interfaces. E.g.:
  - Short-term X.509 for some legacy interfaces
  - WS-Trust STS to create signed SAML chain for Web services interfaces
- Users and service providers can define groups and use those groups for authorization
- Application and service developers can access identity services via CLI, GUI, or API

# The Globus Auth solution (1) Linked identities

- Build on **Globus Auth** cloud-based identity management hub
- **Supported identity providers** include:
  - GlobusID
  - OpenID Connect
  - Google
  - CILogon (X.509)
  - SAML
  - Email addresses
  - SSH Public Keys

Globus account

Primary identity	foster@anl.gov
	foster@uchicago.edu
Linked identities	foster@xsede.org
	ianfost@gmail.com
	SSH key for Foster

+ Ability to associate attributes with identities
+ Cached access tokens
+ APIs for management: e.g., Identity provider adds new identity by "POST /v2/identities"

The Globus Auth solution (2) OAuth2 access
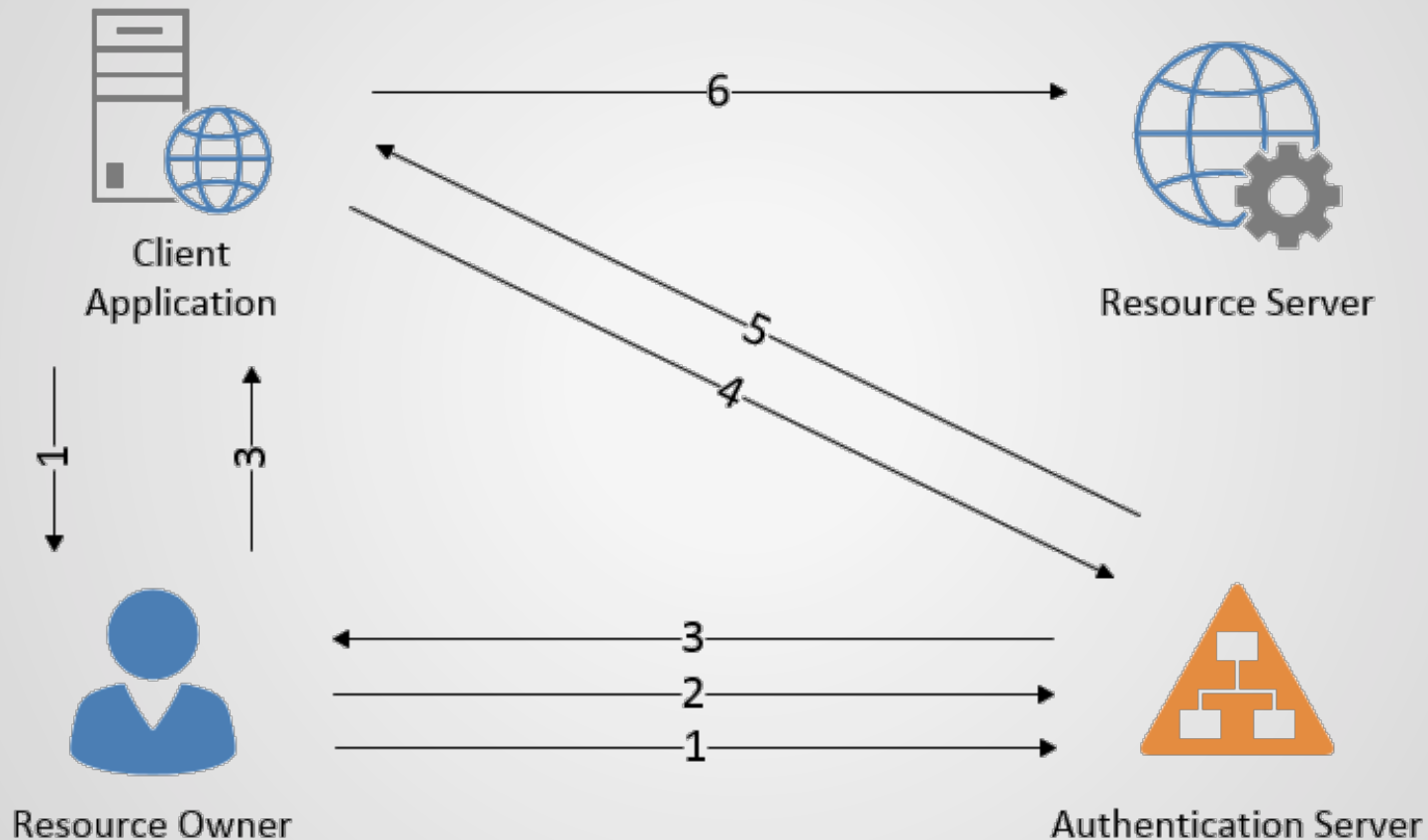
Key point: The resource owner never discloses credentials to the client application

API access to Globus Auth capabilities

- OAuth2 API
  - Authorization Code Grant
  - Implicit Grant
  - Resource Owner Password Credentials Grant
  - Resource Owner SSH Credentials Extension Grant
- Identities
  - GET /v2/identities: Retrieve info about identity(s)
  - PUT /v2/identities: Update info about identity
  - POST /v2/identities: (IdP) Provision identity in Globus Auth
- Identity providers
  - GET /v2/identity_providers: Retrieve info about identity provider

Globus Auth capabilities can be used by many services

For example:

- Globus transfer and sharing

- Globus groups

- Globus publish

- GENESIS-II

- CI Connect

- DOE kBase

- NCAR RDA

Many of which have their own API, e.g., Globus transfer

- tasksummary

- task

- task_list

- endpoint

- endpoint_list

- etc.

Transfer Files | Activity | Manage Endpoints | Dashboard | Console

Transfer Files

Get Globus Connect Personal
Turn your computer into an endpoint.

Endpoint `xsede#comet` ... Go   ◀ ▶   Endpoint `ucrcc#midway` ... Go
Path   Go   Path   Go

Please authenticate to access this endpoint

When you click CONTINUE you will be redirected to the endpoint's login webpage (you will be returned here once you've authenticated).

[Continue]   [Cancel]

Please authenticate to access this endpoint

When you click CONTINUE you will be redirected to the endpoint's login webpage (you will be returned here once you've authenticated).

[Continue]   [Cancel]

▶ more options   Label This Transfer

This will be displayed in your transfer activity.

XSEDE

**Extreme Science and Engineering
Discovery Environment**

Welcome to the XSEDE's Client Authorization Page

Science Gateway Access

The XSEDE Science Gateway or Service below is requesting access to your XSEDE account. If you approve, please sign in w

Note: Only members of active XSEDE project allocations will be able to sign in on this page.

SCIENCE GATEWAY INFORMATION	SIGN IN
The XSEDE Science Gateway listed below is requesting access to your XSEDE account. If you approve, please sign in. *Name:* Globus *URL:* http://www.globus.org/	Username [ ] Password [ ] SIGN IN CANCEL

Please send any questions or comments about this site to help@xsede.org.

"Globus" requests that you select an Identity Provider and click "Log On". If you do not approve this request, do not proceed.

By proceeding you agree to share your name and email address with "Globus".

Site Name:	Globus
Site URL:	https://www.globus.org
Service URL:	https://www.globus.org/service/graph/authenticate_oauth_callback

Select An Identity Provider:

University of Chicago
University of Cincinnati Main Campus
University of Colorado at Boulder
University of Dayton

Search: University of Chicago

Remember this selection: ☐

Log On

By selecting "Log On", you agree to CILogon's privacy policy.

For questions about this site, please see the FAQs or send email to help @ cilogon.org.
Know your responsibilities for using the CILogon Service.
See acknowledgements of support for this site.

Sign In

CNetID: foster

Hospital Employee?

Password: •••••••

Forgot your password?

Login

Signing in allows you to access multiple University of Chicago web applications while entering your CNetID and password only once. To end your session, simply close your browser.

Questions? Contact the IT Services Service Desk by phone at 2-5800 (773-702-5800), via email at itservices@uchicago.edu, or get walk-in help at the TECHB@R on the first floor of Regenstein Library during reference desk hours http://hours.lib.uchicago.edu/.

Alumni account holders may contact alumni-support@uchicago.edu or call 1-877-292-3945 between 9 AM and 3 PM CST with any questions.

THE UNIVERSITY OF CHICAGO

CILogon

XSEDE campus bridging use cases (http://hdl.handle.net/2142/43882)

1) XSEDE systems and services should allow **use of campus identity credentials** by supporting federated identity and authorization mechanisms

2) XSEDE should help campus system administrators create XSEDE-like environments on campus systems to smooth their users' transitions between campus and national systems

3) XSEDE services should provide "remote desktop" access that users on campuses can remotely view graphical displays generated by XSEDE systems

4) XSEDE should enable users on campuses to **conduct integrated data analysis** that includes data on both campus and XSEDE systems

5) XSEDE should enable users on campuses to **initiate automated workflows** (scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services

The data bridge

- Navigating XSEDE security mechanisms
- Moving data from campus systems to XSEDE (or vice versa)
- Moving results from XSEDE to campus (or vice versa)
- Keeping track of which portions of the data have and have not been moved

Globus and the research data lifecycle

Instrument

Compute Facility

Globus transfers files reliably, securely

2

4 Globus controls access to shared files on existing storage; no need to move files to cloud storage!

7 Curator reviews and approves; data set published on campus or other system

Transfer

1

Researcher initiates transfer request; or requested automatically by script, science gateway

3

Share

Researcher selects files to share, selects user or group, and sets access permissions

6

Researcher assembles data set; describes it using metadata (Dublin core and domain-specific)

Publication Repository

Collaborator logs in to Globus and accesses shared files; no local account required; download via Globus

5

Publish

6

- **SaaS → Only a web browser required**
- **Use storage system of your choice**
- **Access using your campus credentials**

Personal Computer

Peers, collaborators search and discover datasets; transfer and share using Globus

8

Discover

17

XSEDE campus bridging use cases (http://hdl.handle.net/2142/43882)

1) XSEDE systems and services should allow **use of campus identity credentials** by supporting federated identity and authorization mechanisms

2) XSEDE should help campus system administrators create XSEDE-like environments on campus systems to smooth their users' transitions between campus and national systems

3) XSEDE services should provide "remote desktop" access that users on campuses can remotely view graphical display generated by XSEDE systems

4) XSEDE should enable users on campuses to **conduct integrated data analysis** that includes data on both campus and XSEDE systems

5) XSEDE should enable users on campuses to **initiate automated workflows** (scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services

Initiating automated workflows

- Navigating XSEDE security mechanisms
  - See "enable use of campus credentials"
- Moving data as part of file management tasks
  - See "enable integrated data analysis"
- Allowing computation tasks to access remote data when moving the data is more costly than accessing it remotely
  - GFFS as a virtual file system
- Controlling the execution of computation tasks on campus and XSEDE systems
  - XSEDE EMS: GENESIS-II and UNICORE

More thoughts on campus bridging

- A university is a set of talented, under-resourced entrepreneurial "small businesses" (labs)
- They need solutions that require little expertise and time

→ Small biz uses software-as-a-service (SaaS). Can labs also?

- Globus research data management services demonstrate value of SaaS approach for science
    - File transfer
    - File sharing
    - Identify, credential, and group management
    - Data publication

Globus by the numbers

5 major services	**100** petabytes transferred	**20 billion** files processed	**25,000** registered users
130 federated campus IdPs	**8,000** managed storage systems	**99.95%** uptime over past 2 years	**>30** institutional subscribers
3 months longest transfer	**1 petabyte** biggest transfer	**50M** most files in one transfer	**13** national labs use services

Globus service APIs serve as a science platform
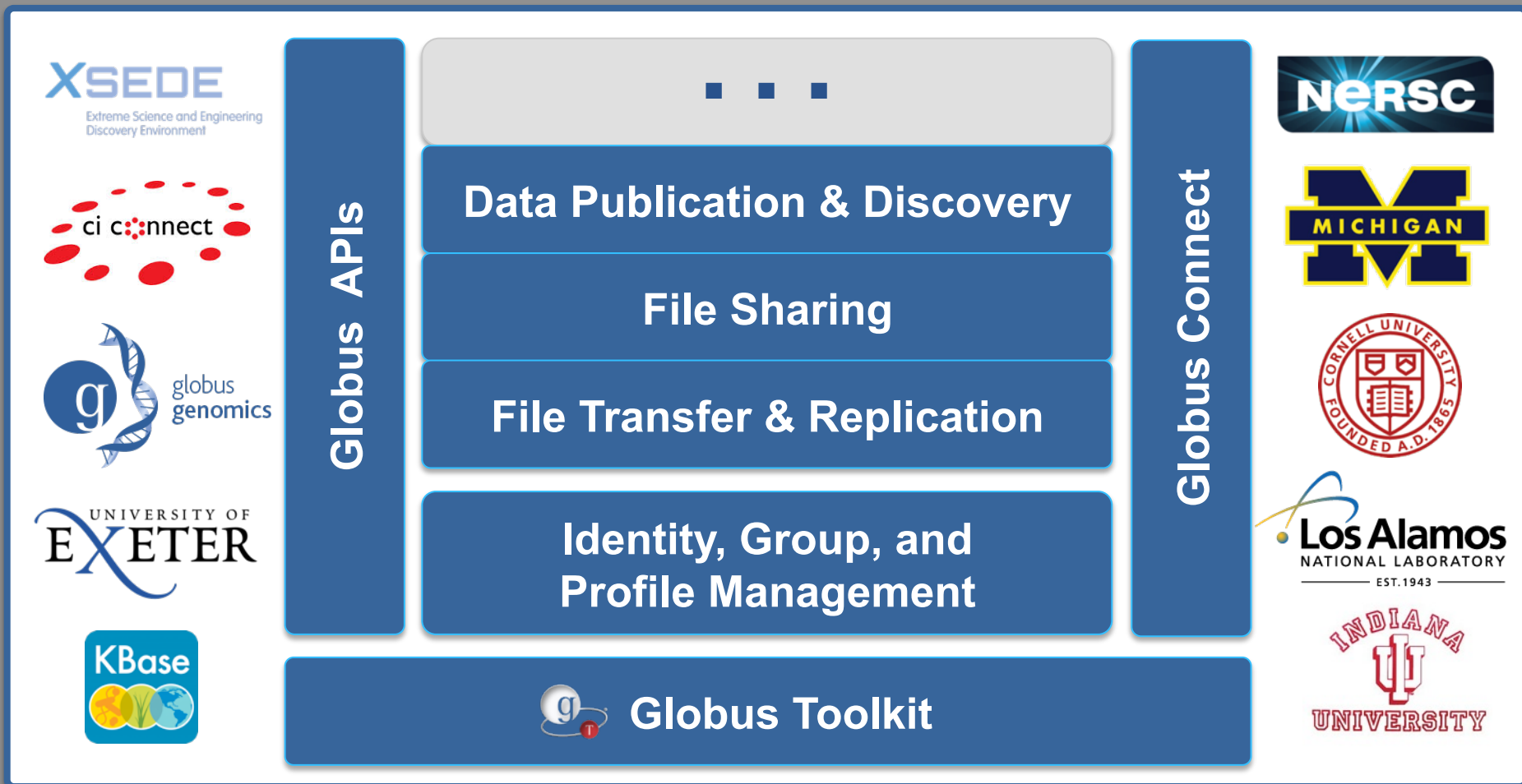
CISL Research Data Archive

RDA

Managed by NCAR's Data Support Section
Data for Atmospheric and Geosciences Research

Go to Dataset: `nnn.n`

| Home | Find Data | Ancillary Services | About/Contact | Data Citation | Web Services | | For Staff |

Dataset Search:

Keyword(s) [_____] [Search] **Advanced Options**

Look For Data:

All Datasets	Variable/Parameter	Type of Data
Time Resolution	Platform	Spatial Resolution
Topic/Subtopic	Project/Experiment	Supports Project
Data Format	Location	Recently Added/Updated

Recently Added Datasets: (within the last 6 months)

- NOAA/CIRES Twentieth Century Global Reanalysis Version 2c
- NCEP GDAS/FNL Surface Flux Grids
- NCEP GFS 0.25 Degree Global Forecast Auxiliary Grids Historical Archive
- Cloud Properties from ISCCP and PATMOS-x Corrected for Spurious Variability Related to Changes in Satellite Orbits, Instrument Calibrations, and Other Factors
- NCEP GFS 0.25 Degree Global Forecast Grids Historical Archive
- NOAA CPC Morphing Technique (CMORPH) Global Precipitation Analyses Version 0.x (June 2014 -

Get Help:

- Frequently Asked Questions
- Reset your password
- A-Z Site Index
- RDA Users Email List
- RDA Blog
- Email Us

From Our Blog:

- **ds608.0 NARR 20090401-20150131 rerun4 updates**
- **GRIB1, GRIB2, NetCDF: What do I use?**
- **The data starts here**
- **2015 Unidata Users Workshop**
- **Accessing Atmospheric and Oceanographic Data Formats with ArcGIS**

More blog posts ...

GLADE Users:

Much of the RDA is directly accessible from CISL's **GL**obally **A**ccessible **D**ata **E**nvironment. /glade files can be read directly in place from Yellowstone and Geyser/Caldera. You can find more information under the "Data Access" tab of individual datasets, including detailed lists of /glade files.

Globus platform services enable new application capabilities

here •
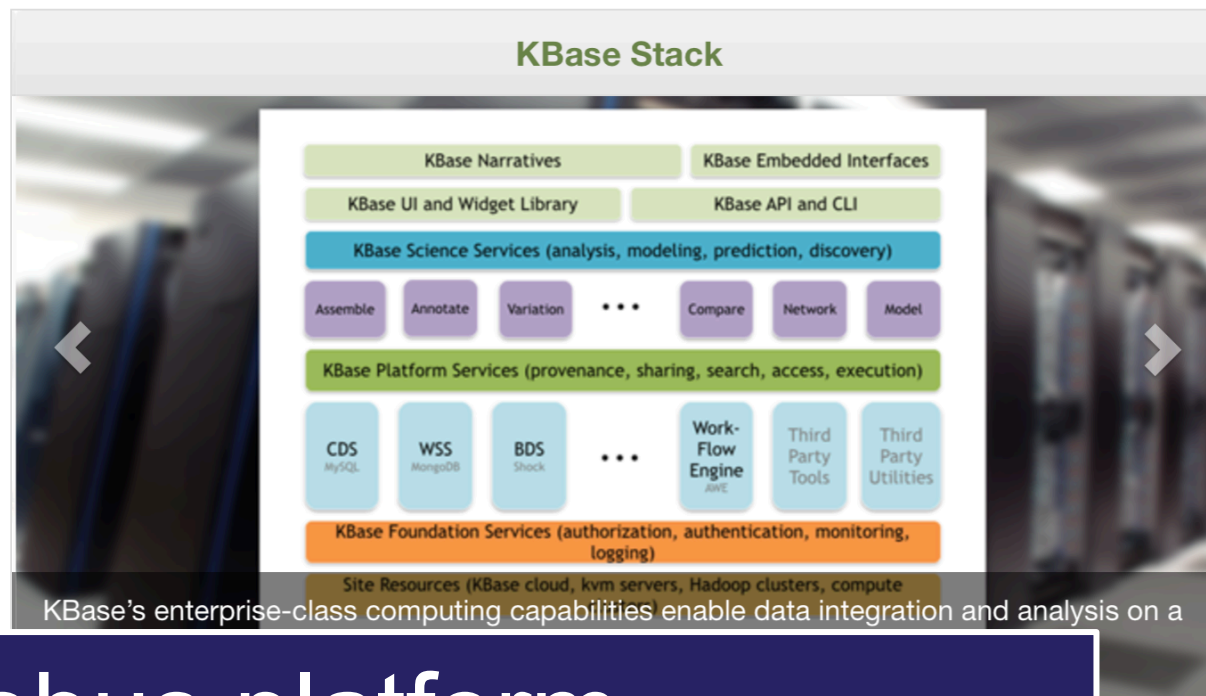
Globus platform accelerates development of new services

Thank you to our sponsors!

For more information: foster@anl.gov Thanks to co-authors and Globus team

Globus services (globus.org)

- Foster, I. **Globus Online: Accelerating and democratizing science through cloud-based services**. *IEEE Internet Computing*(May/June):70-73, 2011.

- Chard, K., Tuecke, S. and Foster, I. **Efficient and Secure Transfer, Synchronization, and Sharing of Big Data**. *Cloud Computing, IEEE*, 1(3):46-55, 2014.

- Chard, K., Foster, I. and Tuecke, S. **Globus Platform-as-a-Service for Collaborative Science Applications**. *Concurrency - Practice and Experience*, 27(2):290-305, 2014.

Publication (globus.org/data-publication)

- Chard, K., Pruyne, J., Blaiszik, B., Ananthakrishnan, R., Tuecke, S. and Foster, I., **Globus Data Publication as a Service: Lowering Barriers to Reproducible Science**. 11th IEEE International Conference on eScience Munich, Germany, 2015

Discovery engines

- Foster, I., Ananthakrishnan, R., Blaiszik, B., Chard, K., Osborn, R., Tuecke, S., Wilde, M. and Wozniak, J. **Networking materials data: Accelerating discovery at an experimental facility**. Big Data and High Performance Computing, 2015.

Acknowledgements

- Work presented here from many individuals and groups & sites
- **Special thanks** to Globus & HTCondor teams!
- ATLAS Connect and OSG teams
  - Dave Lesny (Illinois), Lincoln Bryant, David Champion, Suchandra Thapa (Chicago), Derek Weitzel (Nebraska)
- Duke, Syracuse, Clemson, UChicago, UWisc

A little bit of context

- Efforts began in 2013 to solve two problems:
  - How to provide a virtual cluster experience for small research labs requiring distributed high throughput computing resources (**OSG Connect**)
  - Extend the batch capacity of ATLAS (high energy physics at CERN) Tier3 clusters (**ATLAS Connect**)
- Elements: Unix Acct. ‣ Software ‣ CPU ‣ Data
- HTCondor to be key linking these ... but how?

