

Annual Report for Period:01/2007 - 12/2007**Submitted on:** 09/09/2007**Principal Investigator:** Hayden, Linda B.**Award ID:** 0636361**Organization:** Elizabeth City State Univ**Title:**

CI-TEAM Implementation Project: Cyberinfrastructure for Remote Sensing of Ice Sheets

Project Participants**Senior Personnel****Name:** Hayden, Linda**Worked for more than 160 Hours:** Yes**Contribution to Project:****Name:** Gogineni, S. Prasad**Worked for more than 160 Hours:** Yes**Contribution to Project:****Name:** Fox, Geoffrey**Worked for more than 160 Hours:** Yes**Contribution to Project:****Post-doc****Graduate Student****Undergraduate Student****Technician, Programmer****Other Participant****Research Experience for Undergraduates****Organizational Partners****University of Oklahoma ci-team**

'CI-TEAM Demonstration: Cyberinfrastructure Education for Bioinformatics and Beyond'

\$249,974

12/01/06 - 11/30/08

OCI-0636427

A High Performance Computing Workshop Series entitled 'Supercomputing in Plain English' was provided by the University of Oklahoma ci-team project as content for the virtual classroom.

Other Collaborators or Contacts

Activities and Findings

Research and Education Activities:

The vision for the NSF CI-TEAM Implementation Project entitled "CyberInfrastructure for Remote Sensing of Ice Sheets" is to equip the current and next generation of traditionally, underrepresented minority scientists, engineers and educators with the knowledge and skills necessary to conduct interdisciplinary research in areas including cyberinfrastructure (CI), remote sensing, engineering and modeling related to glaciers and ice sheets. This project is a significant step forward in providing the necessary virtual training environment and grid computing power that the students and scientist associated with the NSF S&T Center for Remote Sensing of Ice Sheets (CReSIS) require.

Primary program accomplishments for year one include initiating the virtual environment training activities for faculty and students related to CI and CReSIS science. There has been significant involvement of undergraduates, graduate students, K-12 and faculty in the CReSIS and CI seminars and workshops via the virtual classroom. Also important was initial configuration of the ECSU GRID for education and training.

The first cohort of student were identified from the Association of Computer and Information Science/Engineering Departments at Minority Institutions (ADMI). These students participated in the 2007 ADMI conference workshops held in Atlanta Georgia and also participated in an 8 week workshop held on the campus of ECSU during summer 2007. The students in this cohort a) learned the fundamentals of GRID computing, documented and created a four-node GRID; b) participated in seminars focused on CReSIS science including GIS and Its Applications, Radar: SAR and INSAR, UAV Design I, Introduction to Ice Sheets and Glaciers, and MATLAB training.

Cohort 1 worked with the GRID manager during summer 2007 to setup a documentation platform for a Condor-based GRID to be established at Elizabeth City State University. The task assigned to the ECSU Teragrid team for the Summer of 2007 was to setup a documentation platform for a Condor-based grid. This documentation platform was a Linux based web sever that utilized Web 2.0 standards to create a virtual documentation web portal. Documentation is critical to communicate with the users, and with those who maintain of the systems.

The web server created utilized an Ubuntu Linux kernel with an Apache web server, a MySQL Database, a PHP scripting package, and a Media Wiki web interface. This particular setup is called a LAMP server. LAMP is the acronym for Linux, Apache, MySQL, and PHP which are all open source applications. The combination of these LAMP applications allows for Media Wiki to function. The documentation server is now being used to record how to setup a Condor-based GRID system.

With regards to the Grid, access is being implemented through web and research portals for use of computational and data resources. The portals will be customized with educational and science inter-faces while still allowing access to large amounts of data. The virtual classroom configuration consist of a Polycom 8000S system with 14-kHz wideband audio, 2 channels of 14-kHz audio, standalone audio conferencing phone, IP (H. 232 and SIP) interfaces and Global Management System collaborative communications. For receiving at the partner institutions, VIA Video for desktop PCs have been provided to ADMI institutions to allow point-to-point connectivity. ConferenceXP by Microsoft is also offered as an option for ADMI institutions.

Findings: (See PDF version submitted by PI at the end of the report)

Nothing Significant to report at this time however, a copy of the summer 2007 student

research paper entitled 'Installation and Implementation of a LAMP Documentation Server' is attached.

Training and Development:

1. Faculty development activities have included the following:

a) STELLA« is a software package for building and simulating models of dynamic systems and processes. Using a set of building block icons, the developer can construct a map of a processes. This allows the user a practical way to dynamically visualize and communicate how complex systems and ideas really work. Faculty from the following institutions participated: Delaware State University, Virginia State University, Elizabeth City State University, Spelman College, and Norfolk State University. <http://cerser.ecsu.edu/07events/070320dls/stella.html>

b) Fall 2007 virtual training workshops are schedules as follows in conjunction with the CI-TEAM project at the University of Oklahoma. CReSIS faculty and students are participating in the following workshops:

Wed Aug 29 3:00pm: Overview: What the Heck is Supercomputing?

Wed Sep 5 3:00pm: The Tyranny of the Storage Hierarchy

Wed Sep 12 3:00pm: Instruction Level Parallelism

Wed Sep 19 3:00pm: Stupid Compiler Tricks

Wed Sep 26 3:00pm: Shared Memory Multithreading

Wed Oct 10 3:00pm: Distributed Multiprocessing

Wed Oct 17 3:00pm: Multicore Madness

Wed Oct 24 3:00pm: High Throughput Computing

Wed Oct 31 3:00pm: Grab Bag: Scientific Libraries, I/O, Visualization

c) Support was provided for two staff to attend the 2007 Midwest Grid Workshop at Northwestern University. This was a two-day weekend workshop program in grid computing and its application to scientific data analysis. The aim of the workshop was to provide a basic foundation in distributed computing and valuable hands-on training in computing techniques. The workshop introduced essential skills that are needed in the natural and applied sciences, engineering, and computer science to conduct and support scientific analysis in the emerging grid computing environment.

2. Graduate, Undergraduate and K-12 training

The project supported participation of graduate students, undergraduate students and middle school students both Cyberinfrastructure and CReSIS related seminars and training events as described below:

a) Open Grid Forum 19 (OGF19) was held in Chapel Hill, North Carolina at The Friday Center on the campus of the University of North Carolina at Chapel Hill. Jeff Wood, graduate student at Elizabeth City State University (ECSU) attended OGF19 as part of a minority serving institutions grant to OGF. Various meetings were held on topics from Grid software to standardization and working groups. Software such as Condor, Semantics, Globus and others were demonstrated by the groups developing the software. A complete listing of the workshops along with documentation can be found at <http://www.ogf.org/OGF19/schedule/>. Highlights are available at <http://nia.ecsu.edu/ur/0607/070129grid/ogf19.htm>

b) On March 21, 2007 students took part in the Spring Seminar Series. Students took part in an interactive seminar with Dr. Seppo A. Korpela of The Ohio State University who's seminar was entitled 'Peak Oil.' Dr. Korpela is a professor of mechanical engineering whose interests focus on worldwide oil depletion and natural gas depletion in North America. He serves on the editorial boards of two journals and on the advisory board of

the Association for the Study of Peak Oil, USA.

c) During the April 11, 2007 CReSIS Spring Seminar, students took part in an interactive seminar with Dr. Robert Bindshadler, of the NASA Goddard Space Flight Center, who presented 'Ice Sheets on the Edge: A Golden Age for Glaciology.' Dr. Robert Bindshadler is the Chief Scientist of the NASA's Hydrospheric and Biospheric Sciences Laboratory. He is a Senior Fellow of the Goddard Space Flight Center, a Fellow of the American Geophysical Union and a past President of the International Glaciological Society. <http://censer.ecsu.edu/cresis/events/2007ybbs/april11.html>

d) On May 9, 2007, Dr. Don Worster of The University of Kansas rounded out the CReSIS Spring Seminar Series. ECSU undergraduates, faculty, and local middle/high school students attended Dr. Worster's seminar entitled 'Feeling the Heat.' Dr. Worster discussed the significance of global climate change for the southern Great Plains. The available projections suggest a hotter, drier future for that region, bringing back conditions that gave us the tragic Dust Bowl of the 1930s. Dr. Worster currently holds the Hall Distinguished Professorship Chair in American History at the University of Kansas. Dr. Worster graduated from KU in 1963 (B.A.) and 1964 (M. A.). Subsequently, he did graduate work in American history and literature at Yale University and earned his Ph.D. there in 1971. <http://censer.ecsu.edu/cresis/events/2007ybbs/may9.html>

e) An eight week summer program was conducted for undergraduate students from ECSU and ADMI institutions. Leveraging funds from the Office of Naval Research, twenty student participated in the summer program. Students from Mississippi Valley State University, Norfolk State University, Winston Salem State University, Elizabeth City State University, Fayetteville State University, Jarvis Christian College, and St. Augustine's College participated. Each student was assigned to a specific research team, where he/she worked closely with the faculty. In addition, seminars, lunch meetings, and social functions were organized to facilitate undergraduate interaction. Highlights are available at <http://nia.ecsu.edu/ureoms2007/index.html>. CReSIS and Cyberinfrastructure related research projects conducted during the summer included:

- Creation of a Flexible and Scalable Distributed Computing Infrastructure Using OpenGRID Project Standards.
- Antarctic Firn Annual Emissivity Trends at the Ski-Hi Automatic Weather Station from in-situ and SSM/I Brightness Temperatures.
- Sub-glacial topography and ice discharge of the Greenland Ice Sheet
- Hardware Modeling and Machining for UAV - Wideband Radar

Outreach Activities:

With regards to outreach activities which increase public understanding of, and participation in, CI and CReSIS related events, both a two week summer program for middle school students was conducted and one live distinguished lecture which was open to the public.

The project is committed to recruiting, retaining and educating students within a diverse, multidisciplinary research team that focuses on topics related to global climate change and remote sensing. Equally important is our goal to reach out to K-12 students, their teachers and the general public. We want to give them resources to learn more about CI, the Polar Regions and understand how they are studied, so all members of society can better comprehend why changes in Polar Regions could impact everyone in the world. Highlights are available on <http://censer.ecsu.edu/cresis/cmosp2007/>. Outreach activities

involve participants in both CI and CReSIS related activities. During the two-week summer 2007 program middle school students:

- *Learned about Remote Sensing and Satellite Imagery
- *Took scientific measurements in the fields of atmosphere and hydrology
- *Learned about the Polar Regions
- *Collaborated with scientists and faculty
- *Used HTML and JavaScript to document their experience on the WWW

The Spring 2007 IEEE-GRSS Distinguished Lecture Series was held in the Center of Excellence in Remote Sensing Education and Research (CERSER) on Tuesday March 20, 2007. Dr. Robert Bindshadler, Chief Scientist of the NASA's Hydrospheric and Biospheric Sciences Laboratory at NASA's Goddard Space Flight Center presented 'Understanding the West Antarctic Ice Sheet from Space: Beyond Dogsleds and Frozen Toes.' Over 50 people participated in the distinguished lecture which was open to the public. Highlights are available on <http://cerser.ecsu.edu/07events/070320dls/07dls.html>

Journal Publications

Books or Other One-time Publications

Linda Hayden, Goeffry Fox, Prasad Gogineni, "Cyberinfrastructure for Remote Sensing of Ice Sheets", (2007). , Published
Collection: TERAGRID Conference, Madison, WI, June 4-8, 2007
Bibliography: Proceedings of the 2007 TERAGRID Conference, Madison, WI, June 4-8, 2007

Linda Hayden and David Braaten, "Development of Educational Partnerships Dedicated to Remote Sensing of Ice Sheets Cyberinfrastructure", (2007). conference paper, Published
Collection: IEEE-IGARSS Conference, Barcelona, Spain, July 2007
Bibliography: Proceedings of the IEEE-IGARSS Conference, Barcelona, Spain, July 2007

Web/Internet Site

URL(s):

<http://cerser.ecsu.edu/citeam/>

Description:

The site contains a description of the project, the management team, personnel contact information, partners and links to related sites. Photos and highlights from project events are also available at this site.