Unix Acct ‣ Software ‣ CPU ‣ Data

- We need solutions for each
- No "development" effort, only integration
- Leverage proven technologies and advanced
  CI activities

  ⇒ Globus Identity, CI-Logon, InCommon

  ⇒ HTCondor, Glideins, CernVM-FS, Xrootd

OSG Connect

Has an *identity bridge*: local campus identity (CILogon) ‣ OSG Connect identity (Globus) ‣ virtual organization (OSG)

+ HTCondor Glidein Overlay

⇒ **Virtual HTC cluster experience**
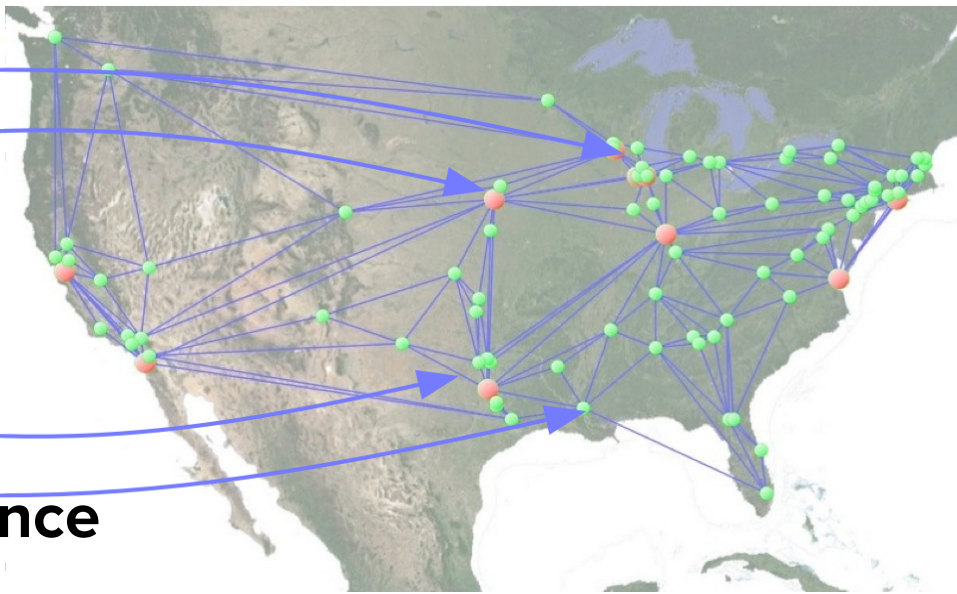
ATLAS Connect

- Many ATLAS Tier3's use HTCondor
- Simple to add flocking targets for one
  - But not managing a mesh (30 sites x N flocking targets)
- Centralize the flocking services
- Provide production backend for CERN based system to shared university clusters
- Leverage work from OSG Connect for end-user physicists

ATLAS Connect: Tier3 Users

Leverage **standard** HTCondor flocking and ClassAd matching

Various job schedulers, queue policies, CVMFS access methods, local scratch & squid setups.
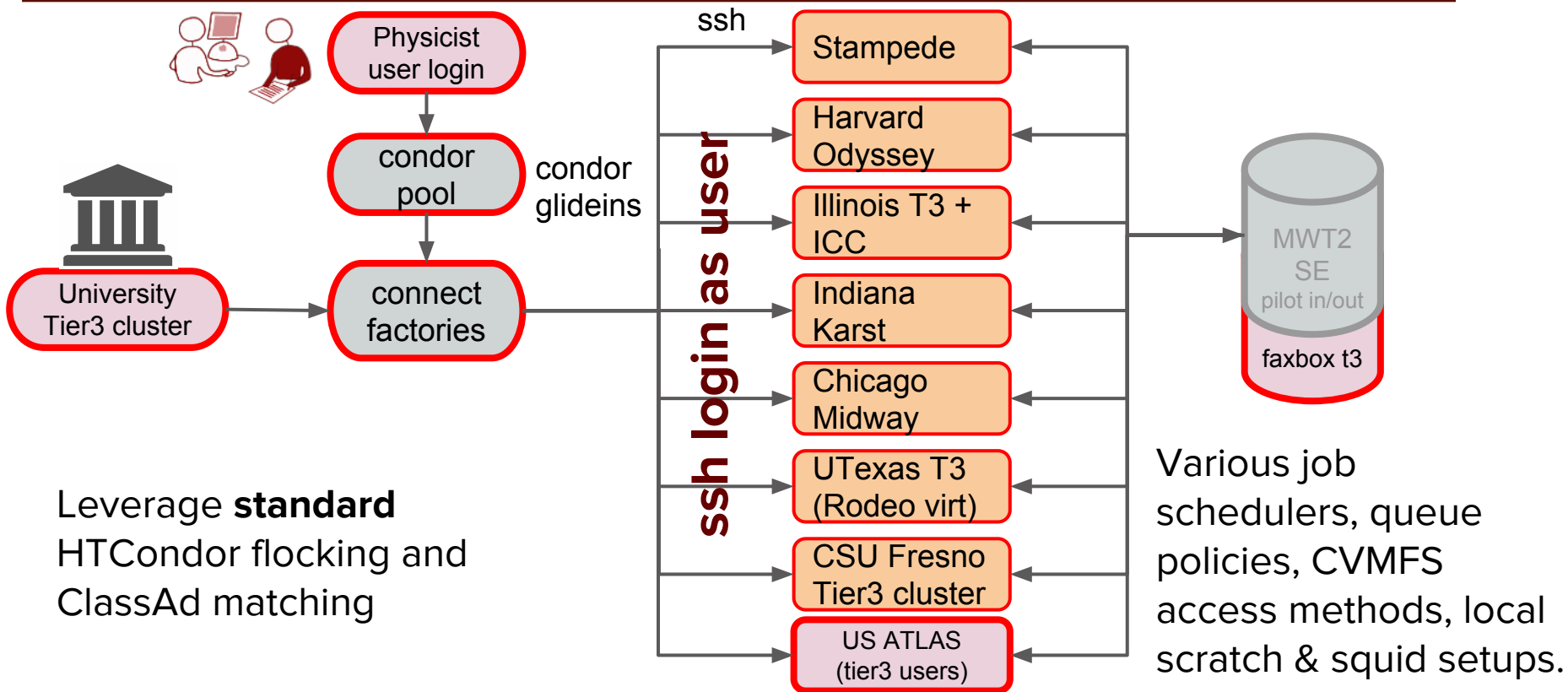
ATLAS Connect: Production backend
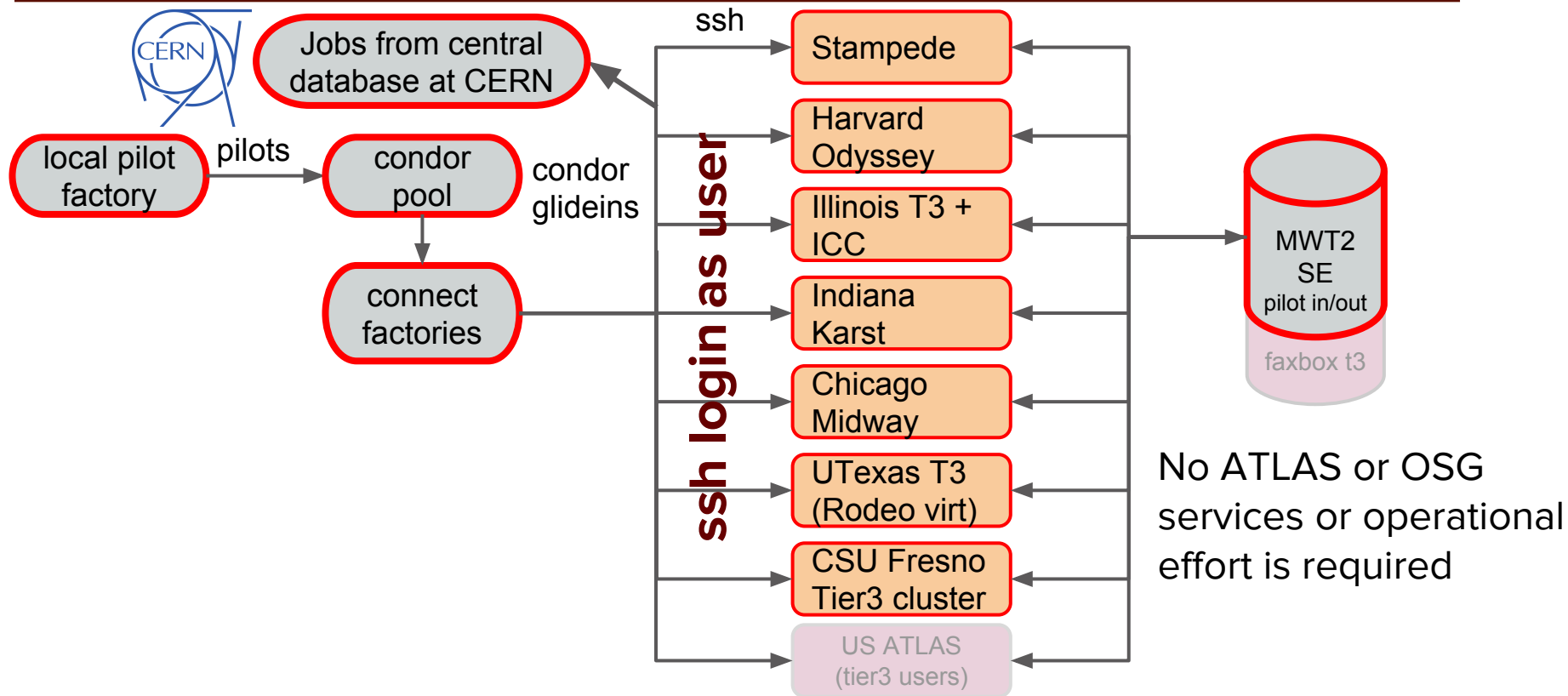
Core Usage by Connect Panda Queues

	max	current
CONNECT	1035	212
CONNECT_CLOUD	1842	23
CONNECT_MCORE	672	80
CONNECT_ES	24	13
CONNECT_ES_MCORE	144	72
CONNECT_PILE	0	0
CONNECT_PILE_MCORE	0	0
ANALY_CONNECT	484	194
ANALY_CONNECT_SHORT	570	136
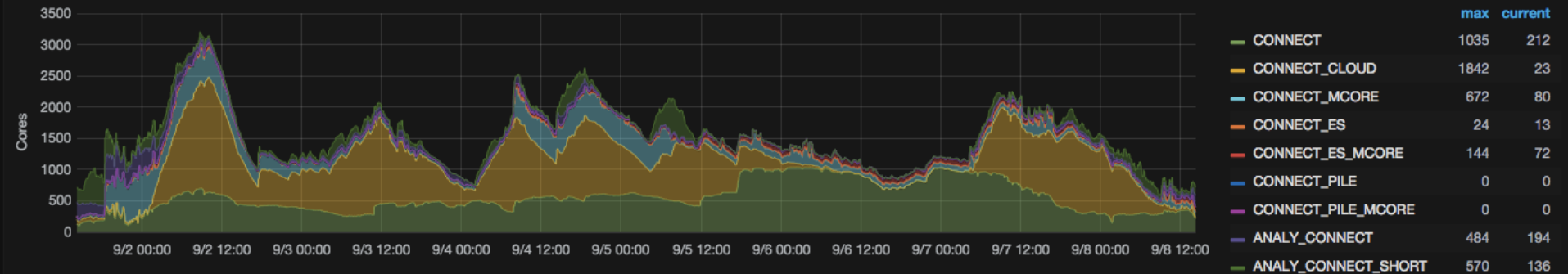
Wall Clock consumption All Jobs in seconds
8759 Hours from Week 26 of 2014 to Week 26 of 2015 UTC

3M CPU hours
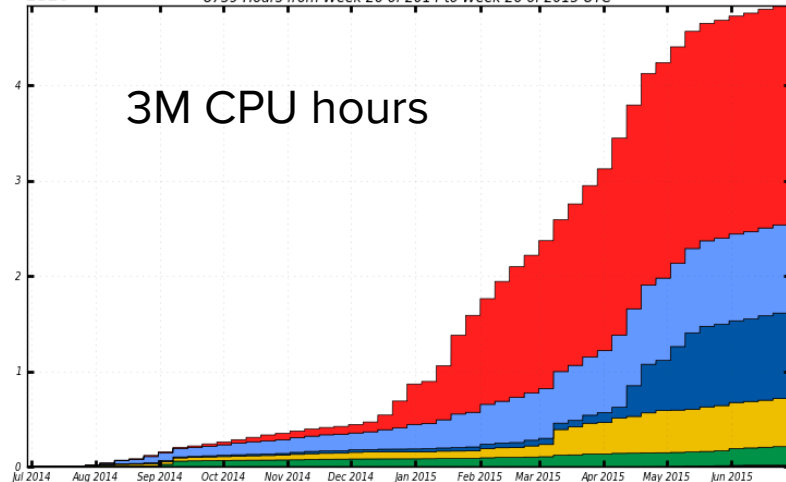
ANALY_CONNECT (22,965,342,049)   ANALY_CONNECT_SHORT (9,249,101,720)   CONNECT_MCORE (8,970,907,333)
CONNECT (5,047,015,168)   CONNECT_CLOUD (1,990,668,399)   CONNECT_PILE_MCORE (98,160,464)
CONNECT_PILE (50,144,424)

Total: 48,371,339,557 , Average Rate: 1,533 /s

Easy to plug in additional resources or grow with new allocations:

● university clusters
● xsede clusters

As important is the approach

- Focus is on **integration** and focused expertise rather than developing something new
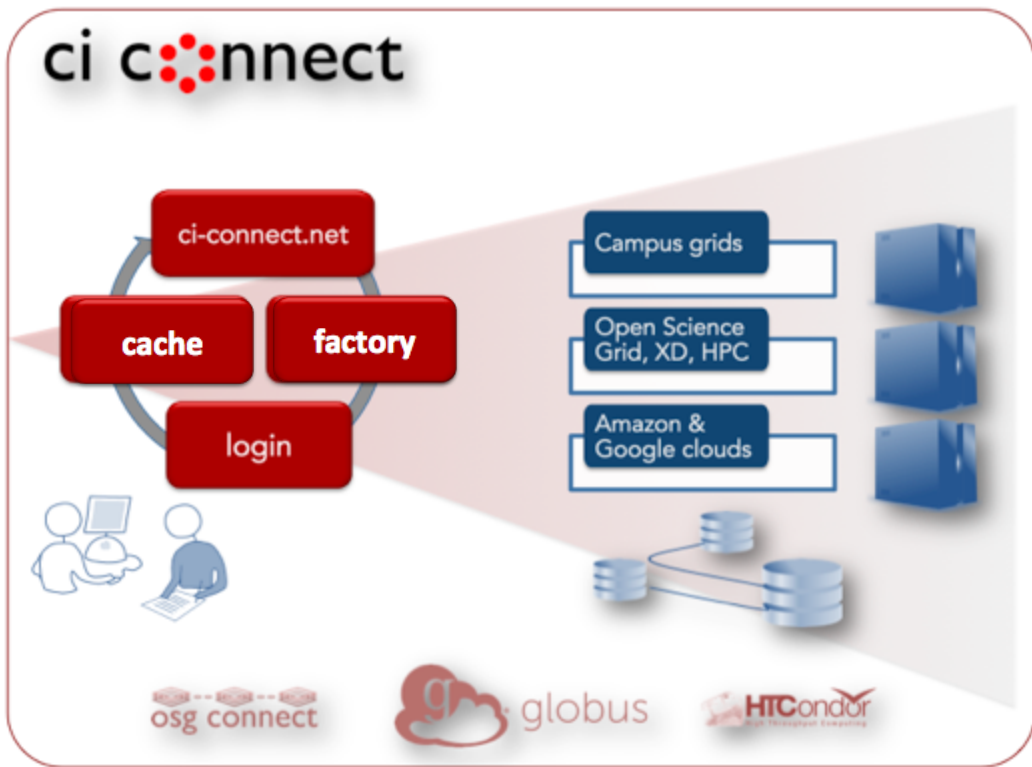
As important is the approach

- Deliver as **hosted service**
- Minimize services & equipment at resource endpoints
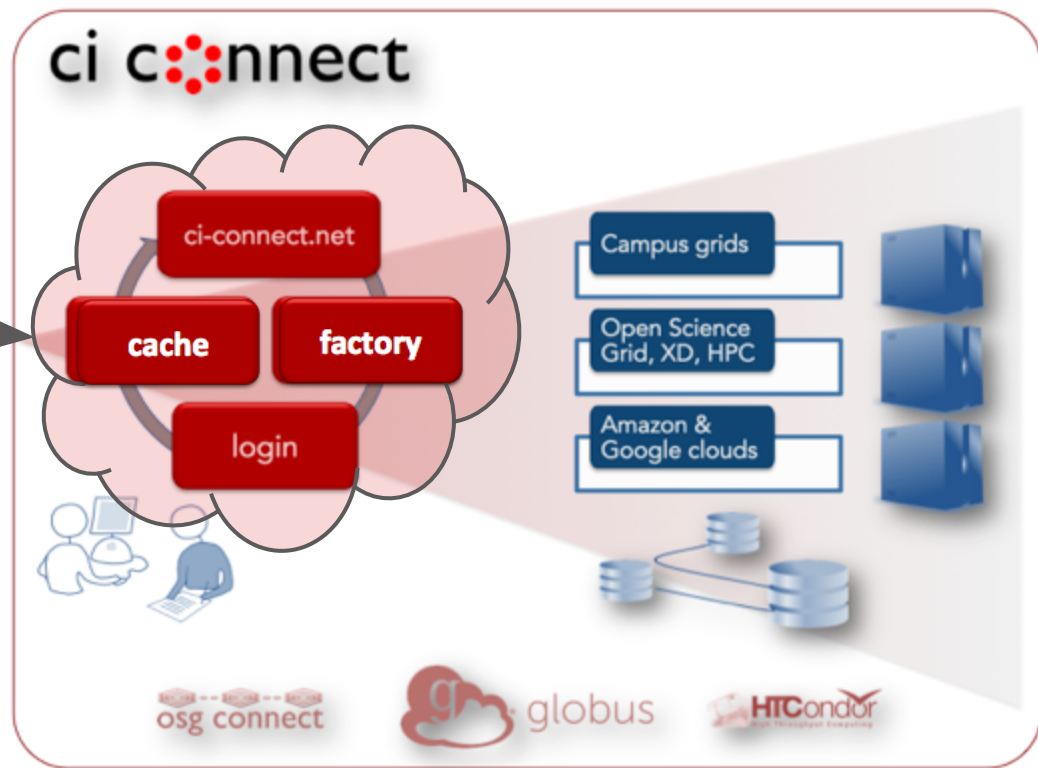
# Bringing HTCondor pools to campus

- Campus Connect Client (install locally)
- Virtual extension of /home comforts:
  - Local software, campus storage, tools
  - Marshall resources from one location & tool set:
    - local campus cluster allocation or general queue
    - XSEDE project allocation
    - shared resources via the OSG
    - multi-campus and community partnerships
    - public cloud resources

Organized into a 'local' queue

```
$ module load connect-client
$ connect setup
$ connect test
$ connect submit myjob.sub
$ connect q rwg
-- Submitter: login.ci-connect.uchicago.edu :
uchicago.edu
 ID       OWNER            SUBMITTED      RUN_TI
252624.0   rwg              9/2  14:21   0+00:0
252624.1   rwg              9/2  14:21   0+00:0
...
$ connect status
$ connect pull (results)
```

Submitted from UChicago Research
Computing Center cluster "Midway"

UChicago CI Connect Service:
Midway cluster + OSG;
to add SDSC Comet allocation &
partner clusters at XENON institutions

Early adopter community:
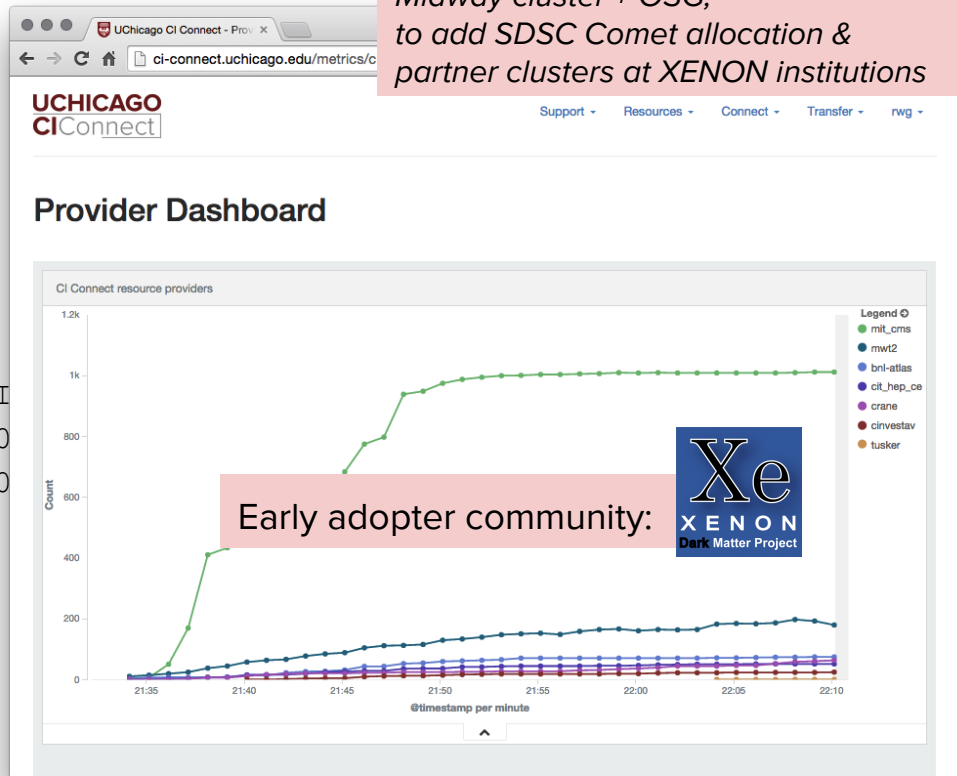
Campus Users (@ Clemson) + OSG

Submission from Palmetto cluster (local)

add OSG nodes

Sharing local resources with communities
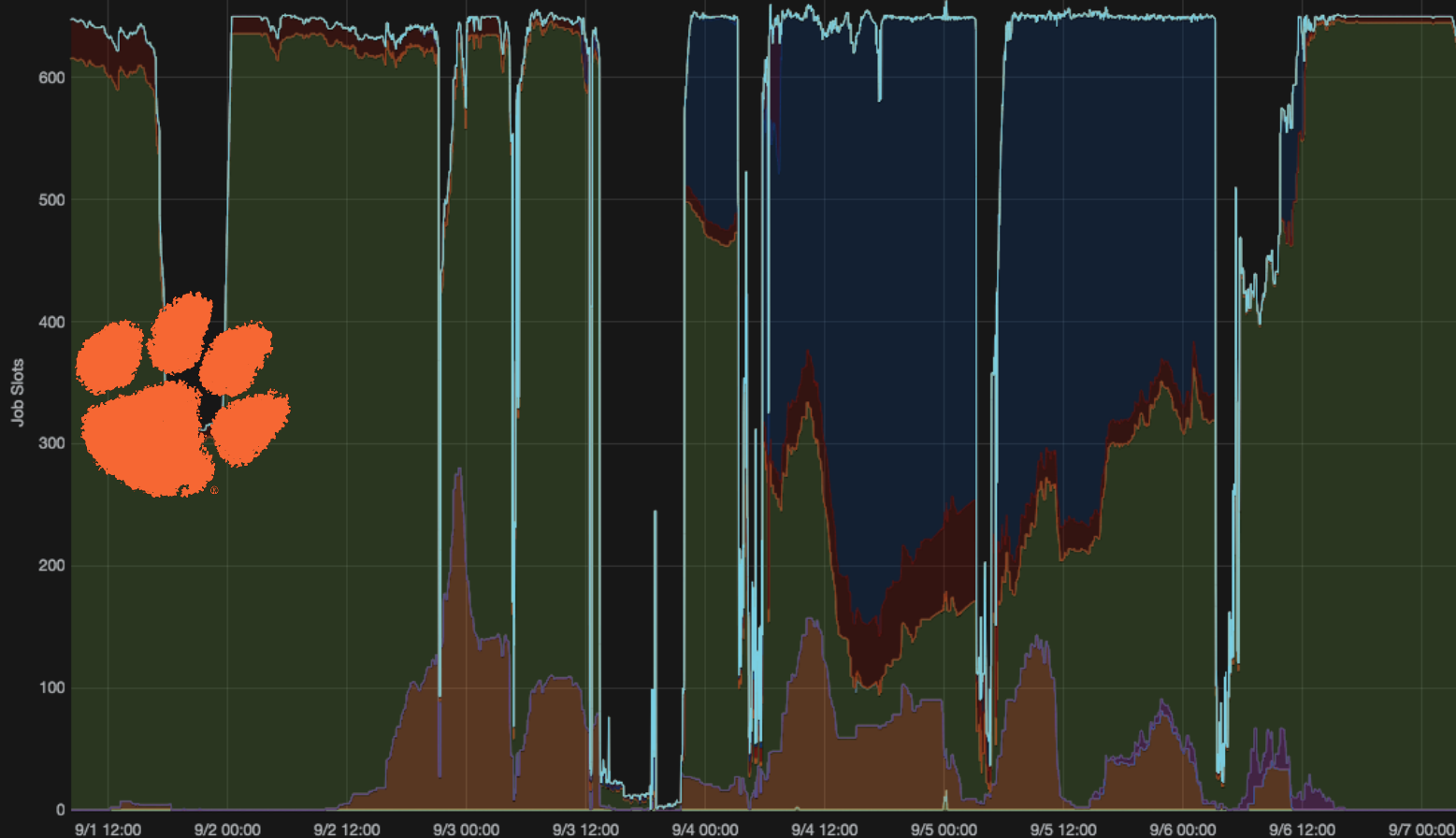
GLOW (UWisconsin-based campus researchers)