Other Specific Products

Contributions

Contributions within Discipline:

Project has created partnerships and training opportunities that extend the capabilities of Polar Scientist within the Center for Remote Sensing of Ice Sheets (CReSIS) to conduct field

operations and to process SAR data. In addition, several accounts have established for CReSIS scientist on the TeraGRID.

Contributions to Other Disciplines:

Contributions to Human Resource Development:

The project staff is actively involved in creating a virtual training environment and educational portals designed specifically for student use. Training events and seminars supported by this project have provided opportunities for underrepresented graduate students, undergraduate students and middle school students to become engaged in both remote sensing of ice sheet research effort and CI.

To date twenty students have participated in the summer 8 week training; fifteen students participated in the ADMI 2007 CI workshops; eight HBCU faculty have participated in visualization training; fifty students and faculty participated in distinguished lectures; and six staff have participated in TeraGrid training events.

Contributions to Resources for Research and Education:

The project has produced a small-node grid for educational purposes. We are currently in the process of creating educational portals designed specifically for student use and a second set of portals designed for use by CReSIS researchers.

Contributions Beyond Science and Engineering:

Special Requirements

Special reporting requirements: None

Change in Objectives or Scope: None

Animal, Human Subjects, Biohazards: None

Categories for which nothing is reported:

Any Journal

Any Product

Contributions: To Any Other Disciplines

Contributions: To Any Beyond Science and Engineering

Cyberinfrastructure for Remote Sensing of Ice Sheets

Dr. Linda Hayden, Dr. Geoffrey Fox, and Dr. Prasad Gogineni

Abstract— Science and engineering research and education are foundational drivers of Cyberinfrastructure. Understanding the relationship between sea level rise and melting ice sheets is the application domain of this project. It is an issue of global importance, especially for the populations living in coastal regions. Scientists are in need of computationally intensive tools and models that will help them measure and predict the response of ice sheets to climate change. To address the Cyberinfrastructure challenges presented immediately by the Center for Remote Sensing of Ice Sheets (CReSIS) and the polar science community in general, the Cyberinfrastructure Center for Polar Science (CICPS), with experts in Polar Science, Remote Sensing and Cyberinfrastructure, has been established. This center includes the University of Kansas (KU), the lead CReSIS institution; Indiana University (IU), which is internationally known for its broad expertise in research and infrastructure for eScience; and Elizabeth City State University (ECSU), a founding member of CReSIS with a center of excellence in remote sensing. CICPS includes CReSIS institutions as collaborators and will drive PolarGrid to meet their goals while using the best-known technologies. Founded with the vision that Cyberinfrastructure will have a profound impact on polar science, CICPS is committed to the effort needed to build the portal, workflow and Grid (Web) services that are required to make PolarGrid real. This paper describes the set of CICPS projects that are being implemented and proposed. The first of these projects is an NSF CI-TEAM project (PI: Hayden, Co-PIs: Fox and Gogineni), "Cyberinfrastructure for Remote Sensing of Ice Sheets," which establishes a virtual classroom environment and a CReSIS Science Gateway for TeraGrid working with IU, Minority-Serving Institution Cyberinfrastructure Empowerment Coalition (MSI-CIEC), and TeraGrid. The second project, called PolarGrid (PI: Fox, Co-PIs: Hayden and Gogineni), as proposed, deploys the Cyberinfrastructure which provides the polar community with a state-of-the-art computing facility to process the large volumes of data to be collected by CReSIS field operations and support large-scale ice-sheet models. PolarGrid will follow modern open data access standards so that raw, processed and simulated data can be archived outside PolarGrid by and for the full science community.

Index Terms— CI-TEAM, Ice Sheets, Polar Science, Cyberinfrastructure

1 INTRODUCTION

Recent polar satellite observations show disintegration of ice shelves in West Antarctica and speed-up of several glaciers in southern Greenland. The great ice sheets in Antarctica and Greenland interact with the global climate in a complex manner, and the impact on global sea level of their retreat would be profound. Most of the existing ice-sheet models, including those used by the Intergovernmental Panel on Climate Change (IPCC), cannot explain the rapid changes being observed. The Center for Remote Sensing of Ice Sheets (CReSIS) is developing new technologies to perform 3-D characterization of ice sheets to understand the physics of rapid changes, and develop models to explain observed changes and predict future behavior. In particular, CReSIS has demonstrated that Synthetic Aperture Radar (SAR) can image the beds of ice sheets. This will enable a new generation of high resolution ice-sheet models with realistic boundary conditions, but it will require distributed Cyberinfrastructure to gather and process data, and assimilate them with large simulations. We propose a sophisticated Cyberinfrastructure instrument that will both enable the crucial ice-sheet science and educate and train a diverse workforce in both Polar Science and Cyberinfrastructure.

We have formed the Cyberinfrastructure Center for Polar Science (CICPS) with experts in Polar Science, Remote Sensing and Cyberinfrastructure. This center includes the University of Kansas (KU),

- Dr. Linda Hayden, *Center of Excellence in Remote Sensing Education and Research, Elizabeth City State University*, haydenl@mindspring.com
- Dr. Geoffrey Fox, *Community Grids Laboratory, Indiana University*, gcf@cs.indiana.edu
- Dr. Prasad Gogineni, *Center for Remote Sensing of Ice Sheets, University of Kansas*, gogineni@crsis.ku.edu

the lead CReSIS institution; Indiana University (IU), which is internationally known for its broad expertise in research and infrastructure for eScience; and Elizabeth City State University (ECSU), a founding member of CReSIS with a center of excellence in remote sensing. CICPS includes CReSIS institutions as collaborators and will drive PolarGrid to meet their goals while using the best-known technologies. Impressive institutional commitments include a new building and faculty lines at ECSU, and resources and system support from IU. We also commit the substantial initial effort needed to build the portal, workflow, and Grid (Web) services that are required to make PolarGrid real.

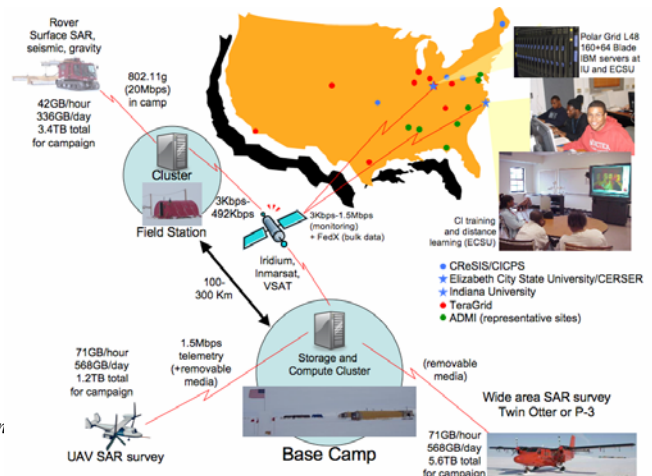


Fig. 1. PolarGrid's major components and concept of operations.

CICPS is founded with the vision that Cyberinfrastructure will have a profound impact on polar science. We will realize this vision through a set of projects that are of the highest quality in all dimensions: Polar Science, Computer Science, Cyberinfrastructure, and EOT (education, outreach and training). The first of these projects is an NSF CI-TEAM project (PI: Hayden, Co-PIs: Fox and Gogineni), "Cyberinfrastructure for Remote Sensing of Ice Sheets," which will establish a CReSIS Science Gateway for TeraGrid working with IU, Minority-Serving Institution Cyberinfrastructure Empowerment Coalition (MSI-CIEC), and TeraGrid.

The second project, as proposed, will deploy the Cyberinfrastructure, called PolarGrid, needed to support polar science and, in particular, the field research program of CReSIS. The major components, including field, base and core systems, are depicted in **Figure 1**.

In past years, most of the data collected in the polar regions were not processed in real time. Instead, data were shipped to computing facilities in the continental US and analyzed well after collection. Real-time processing and data analysis are urgently needed, both in the field and at supporting computing centers, to adjust collection strategies in response to new information contained in the data. The polar community must have access to a state-of-the-art computing facility to process the large volumes of data to be collected by the polar community in 2007-08 as a part of the International Polar Year (IPY) activities, and to support large-scale ice-sheet models.

2 NSF CI-TEAM PROJECT

The vision and goals of the NSF CI-TEAM project, "Cyberinfrastructure for Remote Sensing of Ice Sheets," are based on the fact that "educational settings, audiences, and goals are too important to be adequately addressed as afterthoughts or add-ons to CyberInfrastructure projects and, instead, must be treated as high priorities integrated in a project's overall design" [12]. As such, the NSF CI-TEAM project aggressively engages computer science and engineering students from five minority universities in the Grid, remote sensing, and CReSIS training, seminars, workshops and classes.

ECSU will establish a minimal Grid configuration and a virtual classroom environment with broadcast and receiving capabilities. With regards to the Grid, access will be implemented through web and research portals for use of computational and data resources. The portals will be customized with educational and science interfaces while still allowing access to large amounts of data. The virtual classroom configuration will consist of a Polycom 8000S system with 14-kHz wideband audio, 2 channels of 14-kHz audio, stand-alone audio conferencing phone, IP (H.232 and SIP) interfaces and Global Management System collaborative communications. The system will have standards compliance for traversing NATs and firewalls. Existing plasma display units will be connected to the 8000S system. For receiving at the partner institutions, VIAVideo for desktop PCs will be provided to allow point-to-point connectivity. ConferenceXP by Microsoft will also be offered as an option for partner institutions.

The CReSIS Education Program benefits significantly from the implementation of the NSF CI-TEAM project. The network proposed supports a number of objectives outlined in the CReSIS Strategic and Implementation Plan including: *CReSIS Objective B: To integrate center research to enhance undergraduate curricula and involve undergraduate students in research; and CReSIS Objective G: Assure that graduate students receive the necessary courses and*

training opportunities to succeed in their research endeavors.

The Association of Computer and Information Science/Engineering Departments at Minority Institutions (ADMI) (http://cerser.ecsu.edu/admi2006/purp_gls.html) was founded in August 1989. Represented on the Board of ADMI are Spelman College, Mississippi Valley State University, North Carolina Central University, ECSU, Howard University, Jackson State University, University of the District of Columbia, Hampton University, Fisk University and Florida A&M University. ADMI was established as a national organization dedicated to exploring and providing remedies to the educational issues in computer/information science and computer engineering that confront minority institutions of higher learning. ADMI goals include:

- * Support research initiatives and their continuation at member minority institutions.
- * Serve as an advocate/lobbyist for minority institutions interest at funding agency forums, both public and private.
- * Exchange information and educational training techniques.
- * Identify and seek viable solutions to problems relevant to the disciplines.
- * Exchange information and ideas for improving the quality and effectiveness of computer science and engineering at minority institutions.
- * Work for the continued development of excellent teaching, state-of-the-art curriculum, research, and equipment enhancement.
- * Strive for increased support of the continued, professional advancement of faculty.
- * Solicit and assist in the soliciting of funds for achieving the above stated objectives.