OSG Connect

Palmetto cluster shared 300k CPU-hrs in past 30 days
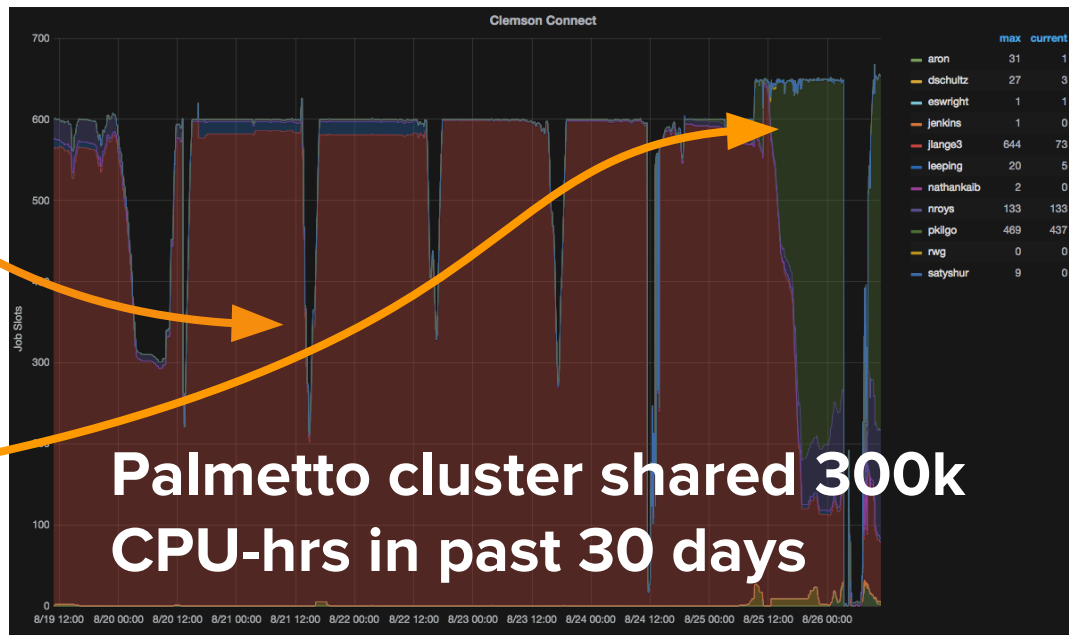
Clemson Connect

2015/09/08 (Last week)

19 active users

	max	current
aron	0	0
chekanov	16	0
dbala	0	0
dschultz	280	0
dweitzel	1	0
eswright	0	0
hkim453	48	0
jenkins	1	0
jlange3	646	549
kavandermeul	0	0
leeping	0	0
nathankaib	14	0
nroys	83	5
pkilgo	503	0
rpizarro	179	0
rwg	0	0
satyshur	8	0
ssericksen	0	0
wpoehlm	0	0

Going forward: automation & self-service

Make it simple to provision & dynamically configure multi-campus, community-based virtual cluster instances
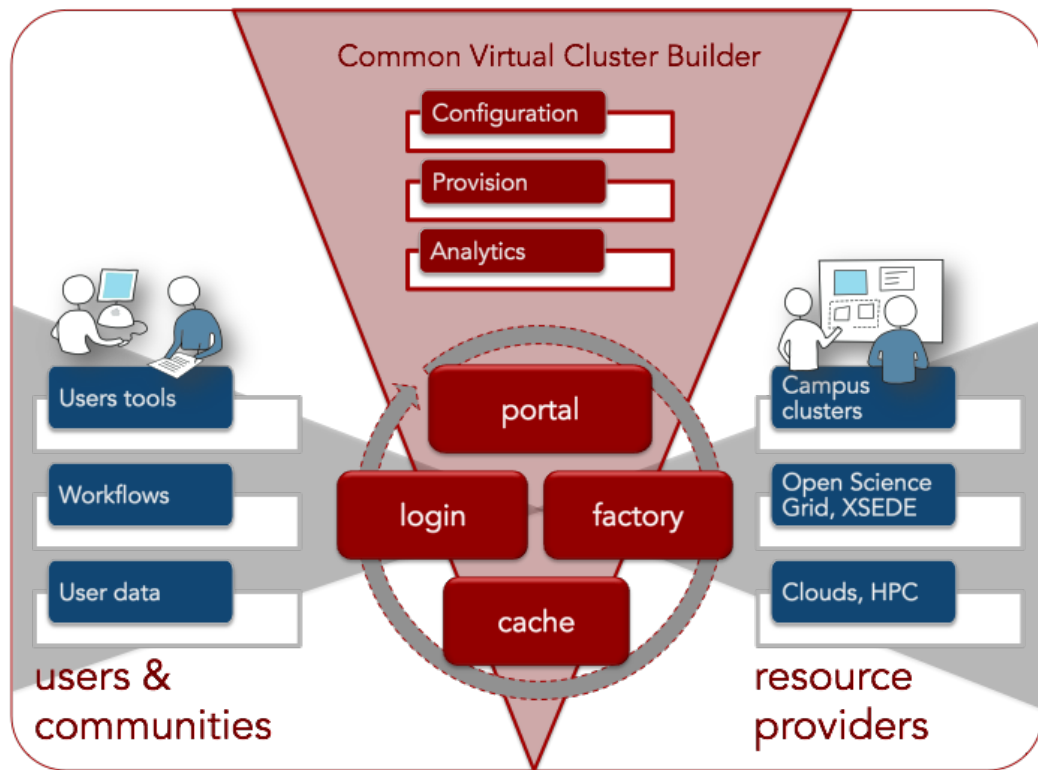
Summary

- Use existing, well-proven technologies to enable sharing among campuses
- Minimize operational effort and equipment at resource endpoints $\Rightarrow$ provide as a service
- Future areas of work:
  - automation & self service
  - adaptive 'policy-based' provisioning services
  - campus connect interfaces (user, compute, data)

The goal is to simplify secure resource sharing within and between campuses and research labs

Resources can mean anything, but here we will focus on compute resources, data resources, and storage resources

XSEDE

Campus Bridging Use cases

- #1- use campus identities
- #4 – access to distributed data resources
- #5 – workflows that consist of XSEDE and campus compute and data resources
- #6 – Shared Virtual Compute Facilities
- #7 – SVCF access "on a service for funds basis", "on demand"

XSEDE

Agenda

- The Global GFFS namespace

- Authentication & Authorization

- Data sharing via GFFS

- Cycle sharing via EMS

- Wrap up

Information & links

- Genesis II Omnibus Reference Manual (GORM)
  - http://genesis2.virginia.edu/wiki/Main/Documentation
- On-line videos/Tutorials
  - http://genesis2.virginia.edu/wiki/Main/Tutorials
- Sample JSDL
  - GFFS:/bin/sample_jobs_gjp; /bin/sample_jobs_jsdl
- Client installers
  - http://genesis2.virginia.edu/wiki/Main/Downloads
- Architectural documentation
  - XSEDE Architecture Level 3 Decomposition (L3D)
    - http://hdl.handle.net/2142/45115
  - Canonical Use Case one Architectural Response
    - http://hdl.handle.net/2142/73149
  - Campus bridging use case response

Movies are at www.genesis2.virginia.edu/wikiGFFS.eu

Tutorials on YouTube

Download and Install the GFFS Client (3:09)

GUI Client Basics (8:28)

Install a GFFS Container (21:37)

Copy files in and out of the GFFS (28:37)

Map the GFFS into your Linux file system using FUSE (9:39)

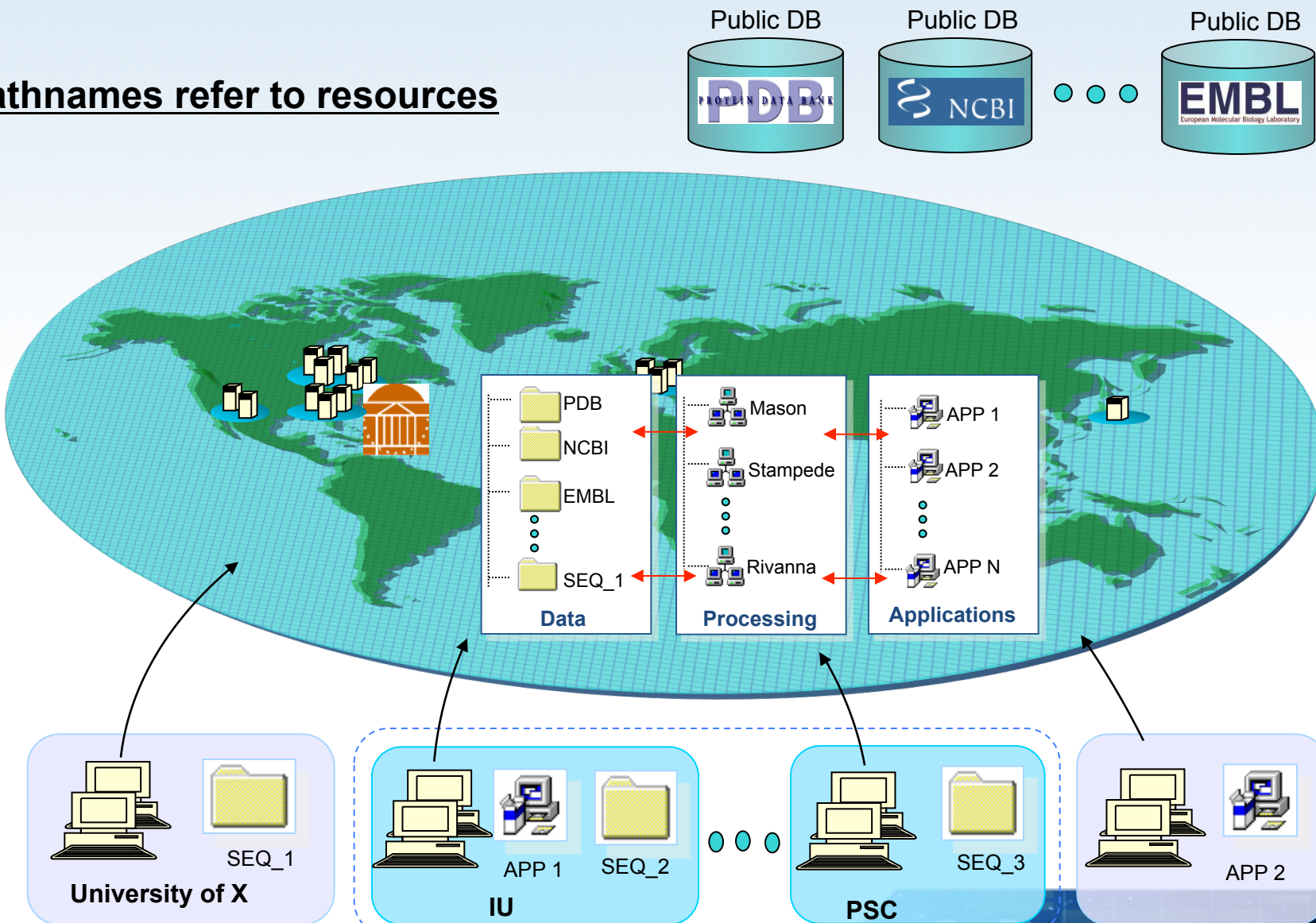Run a simple job with the GFFS (16:30)

XSEDE

GFFS
The namespace hooks everything together

XSEDE

Basic idea: map resources into a global directory structure

Pathnames refer to resources

All kinds of resources

- Compute resources
  - PBS queue on Forge, SGE queue on Ranger, a PBS queue on your cluster
- Data Resources
  - Your home directory at NCSA, your home directory in your lab, and instrument in your lab, a relational database, the archive at PSC
- Identity Resources
  - The XSEDE Kerberos infrastructure, your Kerberos system, your LDAP, or create your own identities
- Scheduling resources
  - Meta schedulers, global job queues, build your own job queue that sends jobs to your cluster and your colleagues cluster
- Job resources
  - Jobs are resources, you can "ls" the jobs in a queue, you can "ls" the working directory of the job while it is running, as well as copy files in and out
- Groups/role resources
  - Create and manage your own groups

View of "/" and "/home"
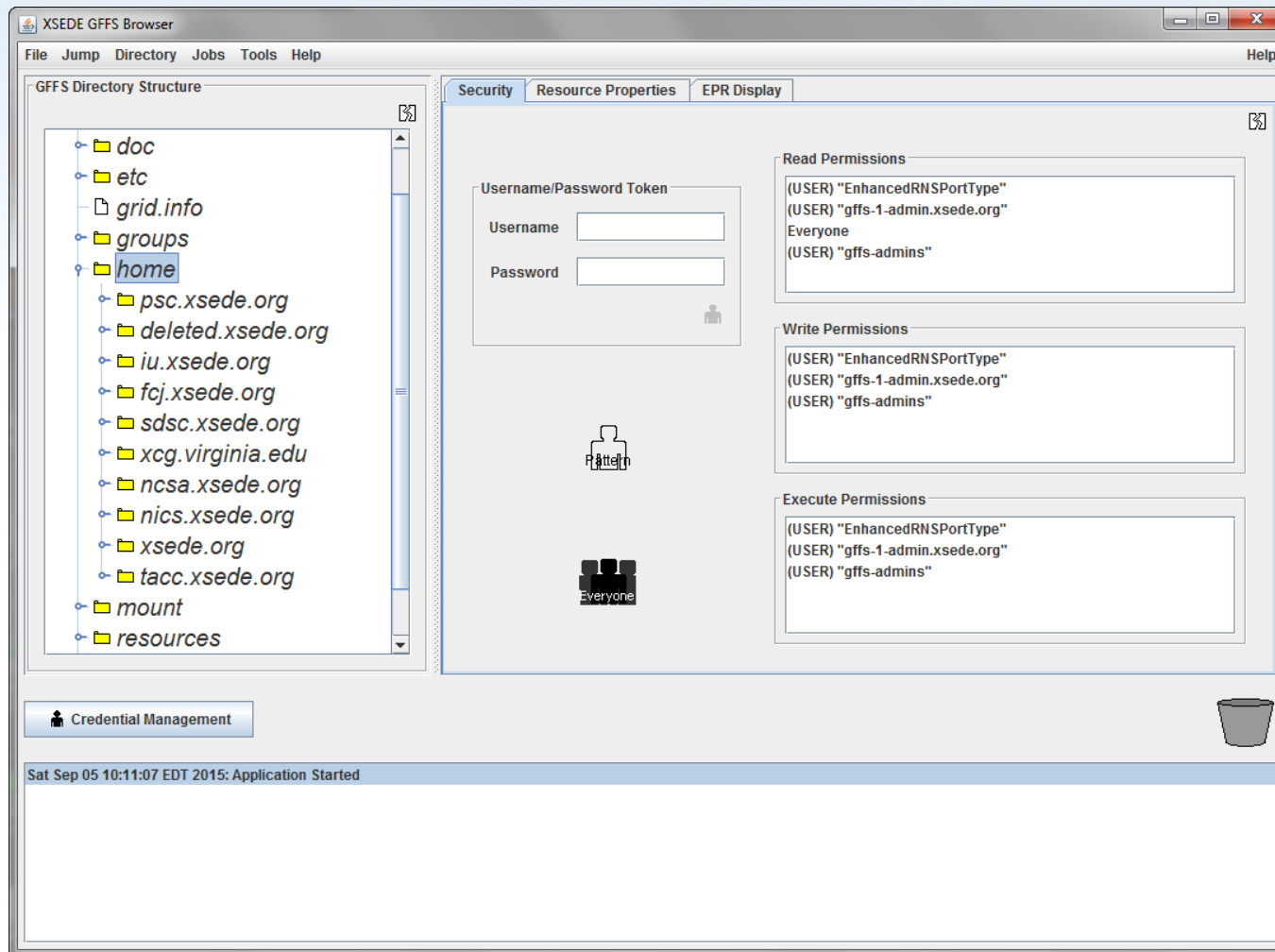
Paths point to everything – including identities. Note ACLs.
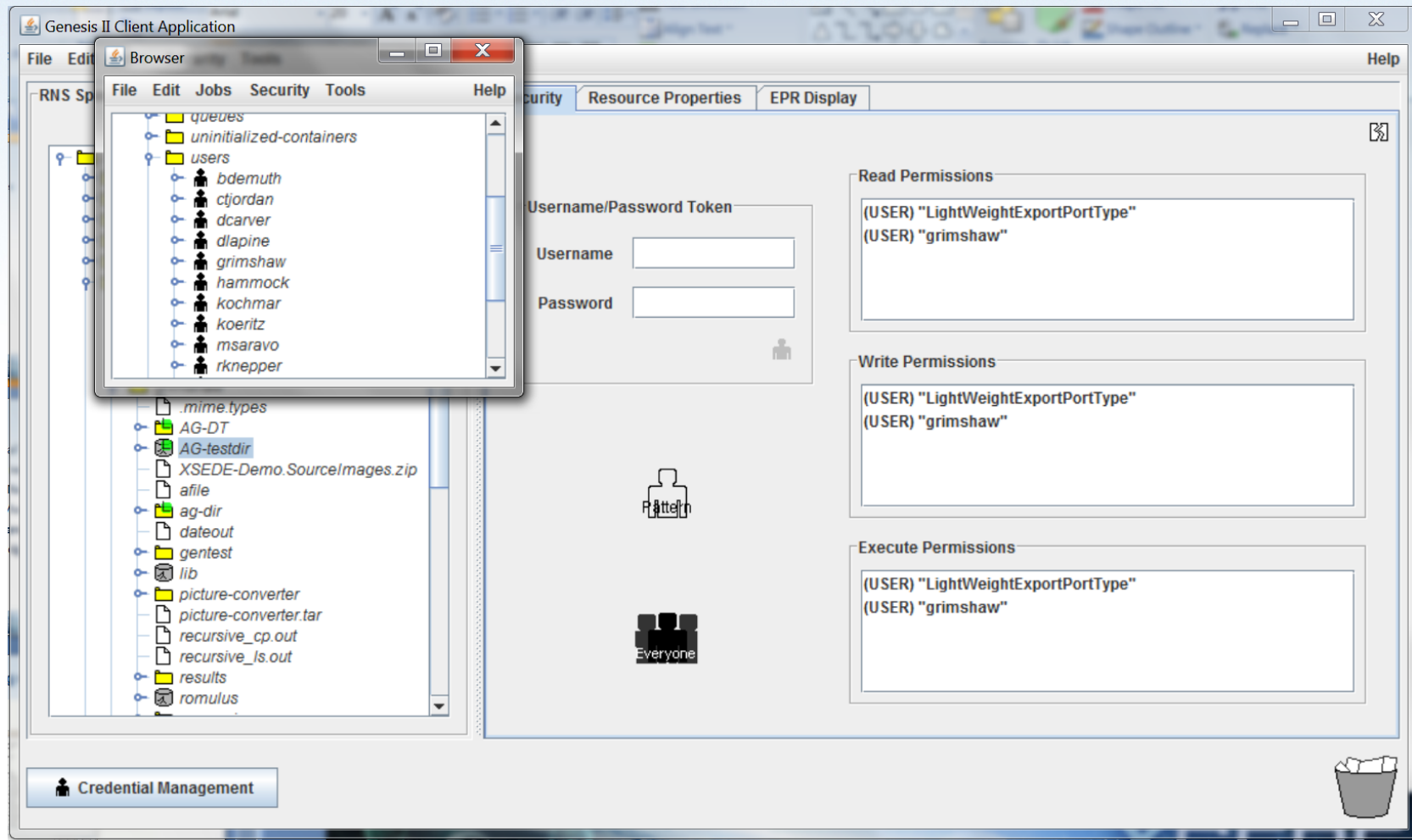
Identity and Authentication

XSEDE

Credential Wallet model – XSEDE case

- Start with a session certificate (X.509)
- xsedeLogin
  - Acquire myproxy certificate and a set of delegated SAML assertions for identity and group membership.
  - Will be shifting to Globus this PY
- You can add credentials to the wallet and take them away.
- Each wallet is contained in a *calling context.*
- Calling contexts are kept on disk as well, $GENII_USER_DIR
- You can add your own groups and identities.