ADMI's role in the proposed ECSU Grid project will be to establish capability within their institutions to allow students and faculty to participate in the virtual workshops, training and courses. In addition, students from the ADMI institutions will participate in an 8-week Grid technology workshop to be conducted during the summers at ECSU.

With particular attention to the current and next generation of traditionally underrepresented minority scientists engineers and educators, the NSF CI-TEAM project prepares students with the knowledge and skills necessary to conduct interdisciplinary research in areas including Cyberinfrastructure, remote sensing, engineering and modeling related to glaciers and ice sheets.

3 POLARGRID

While there is considerable evidence now that fast glacial changes occur and have occurred in the past, there is no consensus yet regarding how this takes place. Better understanding will only come from an ambitious campaign to collect more data, and develop new and better models using such data. The National Science Foundation established CReSIS in 2005 for just this purpose. Major advancement in the quantity and quality of data about polar ice sheets is an important objective of the 2008 and 2009 field data campaigns in which new measurement techniques will multiply the quantities of data available to the polar research community. The field program will consist of coordinated satellite, airborne and in situ measurements. Satellite data acquisitions are being planned as a part of the Global Interagency IPY Polar Snapshot Year (GIIPSY) [7]. The field program includes plans to perform airborne measurements to collect data over an area of 200 km x 200 km with grid spacing of 5-10 km and a 20 km x 20 km area with grid spacing of 1 km or less, and

surface-based observations including ice cores, seismics and fine-resolution radars.

CRISIS science research presents a substantial challenge for PolarGrid. The traditional view regarding the relationship of polar ice sheets and climate has been that polar ice sheets respond slowly to changes in climate. Most existing models, including those used by the IPCC, predict only a small contribution from Greenland and Antarctica to global sea level over the next century [11]. However, evidence for active ice sheets, both present-day and in the past, has mounted, and we now understand that large ice sheets can undergo rapid changes [10]. For example, analysis of deep-sea sediment cores has revealed major oscillations in the Laurentide Ice Sheet volume, believed to be associated with surges of the Hudson Bay lobe [5]. During meltwater pulse 1A, *ca.* 14,000 years B.P., sea levels rose about 20 m over a period of 400 years at an average rate of 5 cm/yr [8] compared to the rate of rise of 1-2 mm/yr during the 20th century [6].

The observed rapid changes cannot be explained by conventional models, and thus critically important predictions about the future responses of polar ice sheets to climate change are certain to be wrong in potentially disastrous ways. The existing models are based on a shallow-ice approximation in which driving forces are balanced by friction at the glacier bed. The models incorporate basal sliding where the bed is lubricated or for deformation of a soft sediment layer. However, they do not fully capture the controlling dynamics of ice streams and outlet glaciers, and do not allow for rapid along-flow propagation of force perturbations, including those resulting from removal of peripheral ice shelves.

SAR imaging of the ice-bed interface has the potential to make a significant contribution to glaciology by providing insights regarding the conditions at this critical boundary, including terrain relief and roughness as well as presence and distribution of liquid water. However, SAR image formation is computationally intensive. While the raw radar data, called phase histories, can be collected and stored for post-processing analysis, the image formation process requires a significant amount of signal processing, regardless of whether it is done in real-time or in post-processing. SAR imaging of the ice-bed interface through 3-km thick ice may require iterative focusing to compensate for the unknown variability in the electrical properties of the intervening ice.

Initial CRISIS SAR data collection flights may produce in excess of 6 million complex samples per second per receiver channel. A single 5-hour flight mission could collect as much as one terabyte of raw SAR data. Table 1 shows estimated data volume that CRISIS plans to collect during the 2008 and 2009 field seasons.

Polar science is an exemplar of the type of science that will benefit from an advanced Cyberinfrastructure. Cyberinfrastructure enables a dynamic Data-Information-Knowledge pipeline accelerating new scientific discovery. This will result in better quality predictions rendered in shorter periods of time. Some of the hardware acquisitions will be quite novel to meet the challenges of operating an IT infrastructure in the polar regions.

The impact of the “data deluge” has motivated substantial progress in the areas of data management, real-time sensors and data assimilation algorithms. Other important general Grid research areas of relevance to PolarGrid include scientific gateways [9] to manage the use of Grid-based services. A critical matter is the development of tools that enable polar scientists to make use of sophisticated supercomputers and Grid computing systems without first having to develop deep expertise in computational science themselves. The

PolarGrid implementation and development will focus on the following key areas sketched in Figure 1.

a) Field data collection systems to be taken with Polar Science researchers as they collect data.

b) A base camp 64-core cluster, allowing near real-time analysis of radar data by the polar field teams.

c) A large 17 Teraflops cluster, to be integrated with the NSF-funded TeraGrid, to serve the polar research community in final processing of data and supporting simulations. This is split between IU and ECSU to support research and education/training respectively.

d) The implementation of new improvements to the speed and capabilities of ice sheet models, and implementation of web portals to make the data, data products, and resources easily accessible to the polar research community.

e) An educational videoconferencing Grid to support educational activities.

We propose two sets of field and base camp equipment, to simultaneously support operations in the Arctic and Antarctic. An initial prototype system from IU’s commitment will be used to confirm the design of the production expedition Grids. It will be used later in other applications where we face similar power and bandwidth challenges.

IU will lead the design and development of PolarGrid software based on their experience with the CIMA [1] [2], LEAD [3] and QuakeSim [4] projects which cover key areas of remote instruments, sensors, Geographical Information Systems (GIS), simulation, and data assimilation. The entire new area of eScience, in which IU is a leader, aims to develop tools by collaboration between computer and discipline scientists to put the tools needed to rapidly make new discoveries into the hands of discipline scientists. We include a discussion of this software in the instrument section as the Cyberinfrastructure will only be effective if we link hardware and Grid-technology based software.

4 CONCLUSIONS

The PolarGrid we propose will transform the capabilities of US polar researchers by enabling them to process and evaluate data as they are collected. This capability is achieved through a combination of laptop computers carried by researchers as they collect data in the field, a 64-core computer cluster to be added to the polar research base camp, and a larger cluster located in the continental US for final data processing and modeling work. The 64-core base camp computer cluster is a system sufficiently powerful that just half a decade ago it might have been considered a university’s major supercomputer. This system must be deployed in the field; much of the data will be collected with Synthetic Aperture Radars (SAR) and will require considerable post-processing; and low network bandwidth between the polar regions and larger supercomputer facilities will preclude real-time supercomputing at centers such as the TeraGrid. However, there will be considerable post-collection analysis and scientific investigation on the collected data that will require the resources of the TeraGrid.

Video conferencing hardware and an educational grid is also included as a part of this project to aid in the dissemination of new research findings to students, particularly from traditionally underserved groups, at our partner institutions.

REFERENCES

- [1] Bramley, R., Chiu, K., Devadithya, T., Gupta, N., Hart, C., Huffman, J.C., Huffman, K.L., Ma, Y., McMullen, D.F. (2006), *Instrument Monitoring, Data Sharing and Archiving Using Common Instrument Middleware Architecture (CIMA)*. Journal of Chemical Information and Modeling, 46(3) p.1017-25, May-June.
- [2] Devadithya, T., Chiu, K., Huffman, K., McMullen, D.F. (2005), *The Common Instrument Middleware Architecture: Overview of Goals and Implementation*. Proceedings of IEEE International Conference on e-Science and Grid Computing (e-Science 2005), December 5-8, Melbourne, Australia.
- [3] Plale, Beth, Dennis Gannon, Jerry Brotzge, Kelvin Droegemeier, James F. Kurose, David McLaughlin, Robert Wilhelmson, Sara Graves, Mohan Ramamurthy, Richard D. Clark, Sepi Yalda, Daniel A. Reed, Everette Joseph, V. Chandrasekar (2006), *CASA and LEAD: Adaptive Cyberinfrastructure for Real-Time Multiscale Weather Forecasting*, IEEE Computer 39(11): 56-64. Available at: <http://kkd.ou.edu/CASA-LEAD-IEEE.pdf>.
- [4] Galip Aydin, Ahmet Sayar, Harshawardhan Gadgil, Mehmet S. Aktas, Geoffrey Fox, Sunghoon Ko, Hasan Bulut, and Marlon E. Pierce (2006), *Building and Applying Geographical Information System Grids*. Special Issue on Geographical information Systems and Grids based on GGF15 workshop, Concurrency and Computation: Practice and Experience, to be published. Available from http://grids.ucs.indiana.edu/ptliupages/publications/GISGrids_Concurrency_submitted.pdf
- [5] Broecker, W.S. (1994), Massive iceberg discharges as triggers for global climate change. *Nature* 372, 421-424.
- [6] Church, J.A., and J.M. Gregory (2001), Changes in Sea Level. In: *Climate Change 2001: The scientific basis (eds. Houghton, J.T. and others)*. Cambridge U.K.: Cambridge University Press, 639-693.
- [7] Gogineni, S., J. Paden, T. Akins, C. Allen, P. Kanagaratnam, D. Braaten, and K. Jezek (2006), "Synthetic Aperture Radar Imaging of Ice-bed Interface," AGU Fall Meeting, San Francisco, CA, December 11-14.
- [8] Paden, J. (2006), "Synthetic Aperture Radar for Imaging the Basal Conditions of the Polar Ice Sheets," Ph.D. dissertation, The University of Kansas.
- [9] Alameda, Jay, Marcus Christie, Geoffrey Fox, Joe Futrelle, Dennis Gannon, Mihael Hategan, Gopi Kandaswamy, Gregor von Laszewski, Mehmet A. Nacar, Marlon Pierce, Eric Roberts, Charles Severance, and Mary Thomas (2006), *The Open Grid Computing Environments collaboration: portals and services for science gateways*. DOI: 10.1002/cpe.1078. Available from <http://www3.interscience.wiley.com/cgi-bin/fulltext/113391287/PDFSTART>.
- [10] Hansen, Jim, "Global Warming: Is There Still Time to Avoid Disastrous Human-Made Climate Change? i.e. Have We Passed a 'Tipping Point'?" Discussion on 23 April 2006, National Academy of Sciences, Washington, DC. Available from http://www.columbia.edu/~jeh1/nas_24april2006.pdf.
- [11] Connor, Steve, "If we fail to act, we will end up with a different planet," *The Independent*, January 1, 2007. Available from <http://news.independent.co.uk/environment/article2116874.ece>.
- [12] M. R. Marlino, T. R. Sumner, and M. J. Wright (2004). Geoscience Education and Cyberinfrastructure. Report of a workshop sponsored by the National Science Foundation, April 19-20, Boulder, CO: Digital Library for Earth System Education (DLESE) Program Center; University Corporation for Atmospheric Research (UCAR), 43p. Available at: <http://www.dlese.org/documents/reports/GeoEd-CI.html>.

Installation and Implementation of a LAMP Documentation Server

Tyrone Whitehurst
ECSU Grid Team
Norfolk State University
Norfolk Virginia, USA

Illiana Thomas
Andrea Grumbles
ECSU Grid Team
Elizabeth City State University
Elizabeth City, USA

Amanda Bland
Camden Hearn
ECSU Grid Team
Mississippi Valley State University
Itta Bena Mississippi, USA

Abstract— The focus of the ECSU TeraGRID team for the summer of 2007 is to setup a documentation platform for a Condor-based GRID to be established at Elizabeth City State University. This documentation platform will be a Linux based web sever that will utilize Web 2.0 standards to create a virtual documentation web portal. Grid computing itself is the creation of a "virtual supercomputer" by using a network of geographically dispersed computers. In order to create such a network infrastructure, documentation is critical to communicate with the users, and the maintainers of the systems.