XSEDE

Example

```
grimshaw@cicero:~$ grid xsedeLogin --username=grimshaw --password=**************
Replacing client tool identity with MyProxy credentials for "CN=Andrew Grimshaw, O=National Center for Supercomputing Applications, C=US".
grimshaw@cicero:~$ grid whoami
Client Tool Identity:
(CONNECTION) "Andrew Grimshaw"
Additional Credentials:
(USER) "grimshaw" -> (CONNECTION) "Andrew Grimshaw"
(GROUP) "gffs-tutorial-group" -> (CONNECTION) "Andrew Grimshaw"
(GROUP) "gffs-users" -> (CONNECTION) "Andrew Grimshaw"
```

XSEDE

Global Federated File System

XSEDE

Four Examples Illustrate Typical Uses Cases

- Directly sharing data with a collaborator at another institution
- Accessing your own data scattered on different resources: at home, in the lab, in the department, at a center, on a colleagues machine
- Accessing data at an NSF center from a home or campus
- Accessing data on a campus machine from an NSF center – particularly to run jobs

We'll come back to these later

XSEDE

GFFS – Basic Idea

- Access the global namespace
  - Command line
  - Graphical User Interface
  - Map into local file system, "mount" XSEDE

- Put resources into the global namespace
  - Export directories
  - Storage with different QoS properties
  - Clusters, supercomputers, cloud resources
  - Identities
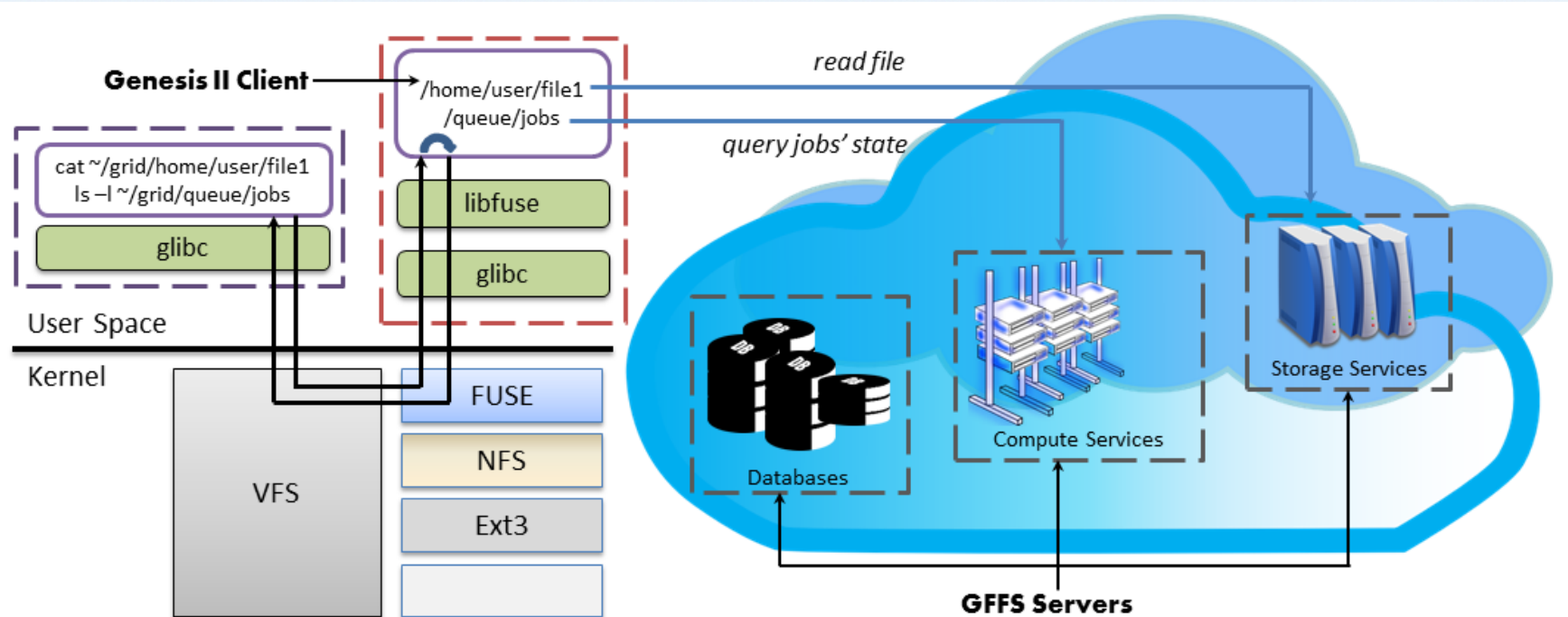
XSEDE

Accessing the GFFS via GUI
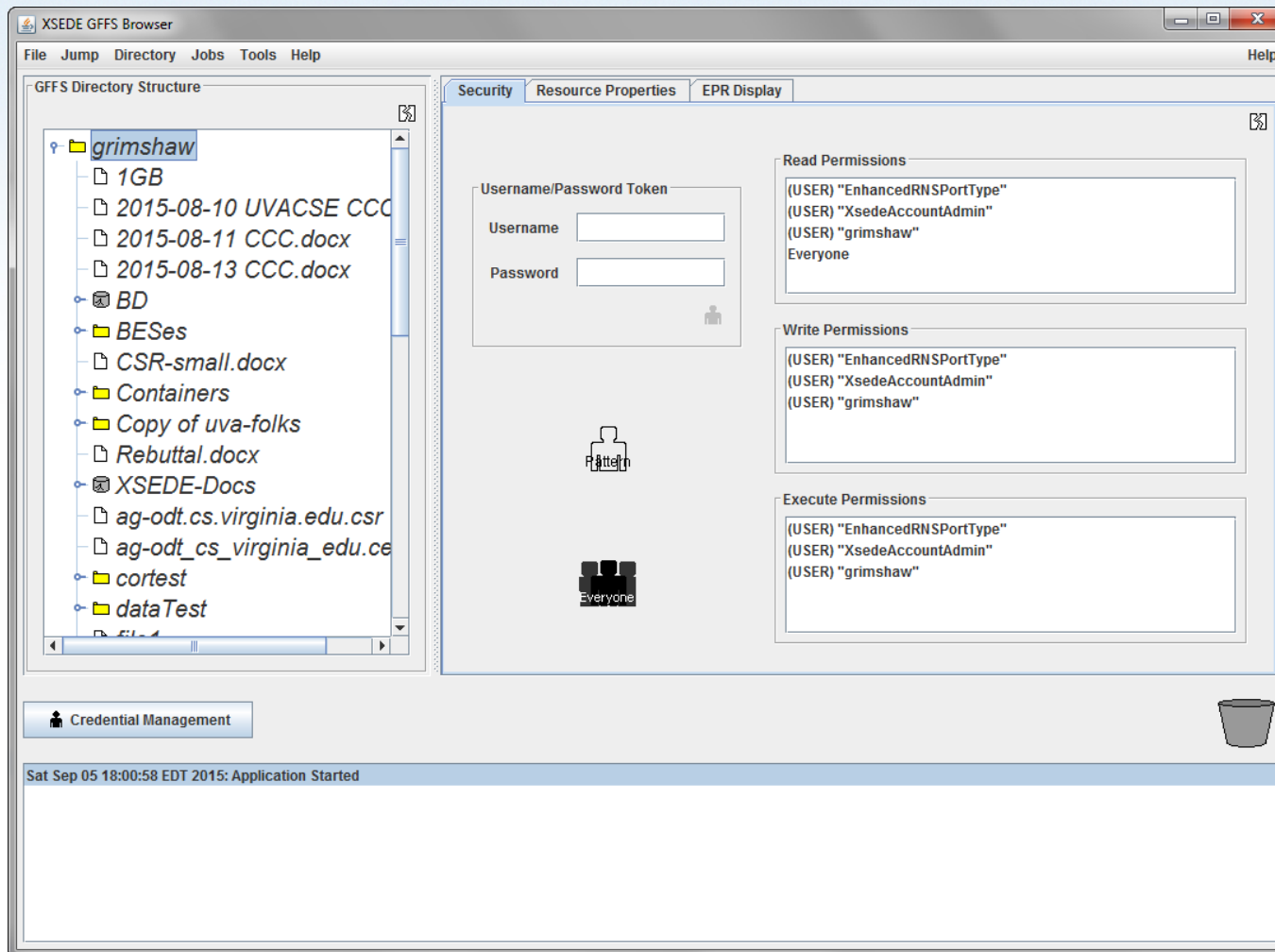
Shared a directory at TACC
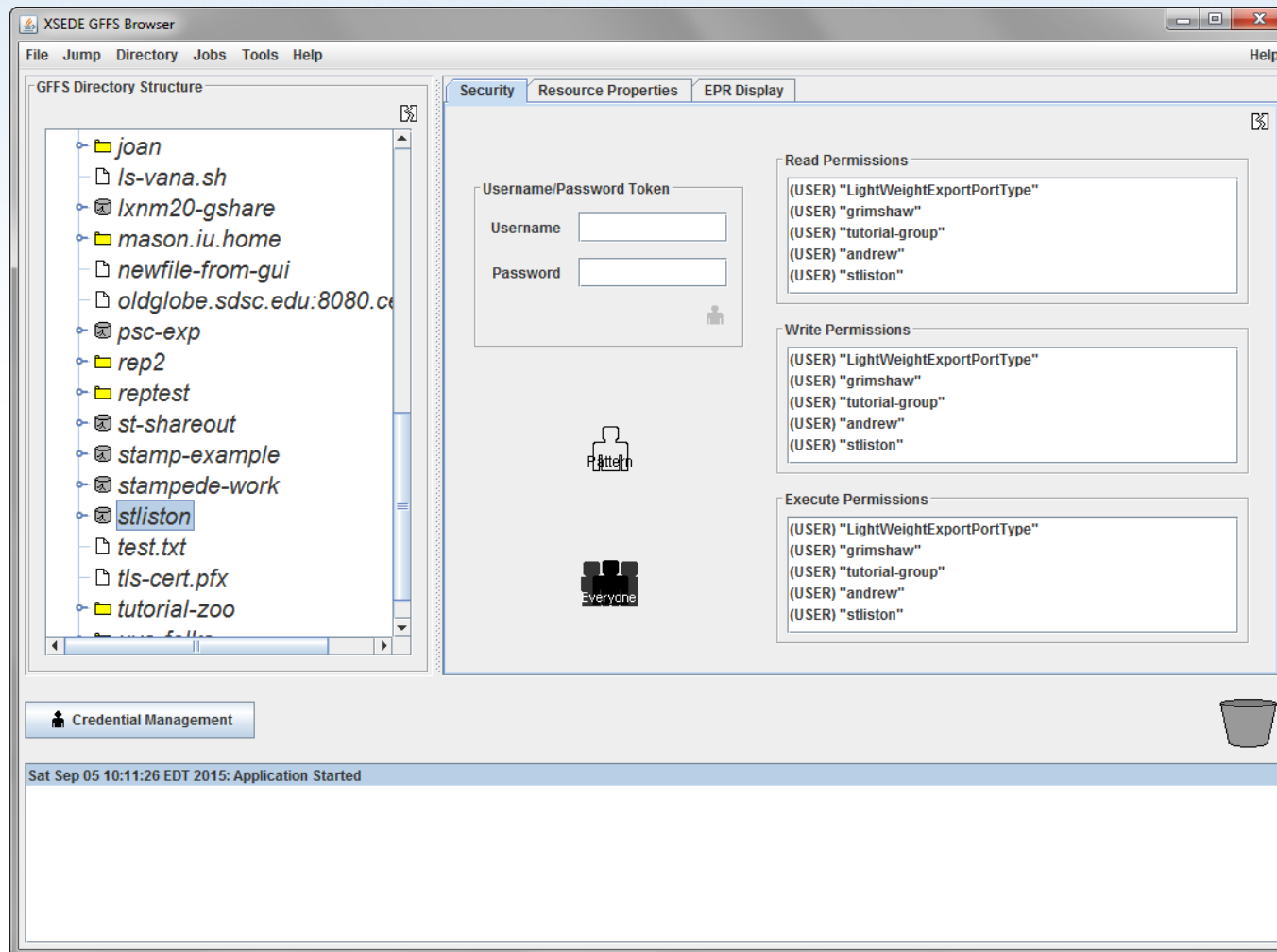
FUSE Mounting the Grid: Overview

- File system in User Space (FUSE) is a loadable kernel module for Unix-like computer operating systems that lets non-privileged users create their own file systems without editing kernel code

- We use FUSE to provide accesses to gird resources directly from your Linux file system via a directory mount point

FUSE access to the GFFS

ls of my home directory

```
grimshaw@cicero> pwd
/if8/grimshaw/grid/home/xsede.org/grimshaw
grimshaw@cicero> ls
1GB                         grimshaw.jpg              Rebuttal.docx
2015-08-10 UVACSE CCC.pptx  invemere.jpg.jpg          rep2
2015-08-11 CCC.docx         iu-collab                 reptest
2015-08-13 CCC.docx         iu-dir                    stampede-work
ag-odt_cs_virginia_edu.cer  jaydir                    stamp-example
ag-odt.cs.virginia.edu.csr  joan                      stliston
BD                          ls-vana.sh                st-shareout
BESes                       lxnm20-gshare             testforclass
Containers                  mason.iu.home             test.txt
Copy of uva-folks           newfile-from-gui          tls-cert.pfx
cortest                     oldglobe.sdsc.edu:8080.cer  tutorial-zoo
CSR-small.docx              pnodes                    uva-folks
file1                       prodhan                   xcgandrew
gnomad                      psc-exp                   XSEDE-Docs
grimshaw@cicero> |
```

XSEDE

ls of Anindya's impromptu queue

Result

- XSEDE resources regardless of location can be accessed via the file system
  - Files and directories can be accessed by programs and shell scripts as if they were local files
  - Jobs can be started by copying job descriptions into directories
  - One can see the jobs running or queued by doing an "ls".
  - One can "cd" into a running job and access the working directory where the job is running directly
- More on this later

Ways to Add Data into the Grid

- Export file system directory
- Create/copy files and directories

Putting resources into the GFFS

- Exporting directory trees

- Changes made in native file system visible to GFFS

- Changes made to files via GFFS propagated to native files
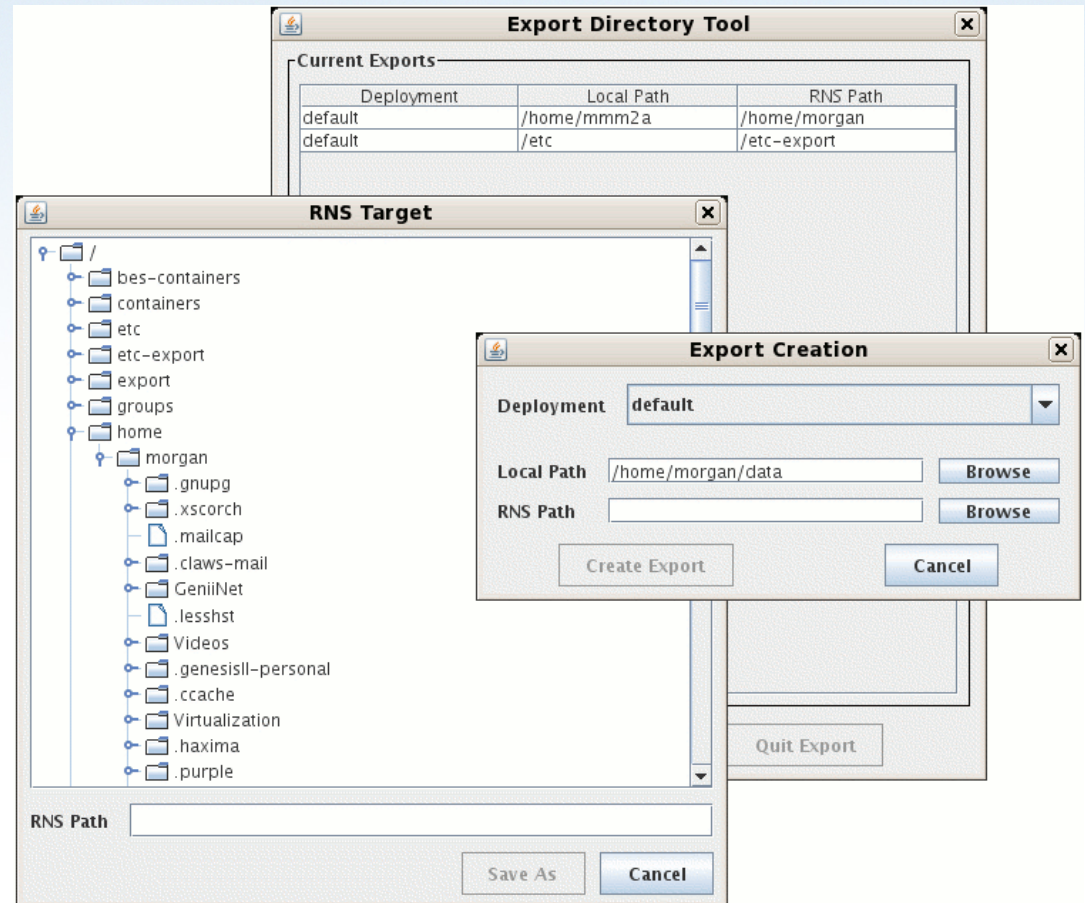
Exporting: Mapping a local directory structure into the global namespace

User runs export command

Export Service

Export service mounts local directory into global namespace

myFiles

/home

user

myExport

Export services redirects calls from grid export to local file system

XSEDE
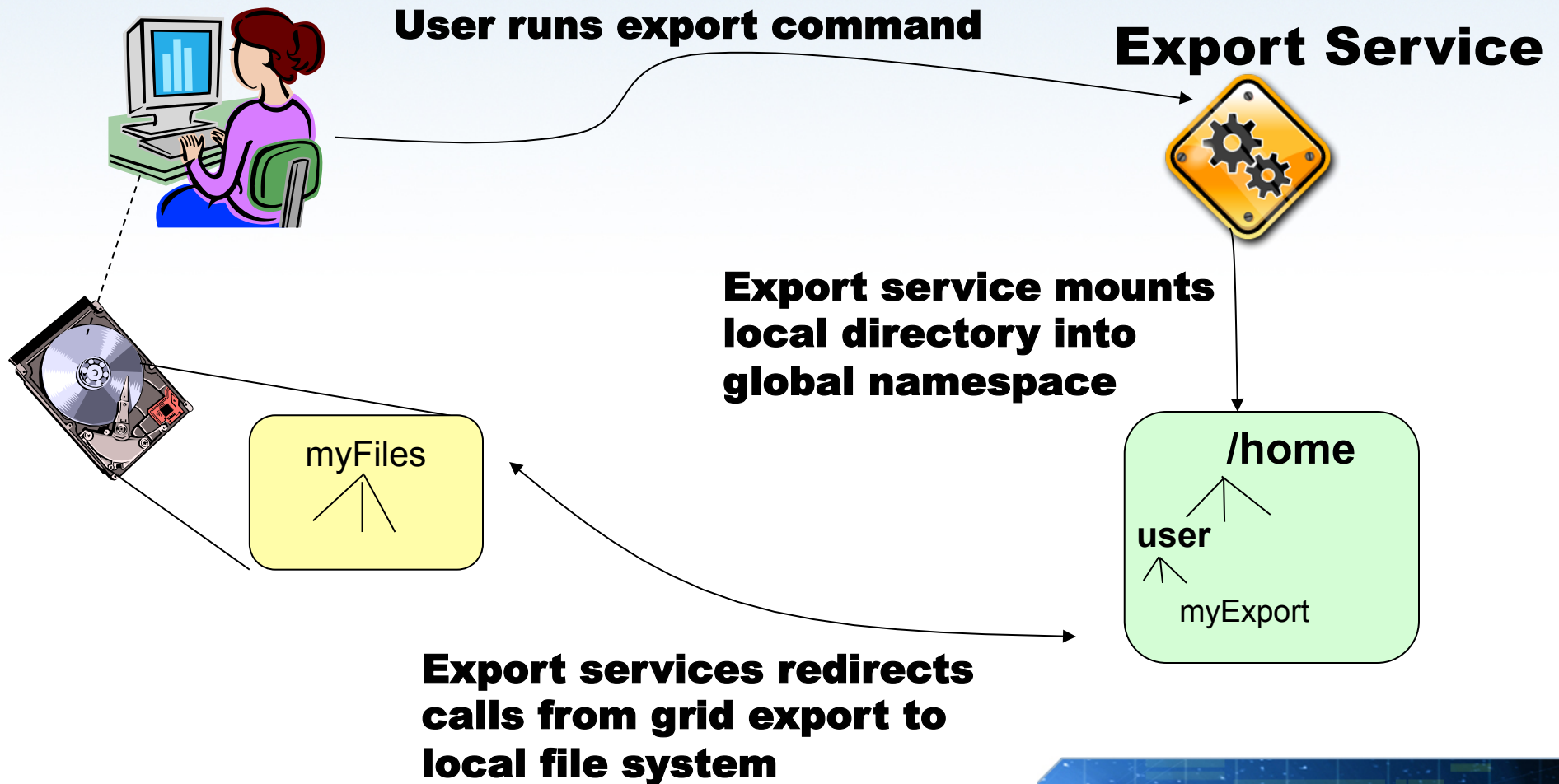
Copying data into the grid

- cp <source> <dest>
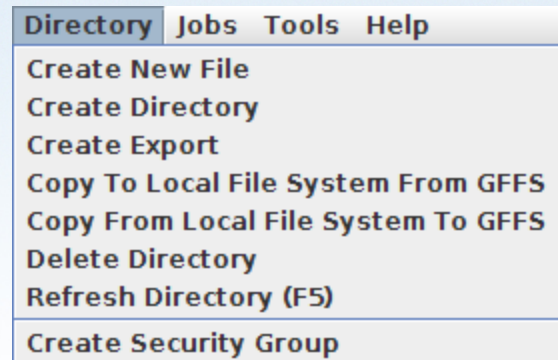
Shared storage as well

- The "rule" is – if you create a file or directory the storage used is in the same storage container as the parent directory
  - For an export this is obvious
- To place data on a remote storage service, mkdir (or use the GUI) and specify the target container. All data going into that directory will be stored on that container

XSEDE

Four Examples Illustrate Revisited

- Directly sharing data with a collaborator at another institution
  - Export directory on campus file server into the GFFS
  - Give your collaborator desired level of access (RWX)
  - Collaborator FUSE mounts the GFFS their desktop
  - Share files.
- Accessing your own data scattered on different resources
  - Export data into your GFFS "home" or other directory
  - FUSE mount and access or use GUI
- Accessing data at an NSF center from a home or campus
  - Export directory at NSF center that you want to access
  - FUSE mount the XSEDE GFFS into your local file system
  - Create, Read, Update, and Delete files at the center from home
- Accessing data on a campus machine from an NSF center
  - Export directory on campus file server into the GFFS
  - FUSE mount the GFFS on the login node at the center, or specify state-in/stage out in a job description
  - Create, Read, Update, and Delete files at home from the center

XSEDE

Execution Management Services

**Launch/monitor jobs from your desktop, run them on *Shared Virtual Compute Facilities*
SVCF
a.k.a. Campus Bridging Use Case #6**

XSEDE

What are Jobs in XSEDE?

- A **job** is a unit of work that executes a program
  - Really pretty generic: much like PBS or LSF job
  - Program may be sequential, threaded, hybrid GPGPU program, or traditional parallel using MPI or OpenMP
  - Programs can be command line programs or shell scripts that take zero or more parameters
- Jobs MAY specify *files* to be staged in before execution and out after execution
  - This MAY include executables and libraries
- Jobs MAY specify *file systems* to mount, e.g., SCRATCH or GFFS (Global Federated File System)
- Jobs MAY specify resource *requirements* such as operating system, amount of memory, number of CPU's, or other matching criteria
- **Jobs MAY be *parameter sweep* jobs with arbitrary number of dimensions**

Creating JSDL Files using the Grid Job Tool

- Manual Creation:
  - Use editor to create XML file
  - Difficult and error-prone due to XML's eccentricities
  - Easiest method: start with existing JSDL and modify (carefully)
- Using Grid Job Tool:
  - GUI builder for JSDL files
  - User describes job in GUI
  - Description can be saved as GridJobTool "project" file
    - edit/re-use project to create new JSDL files
  - Automatically generates XML from user provided description
  - Started with grid command **job-tool**

How to Launch Job Tool from GUI Browser

- Select directory where you want JSDL project file located

  OR

- Select execution container (BES or queue) where you want to execute job
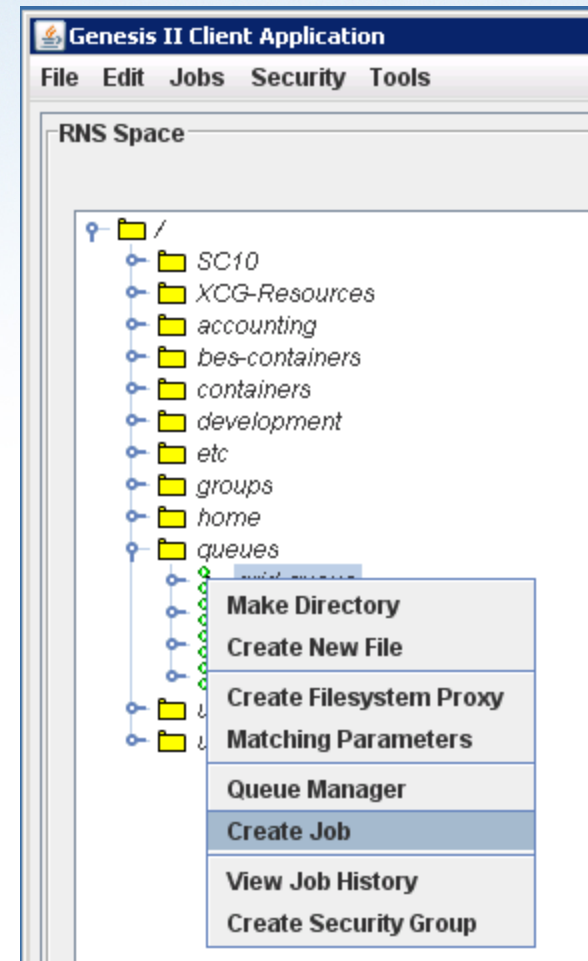
BESes: Basic Execution Services

- BESes run jobs on particular compute resources
  - Manage *data staging* for jobs
  - Monitor job *progress/completion*
  - Maintains *job state*
- "Compute resources" may be workstations, clusters, or supercomputers
- Each BES has a set of resource properties such as operating system, memory, number of cores, etc. that can be used to match jobs to BESes for execution

XSEDE

Grid Queues

- Work much like any other queuing system
- Grid users submit jobs to grid queue
- You can create your own! In XSEDE CB terms, a Shared Virtual Compute Facility
- Maintain:
  - List of (BES) compute resources available for scheduling
  - Description of capabilities of each compute resource
  - List of jobs and statuses
- Match jobs to available compute resources
  - Ask matching resources to run jobs
- Monitor job progress/completion
- Cmd-line and GUI tools to manage jobs in queue
  - qsub, qstat, qkill, qcomplete, queue manager

XSEDE

Run A Remote Job

Grid-Queue

				job4	job3	job2	job1

user submits job
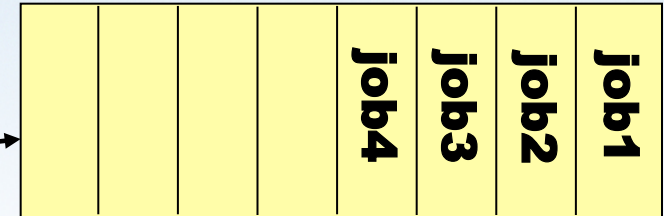
Grid-Queue sends job to matching BES 1

Grid queue interacts with several BESs and matches jobs to BESs. Different resources may use different BES implementations.