The web server will utilize an Ubuntu Linux kernel with an Apache web server, a MySQL Database, a PHP scripting package, and a Media Wiki web interface. This particular setup is called a LAMP server. LAMP is the acronym for Linux, Apache, MySQL, and PHP, which are all open source applications. The combination of these LAMP applications allows Media Wiki to function.

Once installed, the documentation server will be utilized to first display the information on how to create the document server. The documentation server will then later be used to document how to setup a Condor-based GRID system. (*Abstract*)

I. LAMP

LAMP server is a type of software bundle found in a specific server. LAMP stands for Linux, Apache, MySQL, and PHP. The P can also stand for Perl or Python but generally it is assumed to be PHP. It is a solution stack of software programs, commonly open-source programs, used together to run dynamic websites or servers. Linux is the operating system (sort of like a windows system). Apache is the web server used. MySQL is the database management system. PHP, Perl, or Python is used as the programming language for the system. Though the originators of these open source programs did not design them all to work specifically with each other, the combination, has become popular because of its low acquisition cost and because of the ubiquity of its components (which come bundled with most current Linux distributions). When used in

combination they represent a solution stack of technologies that support application servers.

II. LINUX

A. Description

Linux is a Unix-like computer operating system. It is one of the most popular of free software and open source development available. Its primary source code can be modified, used, and redistributed by anyone without any restraints.

Largely known for its use in servers, it is used as an operating system for a variety of computer hardware, such as desktop computers, supercomputers, and embedded devices such as mobile phones and routers.

Most Linux distributions support many programming languages. Primary system software such as libraries and basic utilities are usually written in C. Enterprise software is often written in C, C++, Java, Perl, Ruby, or Python. The most common collection of utilities for building both Linux applications and operating system programs is found in the GNU toolchain, which includes the GNU Compiler Collection (GCC) and the GNU build system. Most Linux distributions also support Perl, Ruby, Python.

Linux seeks for interoperability with other operating systems such as Microsoft Windows and Mac OS X, and by extension the software that runs on Linux aims for interoperability with other Linux and non-Linux software. As an operating system underdog competing with mainstream operating systems, Linux cannot rely on a monopoly advantage. Therefore, in order for Linux to be a convenient operating system for users, it must interact well with non-Linux computers. Interoperability also provides users the choice of software and data formats without restricting them as a result of that choice.

Linux includes, by default, a command line interface (CLI) as part of its Unix-like functionality. Distributions specialized for servers may use the CLI as their only interface, since without a graphical user interface (GUI) the systems resource consumption is reduces. As well, Linux machines can run without a monitor attached. These "headless systems" can be controlled either through X11 remotely, or the CLI must be used through a protocol such as SSH or telnet.

Under Linux, desktop software such as word processors, spreadsheets, email clients, and web browsers are available. The following are some of Linux's major desktop applications:

Office: OpenOffice.org. It may be useful to compare office suites.

Internet: aMule, Azureus, Evolution, Firefox, Thunderbird, Opera, Pidgin.

Multimedia: Amarok, MPlayer, Totem, VLC, Xine, XMMS.

Graphics: Blender, GIMP, Inkscape, Scribus.

B. Ubuntu

Under the umbrella of Linux falls Ubuntu. Ubuntu is an African word meaning "Humanity to others", or "I am what I am because of who we all are". The Ubuntu distribution is known for "bringing the spirit of Ubuntu to the software world."

Ubuntu is a free of charge, community developed operating system that is designed for laptops, desktops and servers. Ubuntu contains the applications needed for such as word processing, email applications, web server software, and programming tools. Every six month a new desktop server is released as well free security updates for eighteen months.

Ubuntu is available for PC, 64-Bit and Mac architectures. CDs require at least 256 MB of RAM. Installation requires at least 4 GB of disk space.

With the installation disk, we were able to download the complete working environment which took a total of twenty-five minutes. Once we installed our system it was immediately ready to use.

The Server Edition of Ubuntu is built on the solid foundation of Debian which is known for its strong server installations and reliable performance.

In 15 minutes, the time it took to install Ubuntu Server Edition, we were also able to install the LAMP (Linux, Apache, MySQL and PHP) server. This feature, exclusive to Ubuntu Server Edition, is available at the time of installation.

The LAMP option meant that we did not have to install and integrate each of the four separate LAMP components, a process which could have taken hours. Instead, we got increased security, reduced time-to-install, and reduced risk of misconfiguration, all of which resulted in a lower cost of ownership.

With Ubuntu Desktop Edition we are able to surf the web, read email, create documents and spreadsheets, and edit images.

When we started the system for the first time we got to see a desktop that was clean and tidy, no desktop icons, and a default theme that is easy on the eye.

Once Ubuntu was installed, all the basics were in place so that the system was immediately usable. Ubuntu's OpenOffice contains a user interface and feature set that is similar to other office suites, and includes the all the key desktop applications needed, including: (1) Word processor - for anything from writing a quick letter to producing an entire book; (2) Spreadsheet - a tool to calculate, analyze, and present your data in numerical reports or charts; and (3) Presentation - an easy, and powerful tool for creating effective multimedia presentations.

C. Procedure

Ubuntu 7.04—also known as Feisty Fawn—is known for its reliable software management tools and catalog of ready-to-install free software applications.

Ubuntu LAMP server installed the following versions: Ubuntu Feisty Fawn 7.04, Apache 2.2.3, MySQL 5.0.38, and PHP 5.2.1.