UNICORE 6 BES 1

Genesis II BES 2

UNICORE 6 BES 3

BES executes job

XCG
Tutorial
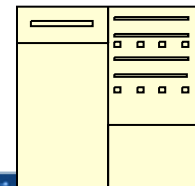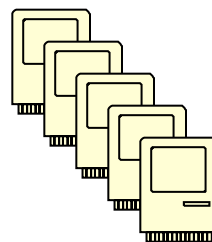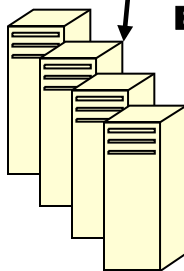
XSEDE

Job Execution – The Working Directory

BES1

User submits job/queue schedules on BES

BES creates unique
working dir for
each job

BES stages data
to/from job working dir
as specified in JSDL

activities

job1 job2 job3

working-dir

my_job_data

runA   runB

XSEDE
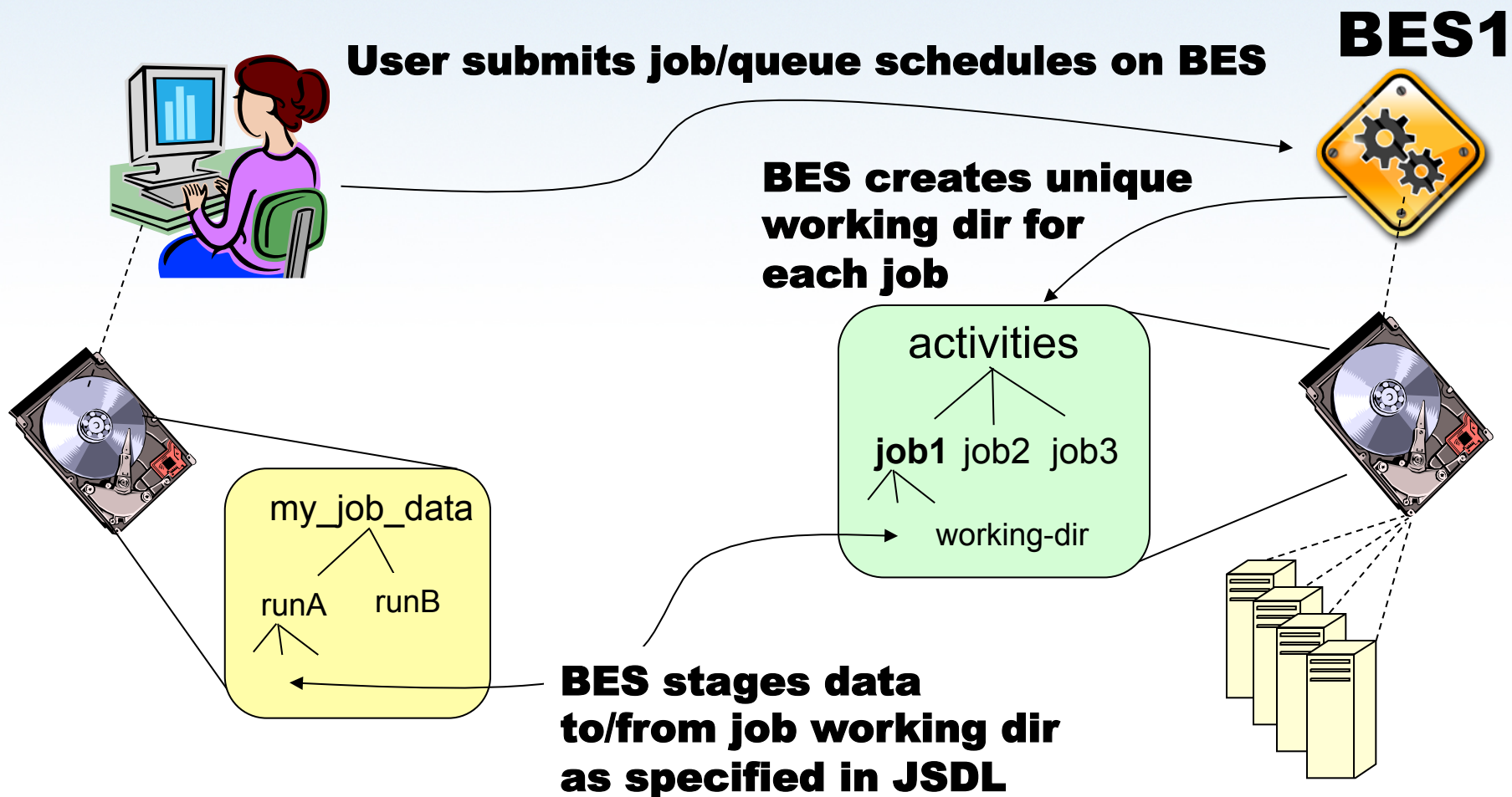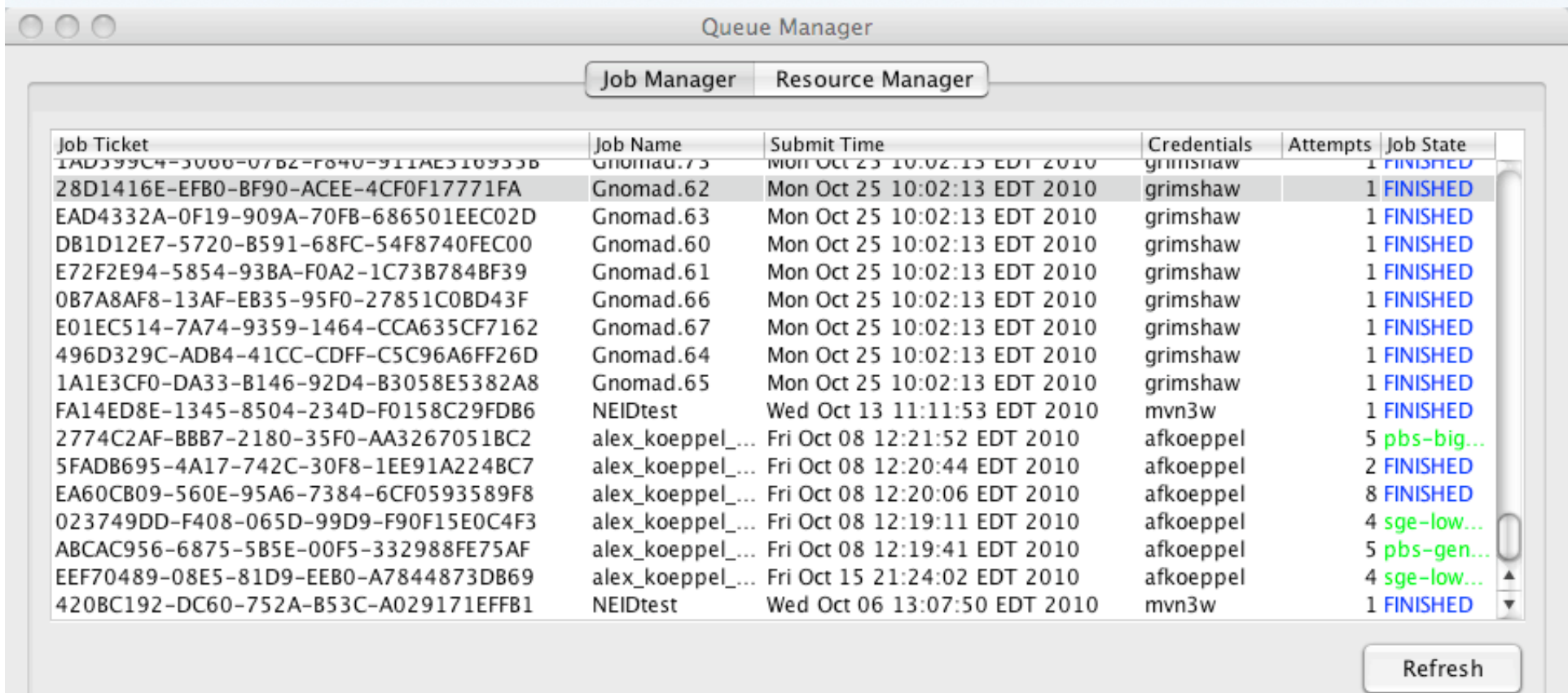
Interact with Jobs via Queue Manager

- You can **stop**, check **status**, examine job **history**, or **reschedule** a job

- You can interact with a job's **working directory** if job is in a running state on a (Genesis II) BES

XSEDE

Grid Queues – GUI Queue Manager

- *Queue Manager* presents information about jobs and resources currently managed by queue

View Job Information in Queue Manager

- Status
    - **QUEUED**: job waiting to be scheduled on BES resources
    - **REQUEUED**: job failed execution at least once and has been automatically re-queued
    - **ERROR**: job failed the maximum allowable execution attempts and will not be re-queued
    - **On <BES name>**: job passed to <BES name> for execution
        - Note: Does not connote status within BES (job may be running, queued, staging data, etc.)
    - **FINISHED**: job executed successfully
- Attempts
    - Number of times queue has tried schedule job for execution
    - Some failures do not increment attempts
        - grid software failures
        - job preempted due to local BES policies
- Ticket
    - Unique ID assigned by grid queue to job on submission

- Queue keeps status of active and completed jobs
    - Jobs in final status (ERROR and FINISHED) need to be cleaned up by user
        qcomplete <queue name> { --all | <job ticket>+ }

Examine Job History in Queue Manager

- Right-clicking on job provides information about job's history in different levels of detail
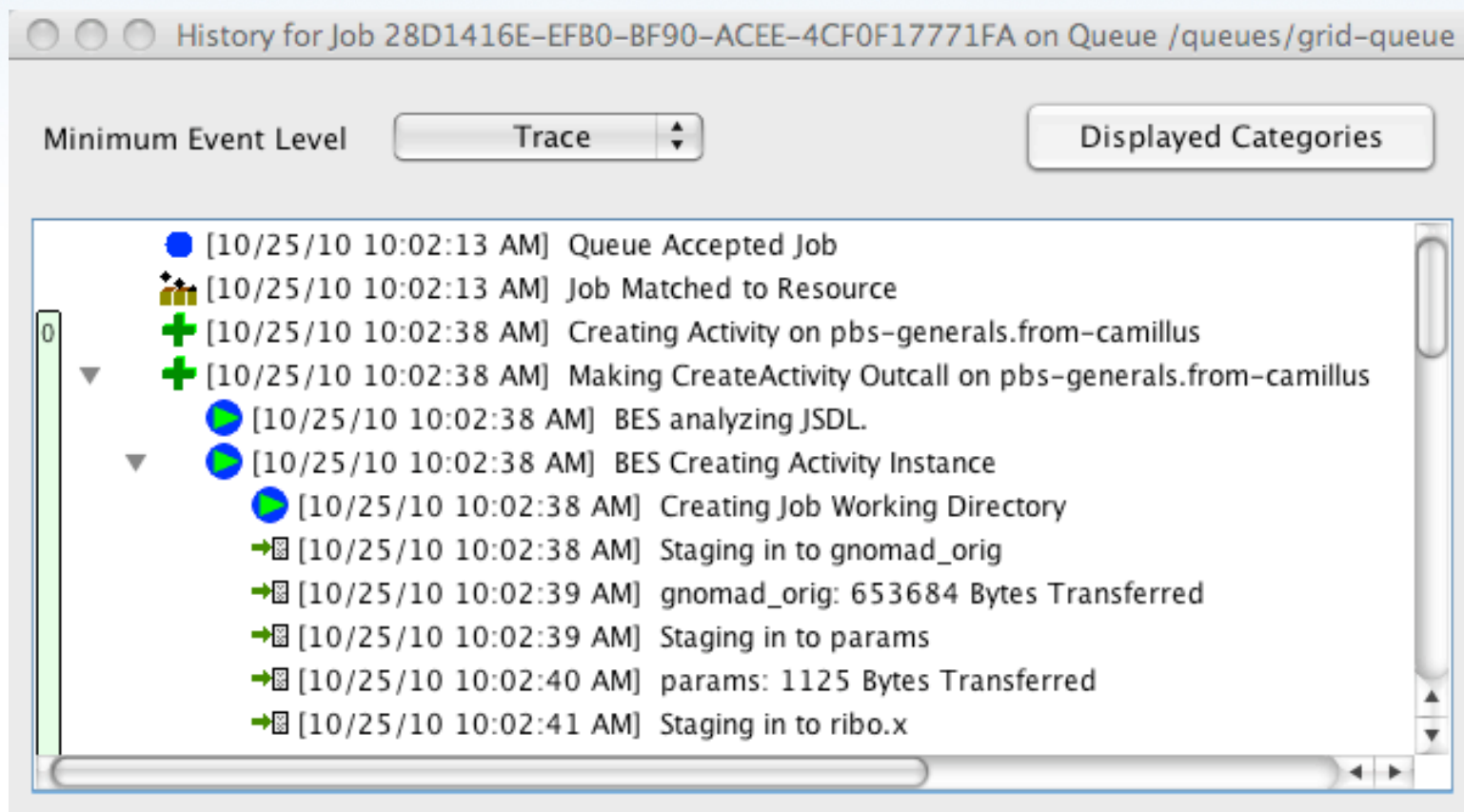
Interact with Job Working Directory

- When using Genesis II BES resources, job working directory is accessible via GFFS
- Working-directory is located in queue where job was submitted at
  <queue-path>/jobs/mine/running
- For each running job, there is a directory with job ticket number with two entries:
  - status
    - file containing state of job (e.g. queued, running)
  - working-dir
    - session execution directory of running job
    - read/write/create/delete files here to interact with running job
- If job was submitted directly to BES, job directory is located at
  <bes-path>/activities

XSEDE

GFFS Wrap up

XSEDE

Extreme Science and Engineering
Discovery Environment

GFFS

- Secure sharing of resources
  - Compute
  - Data
  - Identity
  - Programs/workflows
- Simple, transparent access to remote resources
  - Command line
  - GUI
  - FUSE

XSEDE

More information and help

- http://genesis2.virginia.edu/wiki/Main/HomePage
- xcghelp@cs.virginia.edu
- XSEDE Help Ticket
- Weekly on-line office hours – email xcghelp for times and contact info
- Dan and the rest of the team are willing and able to help you install

Our reach will forever

exceed our grasp, but,

in stretching our horizon,

we forever improve our world.

XSEDE

Extreme Science and Engineering
Discovery Environment

Using the FUSE driver from a workstation in CS at UVA
cat site/1GB > /dev/null

Read					Write			
	PSC	TACC	UVA	LMU	PSC	TACC	UVA	LMU
Avg Time (Seconds)	11.54	36.09	15.52	74.40	78.65	247.34	23.30	200.43
Avg BW in MB/S	88.77	28.37	65.97	13.76	13.02	4.14	43.95	5.11
STD (Seconds)	0.77	8.39	3.43	22.01	48.74	46.28	0.57	76.43
Best MB/S	94.12	39.74	77.69	23.21	23.20	5.77	46.23	6.19

XSEDE

Leave it in the hands of the user!

The principals of distributed computing (Enslow 78 and 81) guided us since we started our work in 1984 to follow an approach that can bring the job management responsibilities (access point) as close as possible to the owner of the jobs.

- Local resource provisioning for job management
- Local policies for provisioning of remote resources
- Local policies for job scheduling
- Local identity management
- Local data access with remote I/O
- Local execution when remote resources are not available

What Did We Learn From Serving a Quarter of a Million Batch Jobs on a Cluster of Privately Owned Workstations

1992

Miron Livny

Computer Sciences Department
University of Wisconsin — Madison
Madison, Wisconsin
{miron@cs.wisc.edu}

User Prospective

- Maximize the capacity of resources accessible via a single interface
- Minimize overhead of accessing remote capacity
- Preserve local computation environment

02/2004

1994 Worldwide Flock of Condors
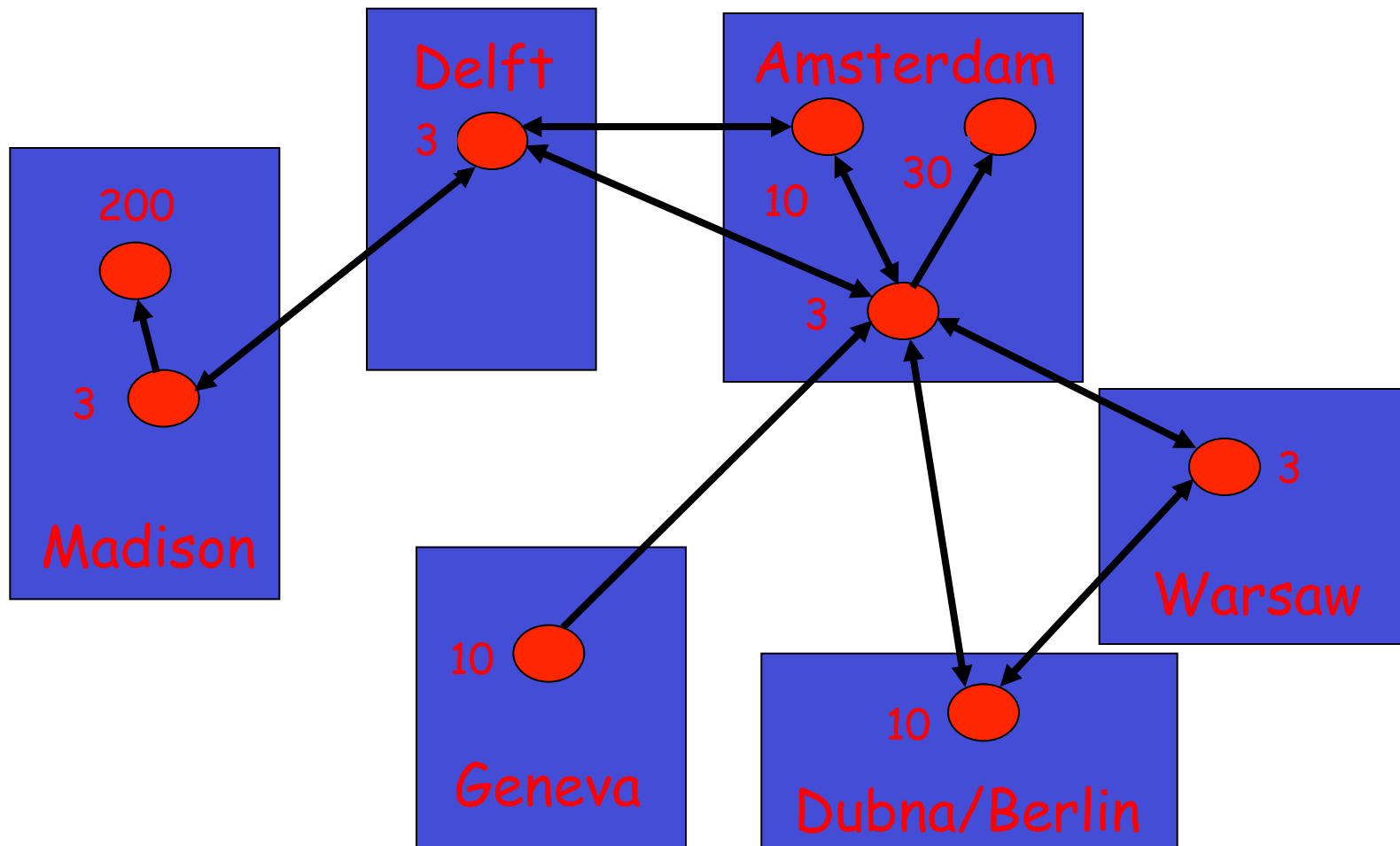
D. H. J Epema, Miron Livny, R. van Dantzig, X. Evers, and Jim Pruyne, "A Worldwide Flock of Condors : Load Sharing among Workstation Clusters" *Journal on Future Generations of Computer Systems*, Volume 12, 1996

Use resource and job management "gateways" to connect the Condor pools.

Established a Peer to Peer relationship between the pools to support full local control.

Followed the routing approach of message passing networks to establish a connection between the source (owner of the work) and the destination (resource).

As many as desired

More than **60** users used on 09/03/15 **21** access points on the UW-Madison campus to perform more than **300K** executions that consumed more than **340K** core hours on the UW-CHTC HTCondor pool.

- This is out of a total of more than **860K** hours provided by more than a dozen HTCondor pools on the campus

- The OSG pool provided more than **75K** core hours

Resource Provisioning
(resource -> (job) manager)
vs.
Work Delegation
(job -> (resource) manager)
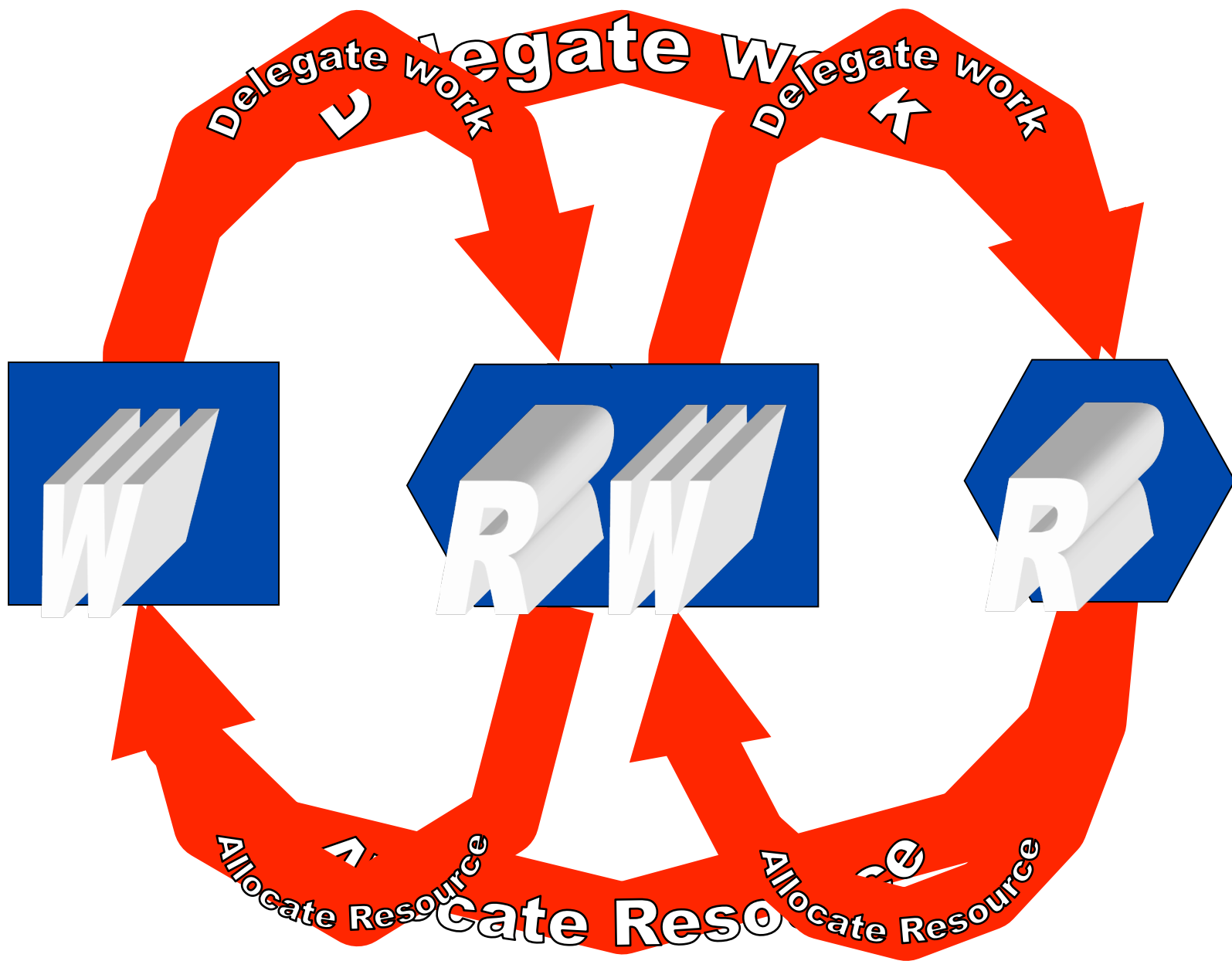
Resource Provisioning

A limited assignment of the "ownership" of a resource

- Owner is charged for allocation regardless of actual consumption
- Owner can provision resource to others
- Owner has the right and means to revoke an allocation
- Allocation is governed by an "agreement" between the client and the owner
- Allocation is a "lease"

Work Delegation

A limited assignment of the responsibility to perform the work

- Delegation involved a definition of these "responsibilities"
- Responsibilities my be further delegated
- Delegation consumes resources
- Delegation is a "lease"

HTCondor uses a two phase matchmaking process to first **provision** a collection of resources to a requestor and then to select a task to be **delegated** for execution within the constraints of these resources

It is all about Sharing

- **Clusters at Universities & National Labs are shared.**
  - Sharing policy is locally controlled. (local autonomy)
  - All owners want to share to maximize the benefit to all. (common goal)

- **Researcher uses a single interface to access local and remote resources …**
  - … they own
  - … others are willing to share
  - … they have an allocation on
  - … they purchase from a commercial (cloud) provider

OSG focuses on making this technically possible for High Throughput Computing applications

1000's of independent researchers

Break the many-to-many relationship into two many-to-few relationships

A single Provisioning System to create community specific overlay batch systems

100's of independent IT infrastructures

VOs are autonomous to schedule the resources provisioned to them. In most cases they use **HTCondor**
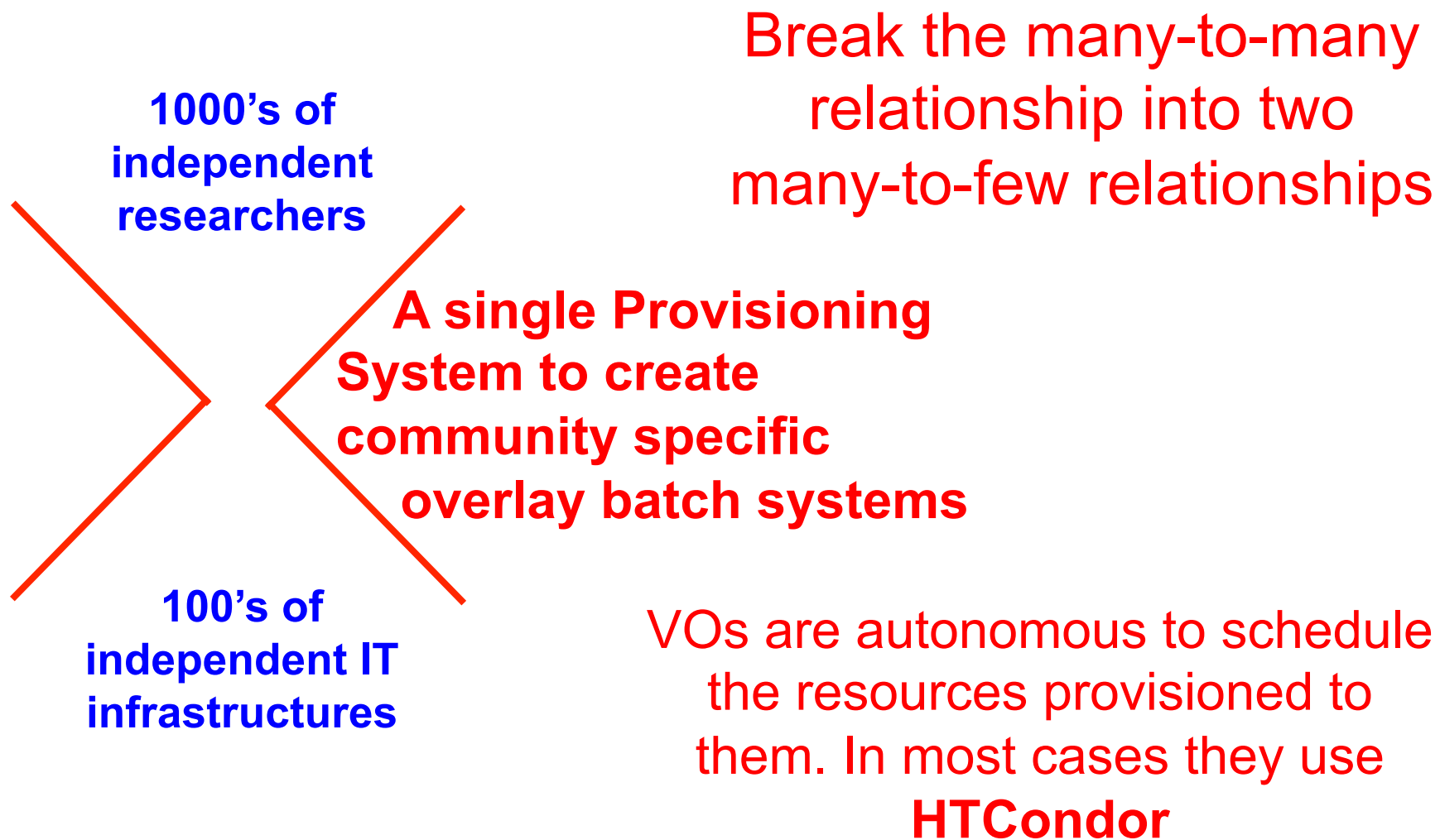
Another OSG Campus was born!

As part of an ongoing collaboration between the **ACI-REF** project and the **OSG** consortium the **Clemson** Campus joined the **OSG** family earlier this summer.

- In the past 30 days, Clemson (Palmetto cluster) contributed 300K opportunistic core hours to the **OSG** community
- The first two Clemson scientists, A.Feltus (Genomics) and J.Tessendorf (Scientific Visualization) used 170k hours on **OSG** last week.

CENTER FOR
HIGH THROUGHPUT
COMPUTING

Open Science Grid

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

Inter-Campus Sharing

In the **last 7** days **OSG** provided more than **414K** of the more than **1,160K** HTC core hours used to research *Genetic Basis of Adaptive Evolution and to perform Analysis of Population Genomic Data* by the group of Assistant Professor John Pool at **University of Wisconsin-Madison**. More than **80K** of these core hours were provided by **Clemson**.

Science beyond Physics since 1/2015

110 Million hours to science other than physics in 2015 so far.

Pie chart segments: 29 other clusters, Syracuse, Caltech, UNL, UCSD, MIT
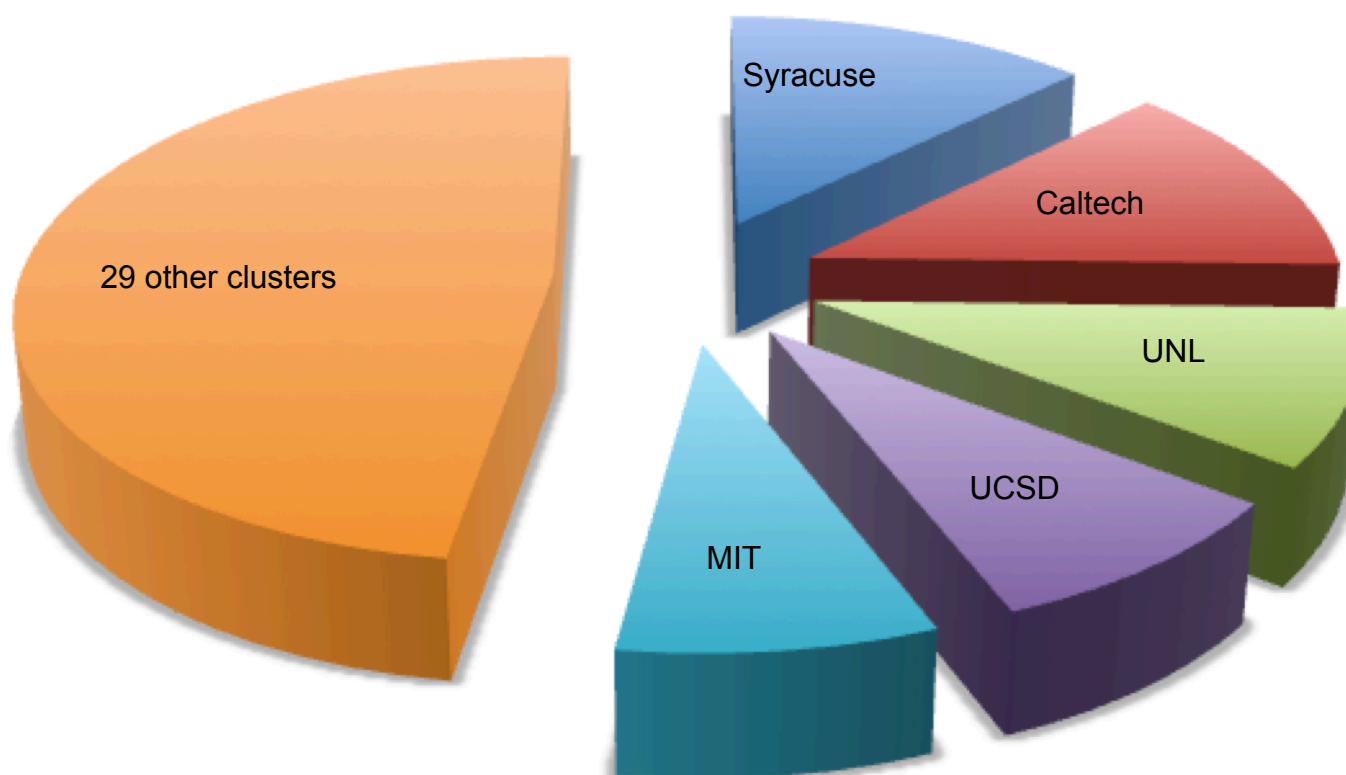
Origin of cycles consumed by Science other than Physics

It is all about Provisioning

- How to express, implement and verify provisioning policies?

- How to monitor consumption rates in "near" real time?

- How to co-provision resources?

- How to debug provisioning systems?

- How to support leasing of resources?

You may have ONE local Access Point managing the execution 100K jobs on 10K remote resources

"Submit Locally" is not limited to hardware and software resources, it also includes human resources.

Three years ago, we switched from an "engagement" or "user support" model to a "facilitation" model. Now, when you submit locally at UW-Madison you are guided by local Research Computing (RC) Facilitators who understand your research problem and the computing capabilities of the local access point.

Facilitating for Global Execution

- Software portability – Making the application run anywhere.

- Understanding resource requirements

- Understanding data needs

- Understanding the potential damage 1M jobs can cause

" ... Since the early days of mankind the primary motivation for the establishment of *communities* has been the idea that by being part of an organized group the capabilities of an individual are improved. The great progress in the area of inter-computer communication led to the development of means by which stand-alone processing sub-systems can be integrated into multi-computer '*communities*'. ... "

Miron Livny, " *Study of Load Balancing Algorithms for Decentralized Distributed Processing Systems*.",
Ph.D thesis, July 1983.

We view users not only as consumers of resources but also as providers of resources

"here is my workload and here are my resources!"

Building Bridges from the Campus to XSEDE

Lee Liming
University of Chicago
5735 South Ellis Avenue
Chicago, IL 60637
+1 (505) 899-4098
lliming@uchicago.edu

Ian Foster
University of Chicago
5735 South Ellis Avenue
Chicago, IL 60637
+1 (630) 252-4619
foster@uchicago.edu

Steven Tuecke
University of Chicago
5735 South Ellis Avenue
Chicago, IL 60637
tuecke@uchicago.edu

ABSTRACT

XSEDE is the integration framework for national-scale, public HPC resources in the United States. XSEDE is used by thousands of researchers at hundreds of college and university campuses throughout the country, as well as many international collaborators. Over the past several program years, XSEDE has redefined its identity management, security, and service interfaces to bridge the gap between national-scale HPC resources and campus-based computing resources. These changes make it easier for research performed on campus to access our national computing resources and make them a part of the everyday research process. We report here on XSEDE's new identity management system and how it provides a smooth bridge between campus and national identity systems. We also describe how this federated security system supports two additional bridges between campuses and national HPC services, one involving data movement and another involving scientific workflows.

1. ENVISIONING BRIDGES TO XSEDE

In 2012, the XSEDE project [1] undertook a focused effort to document the needs of its user community in a wide set of areas. One of these areas was campus bridging. The resulting description of campus bridging use cases [2] listed the following as the most useful "bridges" between campus and XSEDE systems.

1. XSEDE systems and services should allow use of campus identity credentials by supporting federated identity and authorization mechanisms.

2. XSEDE should help campus system administrators create XSEDE-like environments on campus systems to smooth their users' transitions between campus and national systems.

3. XSEDE services should provide "remote desktop" access so that users on campuses can remotely view graphical displays generated by XSEDE systems.

4. XSEDE should enable users on campuses to conduct integrated data analysis that includes data on both campus and XSEDE systems.

5. XSEDE should enable users on campuses to initiate automated workflows (scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services.

The second use case (enabling campuses to make their systems "look more like XSEDE systems") is speculative: it is unclear whether campus IT administrators want to do this, and it is uncertain that XSEDE's environment would be a good fit for campus systems. Nevertheless, XSEDE makes heavy use of open source and free-license software components, and the processes that XSEDE service providers use to create commonality across their systems are open and freely available to the public. We believe that existing documentation for XSEDE systems and services is sufficient to allow anyone to replicate the most common elements of the XSEDE system environment, should they so choose.

The third use case (remote desktop access) is highly specific to the type of resource being accessed. While important in many cases, it is not universally appropriate for all services. Therefore, each service provider must design and implement the mechanism in a manner that best serves the intended purpose of their service.

The remaining three use cases—the first, fourth, and fifth—are amenable to, and in some cases require, solutions that are XSEDE-wide in nature. The rest of this paper focuses on these three use cases.

2. SURVEYING THE SITE

The first step in any construction project (for bridges or anything else) is to survey the existing site and structures and identify the pieces that must change. In 2013 and 2014, XSEDE's Architecture and Design (A&D) team was charged with documenting the existing XSEDE system as the basis for expansion plans. The authors of this paper, along with Andrew Grimshaw (University of Virginia) and Morris Riedel (Juelich Supercomputing Centre), performed this work. The output of this activity is contained in two technical reports: a brief high-level architectural overview of the XSEDE system [3] and a lengthy inventory of the system's detailed architecture and implementation [4].

The XSEDE system has one basic purpose: it allows researchers and scientists to form a relationship with the collection of national services that XSEDE provides and to maintain that relationship over time as their data and computing needs change and as the services themselves change. Six core functions support this basic purpose: identity management, interactive login, accessing remote files, submitting and managing computations, transferring datasets, and discovering and providing resource information. These six core functions are shown as the vertical columns in Figure 1.

Because XSEDE serves many different kinds of researchers and scientists (individuals, small teams, national or international collaborations) and supports several different modes of interaction (single project use, program-wide use, open-ended partnerships), the architecture offers three distinct integration approaches: the access layer (abstract user and developer interfaces for XSEDE's core functions), the service layer (abstract protocol-based interfaces for similar classes of XSEDE

services), and the resource layer (direct access to individual, specialized XSEDE systems and services). These three approaches to integration are shown as the horizontal layers in Figure 1.
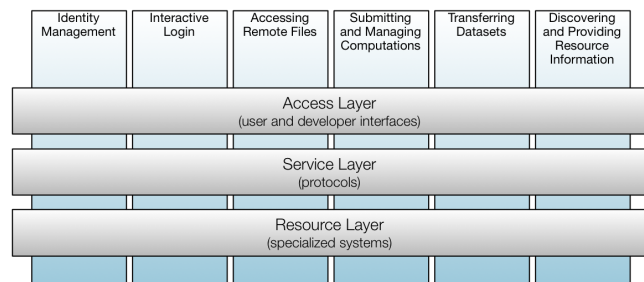

Figure 1: XSEDE core functions and architecture layers

As Figure 1 shows, each of the six core system functions can be accessed via any of the three integration layers, depending on the purpose of the integration and the nature of the relationship between the integrators and XSEDE. The access layer is intended for end users and application developers and provides software APIs (libraries and classes), Web interfaces (REST APIs), and command-line interfaces (CLIs). The service layer is for application developers and long-term partners and their system builders (including campus IT systems), and its interfaces are in the form of standardized network protocols. The resource layer is for short-term, specialized uses, and its interfaces are highly specific to each XSEDE service. Because the primary XSEDE services provide high-performance computing, the interfaces in the resource layer tend to be Unix-based command-line interfaces.

In this section, we provided a brief overview of the initial state of the system prior to our bridge-building projects. The next three sections describe the plans for each of the three bridges identified in §1.

3. BRIDGING CAMPUS AND NATIONAL IDENTITIES

The first bridge needed between campuses and XSEDE is a way for campus IT administrators to allow their users to use campus credentials to access XSEDE services. This bridge corresponds directly to the first XSEDE core function: identity management.

XSEDE's baseline system (inherited from the previous program, TeraGrid [5]) provided identity management via two key mechanisms: an XSEDE-specific userid and password, and an XSEDE-specific X.509 certificate system. The former allowed users to register as XSEDE users and access XSEDE's Web user portal. The second allowed users to obtain and use short-term X.509 digital certificates for single sign-on access to XSEDE services.

While reasonably robust, this baseline system had numerous deficiencies. The tools provided were specific to XSEDE and costly to develop and maintain, and the support across XSEDE services was uneven. Furthermore, X.509 has not achieved widespread adoption by campuses, so it could not provide a smooth bridge between campuses and XSEDE. It was clear that in order to fully support the first campus bridging need, we would need to redefine XSEDE's identity management interfaces. This task became one of XSEDE's primary development activities in 2014 and 2015. (The XSEDE project is intentionally not focused on development, but rather on

operation of the existing system. Resources for development are strictly controlled by the project management. Development resources are provided for only the most critical needs.)

The route that this bridge must take had been mapped out by several earlier efforts: notably, the activities of the NSF Advisory Committee for Cyberinfrastructure Task Force on Campus Bridging [6], the formation of the InCommon community [7], and a prototyping activity in TeraGrid [8]. From these efforts it was clear that a key element would be support for OAuth (now OAuth2), the mechanism for federated authorization used widely on the Internet in both academia (InCommon) and public enterprise (e.g., Facebook, Google) [9]. Supporting OAuth2 would create a smoother bridge between XSEDE and campus systems.

OAuth2, however, is an abstract framework for authorization. It does not specify many of the details necessary to build a complete solution for XSEDE and campuses. Ultimately, XSEDE was able to form a consensus around a design that built on OAuth2 to provide the following elements.

Every XSEDE user initially registers with the XSEDE User Portal (XUP), a website accessible via any standard Web browser [10]. Users may identify themselves to XUP via any of a wide set of identity providers, including InCommon campuses and other OAuth2-based systems. (Users without access to such systems—or who choose not to use them—can register without a pre-existing identity.) Registration consists of creating an XSEDE user ID (unique for each XSEDE user) and an XSEDE password. Once registered, the user may link his/her XSEDE identity to others, such as a campus identity. This linking mechanism is based on OAuth2. Once identities are linked, the user may use any linked identity to login to XUP and gain access to XSEDE system-wide functions.

Authorization to use specific XSEDE services (e.g., a supercomputer) is granted by the operators of individual XSEDE resources. (Merely having an XSEDE identity does not allow the user to do much.) Before a user is authorized to use a national HPC system, XSEDE account management staff independently verify the user's identity. XUP provides the user interface for requesting authorization to use XSEDE resources and for managing access to the resulting "allocations."

XSEDE provides several interfaces for translating the user's OAuth2 token (obtained at XUP login) into a different kind of credential understood by a specific service interface. For example, the MyProxy interface [11] allows creation of a short-term X.509 certificate used by some legacy TeraGrid interfaces; the WS-Trust STS interface [12] allows creation of a signed SAML chain used by XSEDE's Web services-style interfaces.

Application and service developers are working on campus bridging, science gateways, connecting instruments, etc., may require CLI, GUI, or API access to XSEDE's identity services. XSEDE provides all of these mechanisms via its access layer.

Finally, for application developers and system integrators, XSEDE offers a range of APIs for identity and group management, including:

OAuth2: A widely used interface for sharing user identity and authorization information between systems.

OpenID Connect: Another widely used interface for sharing user profile information (name, email address, etc.) based on OAuth2 authorizations [13].

Globus Auth API: The interfaces used for XSEDE's user management functions, most especially for managing links between identities.

WS-Trust STS: A standard Web services interface for translating one set of credentials into another set based on access rules.

MyProxy: A legacy interface used to obtain an X.509 credential that can be used to access some XSEDE system-wide services.

The public rollout of these new interfaces in support of the "campus identity management" bridge is scheduled to be completed by the end of 2015.

4. THE DATA ANALYSIS BRIDGE

The fourth type of bridge identified in §1 concerns integrated analysis of both data on campus systems and data on XSEDE systems. The challenges that arise when trying to do this work include:

- Navigating XSEDE security mechanisms

- Moving the data from campus systems to XSEDE (or vice versa)

- Moving the results from XSEDE to campus (or vice versa)

- Keeping track of which portions of the data have and have not been moved

The first challenge is related closely to the campus identity management bridge discussed in the preceding section, and in fact, we believe that the identity management solution described there is also the solution to this part of the data analysis challenge.

The second and third challenges involve moving data between campus systems and XSEDE systems. This task can be straightforward if the data is small and in only a few files, but is often enormously difficult because the data is large (>100GB) and/or in many files. This campus bridging challenge was the topic of an earlier paper from 2012 [14]. In that paper, we reported on the use of Globus software-as-a-service to simplify the campus-to-XSEDE data analysis bridge. We will not repeat that material here, but note that the addition of the new identity management bridge simplifies the previous one even further, because now all XSEDE users automatically have the credentials needed to authenticate to Globus. There is no longer any need to maintain a separate Globus account or to login separately to Globus and XSEDE to use the methods described in that paper.

The fourth challenge is an important twist on this problem: even if it is easy to move data from campus to XSEDE and back, how can one keep track of what data is where? Without assistance from the system, a researcher or research assistant could spend hours, even days, checking the status of hundreds or thousands of files in a large dataset. Rather than repeating the entire description of how Globus solves the data movement problem, we simply note that Globus services include a synchronization feature that allows users to specify that a complete copy of all files from a given source should be created on the destination, with the understanding that only files that are not already at the destination will be copied. This capability makes it easy for researchers to assure themselves that all of the data that they expect to be on campus (or on XSEDE systems) really is there, without painstakingly inspecting hundreds or thousands of files.

5. THE WORKFLOW BRIDGE

The fifth type of bridge identified in §1 concerns automated workflows (i.e., a scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services. The challenges that arise when trying to do this work include:

- Navigating XSEDE security mechanisms

- Moving data as part of file management tasks

- Allowing computation tasks to access remote data when moving the data is more costly than accessing it remotely

- Controlling the execution of computation tasks on campus and XSEDE systems

As in the preceding section, the first challenge is a restatement of the campus identity management bridging issue, and we refer to §3 for the solution. The second challenge is a restatement of the data analysis bridging issue, and we refer to §4 for the solution.

The third challenge arises when the scientific software used in the workflow expects the input or output data to be locally available (on a storage system connected to the system running the software) and it is time-consuming to make a local copy available. For example, the file may be huge (>100GB) and only part of it is needed. Or, there may not be sufficient local storage to make a local copy. In such cases, it is preferable to create a virtual file system interface [15] to access the data. To the scientific application, it appears that the data is on the local system, when in fact it is on the other end of a network connection.

To overcome this challenge, part of XSEDE's "campus workflow bridge" is a user-initiated virtual filesystem. XSEDE provides users with a thick-client interface (downloadable software) called Genesis II [16], which includes a virtual filesystem, GFFS [4 §3.4.1]. The user installs the software where the data resides (on campus or on XSEDE systems) and exports the data into the virtual filesystem, which is accessible only to themselves and anyone they choose to authorize. The science code used in the workflow sees this filesystem—and the data exported to it—as a local filesystem.

The fourth challenge listed above presents the need for a consistent interface for controlling computational tasks on campus and XSEDE systems: an interface that can be used by software rather than by humans. Specifically, the user's workflow system needs a way to remotely control tasks on campus and XSEDE computation services.

A second part of XSEDE's campus workflow bridge is therefor a remote computation interface. Like the GFFS, this interface is provided as a thick client and is also part of Genesis II. This interface accesses the UNICORE service [17] provided by each XSEDE computation service (via protocols specified in XSEDE's service layer) to initiate, queue, execute, interrupt (if necessary), and obtain the results from computational tasks on the compute service. Many workflow systems in use today have

UNICORE drivers that allow the system to access UNICORE task management services.

In summary, XSEDE's campus workflow bridge includes support for campus credentials when authenticating to XSEDE, Globus services for moving data between campus and XSEDE, Genesis II's GFFS for providing a virtual filesystem interface when necessary, and Genesis II's UNICORE client interface for remote task management on XSEDE compute services.

6. CONCLUSIONS

During its five-year program cycle, XSEDE has constructed several new and improved bridges between campus IT systems and XSEDE's portfolio of national services. The lynchpin of this work was restructuring XSEDE's identity management function to support modern federated identity and authorization mechanisms. In particular, basing XSEDE's identity management function on OAuth2—with additional features from OpenID Connect, Globus Auth, WS-Trust STS, and MyProxy—has made it possible for campus-based researchers to access national HPC services using their campus credentials. To enable this access, campus ID administrators can join the InCommon federation or provide their own OAuth2-based campus identity and authorization system. Once enabled, researchers on campus have easier access to integrated campus and national services for data analysis and scientific workflow.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] J Towns, T Cockerill, M Dahan, I Foster, K Gaither, A Grimshaw, V Hazlewood, S Lathrop, D Lifka, GD Peterson, R Roskies, JR Scott, N Wilkins-Diehr, "XSEDE: Accelerating Scientific Discovery", Computing in Science & Engineering, vol.16, no. 5, pp. 62-74, Sept.-Oct. 2014, doi:10.1109/MCSE.2014.80

[2] Campus Bridging Use Cases. Technical Report, April 25, 2013. (http://hdl.handle.net/2142/43882)

[3] XSEDE Architecture Overview. F Bachmann, I Foster, A Grimshaw, D Lifka, L Liming, M Riedel, S Tuecke, Technical Report, September 1, 2014. (http://hdl.handle.net/2142/50274)

[4] XSEDE Architecture Level 3 Decomposition. Technical Report, July 14, 2013. (http://hdl.handle.net/2142/45117)

[5] C Catlett, et al, "TeraGrid: Analysis of Organization, System Architecture, and Middleware Enabling New Types of Applications." In High Performance Computing and Grids in Action, L Grandinetti (ed), IOS Press, 2008.

[6] NSF Advisory Committee for Cyberinfrastructure Task Force on Campus Bridging. Final Report. March 2011. (http://www.nsf.gov/od/oci/taskforces/TaskForceReport_CampusBridging.pdf) Also available as print-on-demand book from: https://www.createspace.com/3597300.

[7] Barnett, W., V. Welch, A. Walsh and C.A. Stewart. "A Roadmap for Using NSF Cyberinfrastructure with InCommon." 2011. (http://hdl.handle.net/2022/13024) Also available as print-on-demand book from https://www.createspace.com/3630011.

[8] J Basney, T Fleury, and V Welch, "Federated Login to TeraGrid," 9th Symposium on Identity and Trust on the Internet (IDtrust 2010), Gaithersburg, MD, April 2010. (http://dx.doi.org/10.1145/1750389.1750391)

[9] Hardt, D., Ed., "The OAuth 2.0 Authorization Framework," IETF RFC 6749, October 2012. (http://tools.ietf.org/html/rfc6749)

[10] https://portal.xsede.org/.

[11] Basney, J., Humphrey, M., and Welch, V., "The MyProxy online credential repository." Softw. Pract. Exper. 35, 9 (July 2005), 801-816. (DOI=10.1002/spe.v35:9; http://web-test.ncsa.illinois.edu/~jbasney/myproxy-spe.pdf)

[12] "WS-Trust 1.4." 25 April 2012. OASIS standard incorporating approved errata. (http://docs.oasis-open.org/ws-sx/ws-trust/v1.4/errata01/os/ws-trust-1.4-errata01-os-complete.html)

[13] "OpenID Connect Core 1.0." 8 November 2014. OpenID standard incorporating approved errata. (http://openid.net/specs/openid-connect-core-1_0.html)

[14] I Foster, R Kettimuthu, S Martin, S Tuecke, D Milroy, B Palen, T Hauser, and J Braden. "Campus bridging made easy via Globus services." In Proceedings of the 1st Conference of the Extreme Science and Engineering Discovery Environment: Bridging from the eXtreme to the campus and beyond (XSEDE '12). ACM, New York, NY, USA, 2012, Article 50, 8 pages. (DOI=10.1145/2335755.2335847; http://doi.acm.org/10.1145/2335755.2335847)

[15] "Virtual file system." Wikipedia, retrieved on June 19, 2015. (https://en.wikipedia.org/wiki/Virtual_file_system)

[16] Morgan, M.M; Grimshaw, A.S., "Genesis II - Standards Based Grid Computing," Cluster Computing and the Grid, 2007. CCGRID 2007. Seventh IEEE International Symposium on , vol., no., pp.611,618, 14-17 May 2007. (DOI=10.1109/CCGRID.2007.53)

[17] D Snelling. "UNICORE and the Open Grid Services Architecture." In F. Berman, G. Fox, and T. Hey, editor, Grid Computing: Making The Global Infrastructure a Reality, pages 701–712. John Wiley & Sons, 2003.

XCBC and XNIT – tools for cluster implementation and management in research and training

Jeremy Fischer
IU Pervasive
Technology Institute
jeremy@iu.edu

Richard Knepper
IU Pervasive
Technology Institute
rknepper@iu.edu

Eric Coulter
IU Pervasive
Technology Institute
jecoulte@iu.edu

Charles Peck
Cluster Computer
Group
Earlham College
charliep@cs.earlham.
edu

Craig A. Stewart
IU Pervasive
Technology Institute
stewart@iu.edu

ABSTRACT

The Extreme Science and Engineering Discovery Environment has created a suite of software designed to facilitate the local management of computer clusters for scientific research and integration of such clusters with the US open research national cyberinfrastructure. This suite of software is distributed in two ways. One distribution is called the XSEDE-compatible basic cluster (XCBC), a Rocks Roll that does an "all at once, from scratch" installation of core components. The other distribution is called the XSEDE National Integration Toolkit (XNIT), so that specific tools can be downloaded and installed in portions as appropriate on existing clusters. In this paper, we describe the software included in XCBC and XNIT, and examine the use of XCBC installed on the LittleFe cluster design created by the Earlham College Cluster Computing Group as a teaching tool to show the deployment of XCBC from Rocks. In addition, the demonstration of the commercial Limulus HPC200 Deskside Cluster solution is shown as a viable, off-the-shelf cluster that can be adapted to become an XSEDE-like cluster through the use of the XNIT repository. We demonstrate that both approaches to cluster management – use of SCBC to build clusters from scratch and use of XNIT to expand capabilities of existing clusters – aid cluster administrators in administering clusters that are valuable locally and facilitate integration and interoperability of campus clusters with national cyberinfrastructure. We also demonstrate that very economical clusters can be useful tools in education and research.

Categories and Subject Descriptors

Theory of computation - Parallel computing models; Computer systems organization - Grid computing; Computer systems organization - Special purpose systems

Keywords

Cluster Management, computational science education, systems administration education, campus bridging, sysadmin training, cluster training, cluster system software, cluster packaging, open source clusters, XSEDE, Rocks, rolls, cluster, Linux, RedHat, CentOS, rpm, yum, campus, bridging, research, LittleFe

1. INTRODUCTION

An NSF Advisory Committee for Cyberinfrastructure taskforce found in 2011 that [2]

The current state of cyberinfrastructure software and current levels of expert support for use of cyberinfrastructure create barriers in use of the many and varied campus and national cyberinfrastructure facilities. These barriers prevent the US open science and engineering research community from using the existing, open US cyberinfrastructure as effectively and efficiently as possible.

For many researchers in computationally-oriented science, those barriers are due to the state of software running on their local clusters. Since the NSF ACCI taskforce came out in 2011, demand for computational resources has continued to grow, and federal investment in cyberinfrastructure for open research has stagnated. This means that campus-based resources are now even more important to advances in open science research than when the report was finalized in 2011. The scale of the problems identified by the taskforce in 2011 are, barring improvements in the way cyberinfrastructure software is developed, going to be more of a challenge as demand continues to outstrip the pace of federally-funded resources [3].

Clusters are often set up in ways that are just good enough to get the job done, and often not in ways that are optimal for enabling discovery. Such clusters are generally administered by people with too much to do and too little time. Maintenance of clusters is often an ongoing challenge. For practicing scientists, differences in the way clusters are set up is often a source of frustration and time lost reading documentation, simply to figure out how to run a particular application or submit a job on a particular cluster.

The Extreme Science and Engineering Discovery Environment (XSEDE) Campus Bridging group has developed the concept of an "XSEDE-compatible basic cluster" (XCBC) build [4, 5] – a build that enables the creation of a cluster entirely from open-source tools modeled after the clusters supported by XSEDE. XSEDE supports a large group of clusters and supercomputers – more than a dozen clusters and more than 13 PetaFLOPS of computational capacity [6], used by thousands of researchers in the US and supported by more than 250 FTEs (Full Time Equivalents) of professional and academic staff [7]. In the absence of any single model for setting up clusters, using the clusters in this major national cyberinfrastructure (CI) facility as a model can create consistency that helps researchers.

XCBC builds on and currently depends on the very successful Rocks project [8]. Early reaction to XCBC was very positive but there were two very clear community reactions for capabilities not supported in the initial Rocks-based implementation of XCBC. First, many members of the community wanted to be able to add specific software tools and capabilities consistent with XSEDE clusters without starting from scratch. Second, many members of the US research and cyberinfrastructure communities wanted the ability to add into the standard XCBC build tools that were needed by the community but not necessarily a part of the basic software installation on a typical XSEDE-supported cluster. These viewpoints were expressed particularly strongly by two very important groups of community representatives: XSEDE Campus Champions and participants in the ACI-REF (Advanced

Cyberinfrastructure – Research and Education Facilitators) project. The XSEDE Campus Champions serve on their own campuses as local experts in national cyberinfrastructure and within XSEDE as experts on campus computing needs [9]. There are more than 250 individuals at more than 200 institutions of higher education, representing every state in the US and the District of Columbia, Puerto Rico, and the US Virgin Islands. ACI-REF is a national consortium of institutions of higher education "dedicated to forging a nationwide alliance of educators to empower local campus researchers to be more effective users of advanced cyberinfrastructure" [10].

As a result of community input, we created the XSEDE National Integration Toolkit (XNIT). XNIT is based on the Yum repository for installation or updates of RPMs [11, 12]. XNIT includes all of the software included in the standard XCBC build, and more. A cluster administrator may benefit by using the XNIT Yum repository simply as a source for updates of allocations software of interest to a given researcher. XNIT and the Yum repository make it easy for campus cluster administrators to do one-time installations of any particular software capability they want within the suite of the XNIT set, and to subscribe if they wish to automatically be notified of updates to particular packages. XNIT also includes software not included in the basic XCBC build – this will be increased over time in response to community requests.

XCBC and XNIT enable economies of scale in cluster administration and in user support because many of the documents and user training materials prepared for XSEDE can be repurposed and reused to support a campus-based cluster. The commands used to execute open-source applications on any cluster created with XCBC or XNIT are compatible with the way these commands are used on a typical cluster supported by XSEDE. A user's knowledge of software, system commands, etc., becomes portable from one cluster built with XCBC to another, and to XSEDE-supported clusters generally. This makes it easier for a researcher to move from an XCBC- or XNIT-based campus cluster to an XSEDE-supported resource available to the national open research community, and from one cluster to another. XNIT in particular enables such compatibility to be added to an existing, operating cluster in part or in whole, without changing the pre-existing cluster setup.

In this paper, we describe the current state of the tools included in XCBC and XNIT. We provide an overview of current cluster installations using XCBC and XNIT for research. We also describe at length the use of XCBC in training for systems administrators, using two modestly-priced and luggable clusters: LittleFe and Limulus HPC200. LittleFe ("little iron") is a popular, inexpensive, build-it-yourself option for building a small (6-node) cluster for training or production. The Limulus HPC200 is a "cluster in a deskside case" available commercially and intended to support individual computational scientists. Finally, we discuss the suitability of LittleFe and Limulus HPC200 as personal deskside clusters.

Our purposes for presenting this material are to: provide cluster administrators with information about the utility of XCBC and

XNIT so that they can consider adopting these tools, provide information on tools that ease cluster administration, inform cluster administrators about the XNIT Yum repository as a source of RPMs for dozens of useful software packages, and provide information that will enable better training for current and future cluster administrators. The set of tools we describe here can, if adoption continues and grows, help improve the ability of US researchers to make new discoveries by enabling a more consistent national cyberinfrastructure. Ultimately, our goal in this work is to ease use of campus cyberinfrastructure for practicing scientists and students throughout the US and beyond, and to simplify migration between campus and national cyberinfrastructure, such as the facilities supported by XSEDE.

2. XCBC AND XNIT BUILD CONTENTS

The XCBC and XNIT basic builds include a set of open-source tools sufficient to set up and operate a cluster from scratch. Full listings of the software tools available within XCBC and the XNIT Yum repository are available online [13]. Highlights of the software included in the current XCBC release (0.9) are described in Table 1 and Table 2. Table 2 focuses on applications that are kept consistent with versions in use on XSEDE (using the current Stampede system [14] as the definition of "current best practices" for XSEDE clusters). In particular, libraries are in the same place as on XSEDE clusters, versions are the same, and commands work as they do on XSEDE-supported clusters.

Table 1. Components of current XCBC build Part 1 – General cluster setup (mostly existing Rocks optional rolls)

Category	Specific packages
Basics	Rocks 6.1.1, Centos 6.5, modules, apache-ant, fdepend, gmake, gnu-make scons
Job Management	Torque, SLURM, sge (choose one)
Rocks optional rolls	
area51	Security-related packages for analyzing the integrity of files and the kernel
bio	Bioinformatics utilities
fingerprint	Fingerprint application dependencies
htcondor	HTCondor high-throughput computing workload management system
ganglia	Cluster monitoring system
hpc	Tools for running parallel applications
kvm	Support for building Kernel-Based Virtual Machine (KVM) virtual machines on cluster nodes
perl	Perl RPM, Comprehensive Perl Archive Network (CPAN) support utilities, and various CPAN modules
python	Python 2.7 and Python 3.x
Web-server	Rocks web server roll
Zfs-linux	Zetabyte File System (ZFS) drivers for Linux

Table 2. Components of current XCBC build Part 2 – Components specific to XSEDE cluster "run-alike" compatibility

Category	Specific packages
Compilers, libraries, and programming	Charm, compat-gcc-34-g77. gcc, gcc-gfortran, fftw2, fftw, gmp, hdf5, java-1.7.0-openjdk, libRmath, libRmath-devel, mpfr, mpi4py-common, mpi4py-
tools	mpich2, mpi4py-openmpi, mpich2, openmpi, PSM API, numactl, librdmacm, libibverbs, papi, python, tcl, R, R-core, R-core-devel, R-devel, R-java, R-java-devel
Scientific Applications	BEDTools, GotoBLAS2, PLAPACK, PnetCDF, SHRiMP, Abyss, arpack, atlas, autodocksuite, boost, bowtie, bwa, darshan-runtime-mpich, darshan-runtime-openmpi, darshan-util, libgfortran, libgomp, elemental, espresso-ab, gatk, glpk, gnuplot, libXpm,

	gd, gnuplot-common, gromacs, gromacs-common, gromacs-libs, hmmer, lammps, lammps-common, libgtextutils, lua, meep, mpiblast, mrbayes, ncbi-blast, ncl, ncl-common, nco, netcdf, numpy, octave, petsc, picard-tools, plplot, libtool-ltdl, saga, libmspack, wxBase3, wxGTK3, Samtools, scalapack-common, shrimp, slepc, sparsehash-devel, sprng, sratoolkit, sundials, trinity, valgrind
Miscellaneous Tools	ant, scone, giflib, libesmtp, libicu, pulseaudio-libs, libasyncns, libsndfile, libvorbis, flac, libogg, libXtst, rhino, jpackage-utils, jline, tzdata-java, wxBase, wxGTK, wxGTK-devel, xorg-x11-fonts-Type1, xorg-x11-fonts-utils,
Scheduler and Resource Manager	maui, torque
XSEDE Tools	Globus Connect Server, Genesis II, GFFS

Software available as part of the XCBC build includes a number of scientific packages and supporting software added since earlier reports about XCBC [5]. There have been two major XSEDE Rocks Rolls released since the 2014 report. Version 0.0.8 saw a major OS release update from Centos 6.3 to 6.5 and 27 scientific and supporting packages have been added, including GenomeAnalysisTK, gromacs, mpiblast, and others [15]. The 0.0.9 release from November 2014 saw 41 additions, including TrinityRNASeq, R, significant Java updates, and other scientific and supporting packages [16]. Software included in XNIT, but not part of the basic XCBC build, continues to evolve in response to community requests.

3. BUILDING XCBC FROM SCRATCH OR ADDING PARTICULAR COMPONENTS TO AN EXISTING CLUSTER VIA XNIT

The Rocks team has worked for a number of years to help enable the creation of easily deployed and managed clusters [17]. Using CentOS as their base operating system, they have created a system for managing computational nodes from a central (frontend) node [8]. This creates a fairly simple way to deploy a basic cluster. Using an internal database, Rocks can manage many compute nodes. This allows an administrator to easily add, remove, and upgrade software across nodes and to maintain a uniform environment.

Using the XSEDE roll during the Rocks cluster install will add the packages necessary for an XSEDE-compatible basic cluster. Once up and running, to maintain the package levels, you can enable the XSEDE Yum repository, then follow the Rocks instructions or use the preferred method and create an update roll to add to your distribution [18]. The negative side of the Rocks upgrade options is that neither method will seem easy to a novice administrator. So while clusters are relatively easy to bring online and expand, upgrading and other more in-depth maintenance may be daunting to less experienced users, which may mean clusters aren't maintained, kept secure, or 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 XCBC.

Using XNIT to create an XSEDE-compatible cluster is a fairly easy task. An administrator would need to initially set up the repository configuration. There are two ways to do this. The first method is to download and install the XSEDE repo RPM from the XSEDE Yum repository [19]. The second is to install the yum-

plugin-priorities package, then create the file /etc/yum.repos.d/xsede.repo with the lines specified in the XSEDE Yum repository README file [13].

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. Then it will provide the administrator with a full list of packages to be updated. Yum still requires an administrator to periodically run update checks. Tools are available (or admins can write their own scripts and cron jobs) to either automate Yum updates or notify administrators of package updates.

Updating packages automatically may cause unexpected behavior in a production environment, especially for less-experienced system administrators. 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 that do this such as Yum updates developed by Duke and available from CentOS and other distribution packagers.

4. IMPLEMENTATIONS OF XCBC AND XNIT FOR RESEARCH TO DATE

There are now a number of clusters in operation that use XCBC or XNIT as the primary or supplemental source of cluster management and application software. These include clusters at Howard University, Michigan State University, Marshall University, Montana State University, and the University of Hawaii. The first three clusters are built from the ground up with the XCBC Rocks installation media, while those at Montana State University and the University of Hawaii use the package repository.

Two clusters had been in operation and were torn down and rebuilt from scratch with XCBC. The Marshall University cluster, consisting of 264 cores in 22 nodes (2.8TF theoretical), including 8 GPU nodes with 3584 CUDA cores, leveraged the XCBC to replace a prior cluster management system. The cluster at Howard University, which is operated by a professor of chemistry, had also been in service under another management system and was taken down and rebuilt from scratch with XCBC, to the significant satisfaction of the professor responsible for it (Dr. Marcus Alfred). All of these implementations have been done with support from the XSEDE Campus Bridging team. In the case of Marshall University XSEDE campus bridging staff spent a week on site working with the Marshall University IT staff.

XCBC adopters have performed a critical function in hardening the installation and implementation of XCBC, and have provided guidance on which packages should be included in the system. Administrators of the Montana State installation, in particular, have been instrumental in investigating how to implement software from XCBC in environment modules, and integrate it with existing cluster management systems. Colleagues at the University of Hawaii have also been extremely helpful in helping us learn how to successfully integrate particular components of XCBC to supplement an existing commercial cluster management system.

Table 3. Deployed XCBC Clusters that had XSEDE Campus Bridging team involvement.

Site	Nodes	Cores	Rpeak (TFlops)	Other Info
University of Kansas	220	1760	26.0	Will be in production in

				summer 2015
Montana State University	36	576	11.98	300 TB of Luster storage [20]
Marshall University	22	264	6.0	8 GPU Nodes, 3584 CUDA Cores [21]
Pacific Basin Agricultural Research Center (Univ. of Hawaii – Hilo)	16	80	4.3	40TB storage, 60TB scratch
Indiana University	6	12	.54	LittleFe Teaching Cluster
Indiana University	4	16	.79	Limulus HPC 200 Cluster
Total	304	2708	49.61	

Clusters making use of XCBC or XNIT total almost 50 TFLOPS of processing capability. By the end of 2020, nearing the end of the second XSEDE funding, our goal is to have the aggregate processing capacity of the clusters making use of XCBC and XNIT exceed half a PetaFLOPS of processing capacity.

5. XCBC ON LITTLEFE AND XNIT ON LIMULUS HPC200 LUGGABLE CLUSTERS

The need for cluster administrators exceeds supply for a variety of reasons. There is never enough budget, and cluster administration often falls through the cracks in education and training efforts. A minority of computer science departments teaches classes in cluster administration. Also, cluster administration is not generally included in training efforts outside the credit-bearing curriculum. For example, the highly effective Software Carpentry project does not include cluster administration [22].

The LittleFe project began in 2005 under the leadership of Paul Gray (University of Northern Iowa), Dave Joiner (Kean University), Tom Murphy (Contra Costa College), and Charlie Peck (Earlham College). The goal of this project was to engineer a low-cost, easy-to-assemble cluster for the purpose of cluster administration and computer science education. LittleFe is a complete 6-node, Beowulf-style cluster that weighs less than 50 pounds and can be built from easily available components for less than $4,000. It is straightforwardly luggable, if not quite easily portable. The LittleFe project expanded in 2010 with a grant award from Intel to build 25 LittleFe devices and deploy them across the country for computational science education [23].

Another luggable option for teaching cluster administration and computational science is the Basement Supercomputing Limulus series Personal Cluster Workstation. The Limulus HPC200

Personal Cluster Workstation encases one headnode and three compute nodes in a single case [24] weighing 50 pounds. It is built using Scientific Linux, an RPM-based Red Hat Linux variant [25]. The Limulus system is more polished and self-contained with some integration work that gives adequate power and resource management for $5,500 to $8,000.

5.1 A modification to standard LittleFe design enabling use of XCBC and improving numerical performance of LittleFe

We have developed a modification of the standard LittleFe design to enable use of XCBC with LittleFe and improve numerical performance of the LittleFe design. The LittleFe design is similar to a blade chassis. Instead of using boards designed for a specific footprint, backplane, and task, it uses off-the-shelf parts to accomplish the same idea of a (relatively) large number of cores in a small, economical footprint. The goal is to provide an environment suited to teaching computational science with the most computing power in a portable package. The LittleFe design uses atom-based processors that provide x86 instruction set compatibility in a low-power configuration on mini-ITX, small-footprint boards. These system-on-a-board configurations allow for a very small overall design with modest power needs.

As is the nature of the computing world, the choice of components for building a LittleFe have evolved since the last instruction set (LittleFe v4 [26]). This is relevant to assembling a cluster for training purposes and provides the option of creating a LittleFe cluster for research purposes. One could change power supply options and use mini-ITX-based, Haswell-based Celeron CPUs for a more modern, yet cost-effective CPU. This increases power requirements somewhat for significant gains in single-core performance [27]. (These CPU choices also eliminate the option of using hyperthreading, which may be an issue depending on training goals.) Also, because the XCBC-from-scratch installation is based on Rocks, and because Rocks does not support diskless installation, the standard LittleFe components must be expanded to include a hard drive to be used as a training tool specifically with XCBC. This means adding some sort of hard disk drive (HDD) or solid-state drive (SSD) for each node. One could physically mount a 2.5-inch laptop-type drive for each node. An alternative would be to use an internal mini Serial-ATA (mSATA) drive that directly mounts to a compatible motherboard. The advantage is minimizing space in the LittleFe rack while minimizing components that need to be isolated electronically as well as physically secured. The disadvantage is that each added component increases the power needs.

We built an exemplar of a modified LittleFe using Haswell-compatible Gigabyte mini-ITX boards utilizing the LGA-1150 socket [28]. In addition, we added Crucial 128gb internal mSATA drives [29] on each node. We used a hard-wired connection using a dual-homed headnode. All nodes utilize the same motherboard, but only one of the two network interfaces will be used on compute nodes. The differences in power needs for the CPU and disk on each node meant that we had to diverge from the single power supply LittleFe calls for. Instead, we added an individual power supply for each node. This adds complexity to the assembly process but enables more flexibility for future upgrades.

Figure 1. A LittleFe V4 frame showing six nodes exposed in a single, portable chassis, rear view.

The original LittleFe used a heat sink on the CPU and a small add-on fan to blow air over the heat sink fins. Since the power needs of the Haswell CPUs are higher, we had to add a CPU fan for cooling. The Atom (D510) used historically in the LittleFe build uses 10.56 watts versus 43.06 watts for the Celeron G1840 [27]. The fan that comes packaged with the Celeron G1840 processor we used is too large to fit in the space allocated per LittleFe node. You need to use a lower-profile fan assembly. We chose the Rosewill RCX-Z775-LP 80mm Sleeve Low Profile CPU Cooler as it fits well in the allotted space.

Figure 2. A LittleFe V4 frame showing six nodes exposed in single, portable chassis, front view

Figure 1 and Figure 2 show the LittleFe frame with the slightly modified design. The boards are still mini-ITX form factor, but using Gigabyte GA-Q87TN motherboards that use the LGA-1150 socket for more modern processors [28]. The fan housing is visible in these pictures.

As of the publication of this paper, XCBC will be included among the standard supported options for operating environments within the LittleFe project. Instructions for XCBC on LittleFe clusters and the parts list and building instructions are included in the LittleFe web site and class materials [30].

5.2 LIMULUS HPC200 AND XNIT

The Limulus HPC200 from Basement Supercomputing is a commercial product that puts a cluster in a deskside computer enclosure. It includes fewer compute nodes than the Rocks-based LittleFe but they are diskless in design, so a little less complex. The HPC200 has an 850W power supply, allowing for more powerful CPUs, consistent with its main purpose as a personal cluster The current build uses i7-4770S CPU Haswell (3.10GHz, 8MB cache, 65 watts). Since there are fewer nodes, maximizing the CPU power available for this power footprint is key. The HPC200 currently provides 16 cores of Haswell-generation die CPUs versus the 12 cores in the IU-built LittleFe. [1]. Further, there is power management that turns nodes on and off as needed for maximum power efficiency. This can also be scheduled [31]. The Limulus HPC200 with cover removed is shown in Figure 3.
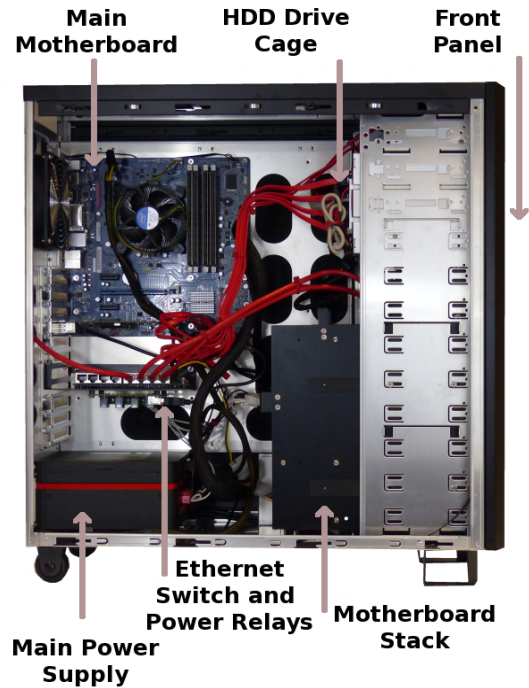

Figure 3. The internals of the Limulus HPC200 Deskside Cluster architecture.

The HPC200 is delivered with software cluster management utilities off the shelf, so one has only to add RPMs from the XSEDE Yum repository to get the desired XCBC capabilities.

6. XCBC, LITTLEFE, XNIT, and LIMULUS HPC200 FOR TRAINING AND EDUCATION

LittleFe was developed to support education in high performance and parallel computing. Since 2005 LittleFe has been featured in workshops and conferences, including Supercomputing (SCxx) and TeraGrid/XSEDE, and at a number of smaller conferences and internship programs [32]. In 10 years of instruction, LittleFe has shown that it meets the needs as an inexpensive and portable means for teaching HPC principles.

We are expanding this rich history by adapting XCBC to the LittleFe model. A curriculum module entitled "Building and administering a Beowulf-style cluster with Little Fe and the XSEDE-compatible Basic Cluster build" is available from the LittleFe web site [30]. Similarly, XSEDE and HPC University offer a variety of online training modules on parallel computing, scientific computing, and available software [33, 34]. The XSEDE campus bridging team will maintain a web page on the XSEDE site [4] that links to training modules that may be used as is with an XCBC build operating on a LittleFe or XNIT and Limulus HPC200 cluster. Together, these resources will enable educators to teach cluster administration and a good variety of important topics in parallel computing.

7. XCBC, LITTLEFE, and LIMULUS HPC200 FOR RESEARCH

The Limulus HPC200 cluster is billed as a turnkey solution for personal scientific computing, targeting workloads requiring fewer than 16 cores [31]. LittleFe was developed as a teaching tool. However, given the CPU modifications of LittleFe presented in this paper, it's worth considering either system as a potential research computing resource for an individual researcher.

Table 4 shows the basic processing components of the Limulus HPC200 and LittleFe cluster as constructed with the components described here. Table 5 shows the peak theoretical processing capability (Rpeak) and maximum achieved processing capability (Rmax) in GFLOPS of the LittleFe described in this paper and the Limulus HPC200. The Rmax shown for Limulus HPC200 is based on actual results of tests conducted by Basement Supercomputing [35], based on the HP Linpack benchmark [36].

Table 4. Basic characteristics of a Limulus HPC200 cluster and a LittleFe cluster

Cluster	Nodes	CPU clock rate	CPUs	Cores
LittleFe	6	2.8 GHz	6	12
Limulus HPC200	4	3.1 GHz	4	16

Table 5. Performance and price/performance for LittleFe and Limulus HPC200.

System	Rpeak	Rmax	Cost	Rpeak $/GFLOPS	Rmax $/GFLOPS
LittleFe	537.6	403.2*	$3600	$7/GFLOP	$9/GFLOPS
Limulus HPC200	793.6	498.3	$5995	$8/GFLOP	$12/GFLOPS

* Rmax for LittleFe is estimated due to a hardware failure prior to Linpack. Estimated at 75% of Rpeak. Testing will be complete prior to conference.

While LittleFe was originally conceived as an educational tool, the components used in the LittleFe system constructed here provide a very reasonable solution for a deskside cluster. A half-TeraFLOPS deskside cluster for under $4,000 could be attractive to a number of researchers as could a roughly $6,000, three-quarter-TeraFLOPS deskside system with considerable local storage capabilities as a commercial product. Additionally, these prices are an order of magnitude lower than similarly powered systems in a typical server configuration [36] [37]. Use of the XNIT Yum repo helps provide a straightforward way to keep such

a cluster updated, which is certain to be a concern for scientists using deskside systems for parallel computing.

8. CONCLUSION

The Rocks project, XCBC Rocks Roll and the XNIT Yum Repository are important tools that enable automating many cluster administration tasks. With the XCBC build and the associated implementations for installation from scratch via Rocks, or addition of specific sets of XCBC components, we have made it possible for cluster administrators to manage a cluster that keeps the software consistent with the current open-source software available via XSEDE-supported clusters. While there is no one gold standard for how best to set up a cluster, consistency with XSEDE offers many advantages to cluster users and administrators.

These advantages include using a tried-and-true cluster management system such as Rocks. Rocks works well for the experienced, novice, or intermediate administrator. Rocks is proven on systems from test clusters to intermediate clusters like Marshall University's, all the way up to national resources such as the San Diego Supercomputing Center Gordon supercomputer [38]. In addition, the common software packages and configurations on XSEDE resources packaged for local clusters should help reduce use barriers for researchers. A common repository for maintaining and upgrading these scientific software packages simplifies keeping the system up to date. Finally, the XSEDE Campus Bridging team is very active and strives to be responsive to XCBC administrators' and users' needs and requests.

Our work with XSEDE Campus Champions, ACI-REF, and other information technology professionals at small colleges and universities confirms that there are inadequate staff resources to administer and support local cyberinfrastructure resources. Our interactions with faculty at such schools confirm that for faculty, one of the most difficult types of time to get is time for curriculum development. Of the existing XCBC and XNIT installations, all but one are at universities that are either Minority Serving Institutions or Institutions in an EPSCoR state. Our initial beliefs about the utility of XCBC and XNIT to researchers and IT professionals at such institutions are borne out by the current pattern of adoption.

Relatively low-cost solutions can be utilized in secondary-education STEM classes as well as post-secondary institutions. While learning the basic principles of computer science may be the key focus in secondary education, preparing students to use HPC resources and understand the fundamentals can only help to create a solid foundation for future education and research. Using free software such as XCBC and XNIT can serve to train secondary students in programming and research principles and possibly in system administration principles as well.

A result of previous work is that faculty at small colleges with limited time for curriculum development can now create and administer a cluster using XCBC or XNIT, and use curriculum tutorials on XSEDE resources. They do not need to re-create new materials specialized for local resources. And for faculty who are their own cluster administrators, that task is easier and less time consuming. With tools such as LittleFe or a Limulus HPC200 cluster, classes can use these clusters without impacting research going on with departmental or university resources. Also, bare-metal installations can be done as part of the curriculum, meaning students experience installing clusters and software and

monitoring. This helps educate the next generation of server administrators and admin-savvy researchers.

While there are many ways to achieve the goals of a teaching HPC system and a practical, deskside HPC system, these two methods seem to fit the criteria. Both systems are portable, with the LittleFe weighing under 50 pounds and the Limulus HPC200 weighing in at 50 pounds. They are similar in size, reasonably portable, and have enough power to demonstrate HPC capabilities. Both systems offer advantages. LittleFe is built from off-the-shelf components for a modest budget ($3,000 to $4,000) and can be easily upgraded.

The same result could be accomplished with "scrap" or recycled hardware, but this has several downsides. Generally you'd be using hardware a number of generations old, and often in the academic world, at the end of its lifecycle. Old workstations will have a considerably larger footprint, and will be noisier and often ill-configured to become a modest HPC cluster. Old servers, by contrast, may be better suited for the task. However, they are often very large form factor (multi "u" 19-inch rackmount or comparable), have air handling that may be loud and unsuited to an office or educational environment, and are definitely not portable or practical as deskside or teaching applications.

While options such as the Raspberry Pi are often used for teaching computer science principles, such solutions aren't as practical for teaching real-world parallel languages or HPC applications, or for small problem-solving or experimental clusters because they are not based on the x86 instruction set. LittleFe and the Limulus HPC200 cluster are. Combined with the XCBC and XNIT tools software, they may be used as instructional clusters using the same software set found on a typical XSEDE or campus cluster.

The concept of creating a small cluster from inexpensive hardware is hardly a new one. The Beowulf concept is more than 20 years old. The idea of creating a small-footprint cluster for teaching, outreach, and even prototyping projects, however, is somewhat new. LittleFe and the Limulus HPC200 cluster models embrace this compact design and show that there are many excellent uses and possibilities with small, portable clusters.

Commercial cloud services are often touted as a resource for HPC education, and in certain contexts may be an excellent choice. However, there is a fundamental difference in the cost and payment models of a small cluster vs. use of AWS or other commercial cloud providers. With a small cluster, one-time monies can be pooled to purchase a hardware resource that is as large as appropriate / possible, and maintained over time with very little cost. Cost is fixed at purchase time, which can be very practical for any group operating on a limited budget. The usage of such a cluster is capped by the capabilities of the system purchased. Use of commercial cloud is typically an ongoing service expense rather than a one-time capital expense. It can be surprisingly straightforward for an enterprising student to use more resources (and commit more university funds) than intended, since not all commercial services support proactive capping of usage. These are some of the practical reasons for choosing the approach of a small cluster such as LittleFe or Limulus HPC200 for institutions with limited budgets.

The Limulus HPC200 also gives a different perspective on teaching the mechanics of creating an XCBC. The XSEDE Yum repository allows a user to take an existing cluster and install software that makes it the equivalent of an XSEDE cluster [11]. Most prior discussions of creating an XCBC have focused on using the Rocks installation to start from bare metal. Using the Limulus HPC200, one can take the running cluster, and with

XNIT add software, change the schedulers, and easily document the approach to make it reproducible.

In summary, the XCBC build is a useful addition to existing tools for cluster management and cluster systems software. XCBC has already proven useful in aiding research at a small set of institutions with high-quality researchers and overworked IT staff. The integration of XCBC with the LittleFe project will expand the utility of XCBC in training and educating sysadmins. Further adoption of XCBC will facilitate training and education of parallel computing users generally. Finally, we have demonstrated here that the LittleFe modified design we present offers performance comparable to the Limulus HPC200 at a lower price point. Both offer an option for a deskside-computing environment that may be easily maintained by the practicing scientist using XCBC and the XSEDE Yum repository for cluster management and maintenance. We expect wider adoption of XCBC in the future and believe that such adoption will help successfully address some of the challenges in US cyberinfrastructure identified by National Science Foundation task forces. Our focus on XCBC implementation and support has been the US, because this effort is funded to address specific challenges within the US research community. However, this project is entirely open source and the software, repositories, and training materials are readily accessible to anyone to download and use throughout the world.

9. ACKNOWLEDGMENTS

10. PHYSICAL ADDRESSES

IU Pervasive Technology Institute: Indiana University, 2709 E. Tenth Street, Bloomington, IN 47408-2671

Earlham College: 801 National Road West, Richmond, IN 47374

11. REFERENCES

1. Basement Supercomputing. *Limulus HPC 200 Personal Cluster Workstation*. [cited 2015 February 9]; Available from: http://basement-supercomputing.com/index.php/products/hikashop-menu-for-products-listing/product/2-limulus-hpc-200-personal-cluster-workstation.
2. NSF Advisory Committee for Cyberinfrastructure Task Force on Campus Bridging. *Final Report*. [PDF] March 2011; Available from: http://www.nsf.gov/cise/aci/taskforces/TaskForceReport_CampusBridging.pdf.
3. Hackworth, K., *Finding data (Personal communiction + Spreadsheet)*, J.L. Fischer, Editor. 2014.
4. XSEDE. *Campus Bridging Home Page*. March 2015]; Available from: https://www.xsede.org/campus-bridging.
5. Fischer, J., R. Knepper, M. Standish, C. Stewart, R. Alvord, D. Lifka, et al., *Methods For Creating XSEDE Compatible Clusters*, in *XSEDE '14*. 2014, xsede.org: Atlanta, GA.

6. XSEDE. *Resources Overview*. March 2015]; Available from: https://www.xsede.org/resources/overview.
7. XSEDE, *2013 XSEDE Staff Climate Study Report*, Lizanne DeStefano, L. Rivera., and M.J. Culbertson, Editors. 2013.
8. Rocks Clusters. *Home page*. 15 Jan 2015]; Available from: http://www.rocksclusters.org/.
9. XSEDE. *XSEDE Campus Champions*. 20 Feb 2012]; Available from: https://www.xsede.org/campus-champions.
10. ACI-REF. *Advanced CyberInfrastructure - Research and Education Facilitators - Home Page*. Available from: http://www.aciref.org/.
11. Fischer, J.L., Brown, Paul. *What is the XSEDE Yum Repository, and how do I use it?* 2013 [cited 2015 February 9]; Available from: https://kb.iu.edu/d/bdwx.
12. Fischer, J.L.a.B., Paul. *What software is installed on a "bare-bones" XSEDE-compatible Rocks cluster?* 2013 [cited 2015 January 14]; Available from: https://kb.iu.edu/d/bdww.
13. XSEDE. *readme.xsederepo*. March 2015]; Available from: http://cb-repo.iu.xsede.org/xsederepo/readme.xsederepo.
14. XSEDE. *User Portal - Software Listing*. March 2015]; Available from: https://portal.xsede.org/software/.
15. XSEDE. *README.0.0.8*. March 2015]; Available from: http://software.xsede.org/cb/centos6/x86_64/README.0.0.8.
16. XSEDE. *README.0.0.9*. March 2015]; Available from: http://software.xsede.org/cb/centos6/x86_64/README.0.0.9.
17. Rocks Clusters. *About*. March 2015]; Available from: http://www.rocksclusters.org/wordpress/?page_id=57.
18. Rocks Clusters. *The Rocks Group. Base User Guide Rocks 6.1. Chapter 8: Advanced Tasks – 8.10 System Updates*. March 2015]; Available from: http://central6.rocksclusters.org/roll-documentation/base/6.1/update.html.
19. XSEDE Campus Bridging. *XSEDE Yum Repository*. Available from: http://cb-repo.iu.xsede.org/xsederepo/.
20. Montana State University. *Hyalite Research Cluster*. Available from: http://www.montana.edu/researchcomputing/HyaliteCluster.html.
21. Marshall University. *New high-performance computing cluster gives Marshall unprecedented research, teaching and learning tools*. 2011; Available from: http://www.marshall.edu/murc/computing-cluster/.
22. Software Carpentry. *Home page*. Available from: http://software-carpentry.org/.
23. LittleFe. *History*. [cited 2015 February 9]; Available from: https://littlefe.net/history.
24. Basement Supercomputing. *Limulus HPC 100 Personal Cluster Workstation*. Available from: http://basement-supercomputing.com/index.php/products/hikashop-menu-for-products-listing/product/1-limulus-hpc-100-personal-cluster-workstation.
25. Scientific Linux. *Why make Scientific Linux?* [cited 2015 February 3]; Available from: https://www.scientificlinux.org/about/why-make-scientific-linux/.
26. LittleFe. *LittleFe 4d Parts List*. March 2015]; Available from: https://littlefe.net/parts-v4d.
27. CPU Boss. *Intel Celeron G1840 vs Atom D510* [cited 2015 February 9]; Available from: http://cpuboss.com/cpus/Intel-Celeron-G1840-vs-Intel-Atom-D510.
28. Gigabyte. *GA-Q87TN*. Available from: http://www.gigabyte.com/products/product-page.aspx?pid=4755 - ov.
29. Crucial. *Crucial M550 128GB mSATA Internal SSD*. March 2015]; Available from: http://www.crucial.com/usa/en/ct128m550ssd3.
30. LittleFe. *Curriculum*. Available from: https://littlefe.net/curriculum.
31. Basement Supercomputing. *Limulus Personal Cluster Questions*. Available from: http://basement-supercomputing.com/index.php/faq/personal-cluster-questions.
32. LittleFe, *On the Road*. https://littlefe.net/on-the-road
33. HPC University. *Home Page*. March 2015]; Available from: http://www.hpcuniversity.org/.
34. XSEDE. *XSEDE User Portal Online Training*. March 2015]; Available from: https://portal.xsede.org/online-training.
35. Basement Supercomputing. *i7-4570S-Raw-64G-HPL*. March 2015]; Available from: http://limulus.basement-supercomputing.com/wiki/i7-4570S-Raw-64G-HPL.
36. TOP500.Org. *TOP500 Description*. March 2015]; Available from: http://www.top500.org/project/top500_description/.
37. Dell Computers. *Poweredge VRTX*. 2015; Available from: http://www.dell.com/us/business/p/poweredge-vrtx/fs.
38. San Diego Supercomputer Center. *SDSC User Support: Resources: Gordon*. March 2015]; Available from: http://www.sdsc.edu/us/resources/gordon/.