The first step necessary is to download the server version of Ubuntu onto a CD and proceed to boot the CD. Once it has started booting you will see the first screen and select the second option to "Install to the Hard disk Option" and press enter. > Next choose the language and press enter> Choose your location and press enter > If you wanted to try to have the keyboard layout detected by pressing a series of keys you need to select the "yes" option and if you didn't want that and wanted to choose from a list click no, [we selected NO]> Select Origin of keyboard and press enter> Select keyboard layout and press enter.

The following download screens will appear: Detecting hardware to find CD-ROM Drivers in progress> Scanning CD-ROM in Progress> Loading additional components progress bar> Configures the network with DHCP if there is a DHCP server in the network>

Next enter the Hostname of the system: [we set it ubuntu-lamp].

The following download screens will appear: Detecting Disks and Hardware in progress> Starting up the partitioner in progress> Next you must partition the hard disk. [We selected "use entire disk option"] and press enter> Warning message about data lost on hard disk.

The following download screens will appear: Creating partitions in hard disk> Write the changes to disk option: select yes and press enter> Creating ext3 file system in progress> Configuring the clock option [since we wanted to leave UTC we selected yes] and press enter> Next you enter the Full name of the user [administrator] you want to create for the server select continue and press enter> Enter username for account continue and press enter> Enter the password for administrator user select continue and press enter> Confirm the password for administrator user select continue and press enter.

The following download screens will appear: Installing the base system in progress> Configuring package mirror; this is related to the country option.

Now it Installs software and select the server options if DNS or LAMP is desired [we chose LAMP for our LAMP server installation].

The following download screens will appear: Software Installation in Progress> Installing GRUB Boot loader in progress> Installation complete message will appear and the CD will eject> Select continue and press enter it will reboot the server> After rebooting, the screen will prompt for username> This completes the Ubuntu LAMP Server Installation and the server is ready for installing applications which supports Apache, MySQL and PHP.

Once the following steps are complete, use the code "sudo apt-get update" to update the operating system. Then by re-inserting the operating cd into the cd drive use the code "sudo apt-get install ubuntu-desktop" to install the ubuntu GNOME desktop.

III. APACHE

A. Description

Robert McCool created the first Apache web server. He was involved with the National Center for Supercomputing Applications web server, also known as NCSA HTTPd. Robert McCool left the HTTPd in the mid 1994. Development of the server technology stalled as a result of his absence. Several other people assisted McCool with his development of the server. The co-creators of Apache are Brian Behlendorf, Roy T. Fielding, Rob Hartill, David Robinson, Cliff Skolnick, Randy Terbush, Robert S. Thau, Andrew Wilson, Eric Hagberg, Frank Peters, and Nicolas Pioch.

Version 2 of the Apache server was a substantial re-write of much of the Apache 1.x code, with a strong focus on further modularization and the development of a portability layer, the Apache Portable Runtime. The Apache 2.x core has several major enhancements over Apache 1.x. These include UNIX threading, better support for non-Unix platforms, a new Apache API, and IPv6 support. The first alpha release of Apache 2 was in March 2000, with the first general availability release on April 6, 2002. Version 2.2 introduced a new authorization API that allows for more flexibility. It also features improved cache modules and proxy modules.

Apache is primarily used to serve both static content and dynamic Web pages on the World Wide Web. Many web applications are designed expecting the environment and features that Apache provides. It is the web server component of the popular LAMP web server application stack along with Linux, MySQL, and the PHP/Pearl/Python programming language.

Apache is redistributed as part of various proprietary software packages including the Oracle RDBMS or the IBM Web Sphere application server. Mac OS X integrates Apache as its built-in web server and as support for its WebObjects application server. Borland in the Kylix and Delphi development tools also supports it in some way. Apache is included with Novell Net Ware 6.5, where it is the default web server.

Apache is used for many other tasks where content needs to be made available in a secure and reliable way. One example is sharing files from a personal computer over the Internet. A user who has Apache installed on their desktop can put arbitrary files in the Apache's document root, which can then be shared.

Programmers developing web applications often use a locally installed version of Apache in order to preview and test code as it is being developed. The Apache HTTP Server is a web server notable for playing a key role in the initial growth of the World Wide Web. Apache was the first viable alternative to the Netscape Communications Corporation web server and has since evolved to rival other Unix-based web servers in terms of functionality and performance. Since April 1996 Apache has been the most popular HTTP server on the World Wide Web. Apache is developed and maintained by and maintained by an open community of developers under the auspices of the Apache Software Foundation. The application is available for a wide variety of operating systems including Microsoft Windows, Novell Net Ware and Unix-like operating systems such as Linux and Mac OS X. Released under the Apache License; Apache is free and open source software.

B. Procedure

The step to installing apache to the grid was to locate a suitable version of the file. To locate the file type in your web browser www.apache.org and it should take you to the apache web site where you can download the version that best suites your computer. After downloading apache the next step is to unzip and extract the file. Double clicking on the file, which looks like a box, does this. The file itself is located on your desktop. Once opened, just click extract and Ubuntu will extract your files and place them in a folder.

Now that the files have been extracted it is time to open the terminal window. If you do not have administrative privileges you will have to use sudo code to trick the computer into thinking that you have administrative privileges. The first thing you must do when the terminal window opens is type `ls` onto your keyword. Ls (list) tells the computer that you want to see a list of all of the files that are present in its database. Once the computer has generated a list of all the files you must find the name of the computer among those files. The computer we used name was Desktop. Desktop is the default name of all computers that use Ubuntu. The next step is to type `cd` (change directory) and Desktop. This tells the computer to move its focus to Desktop. Once inside Desktop type `ls` to list all of the applications inside of Desktop. Depending on the type of apache you downloaded the name should be written in blue. Once you have located the files you must change directory again to that file. For us the file name was `httpd-2.2.4`. Type `ls` again then type `./configure --enable-so`. The computer should start configuring the files so it can install them. Once it finishes it may have run into a problem with the c compiler.

At this point in time you must click on system, administrator, synaptic package manager and click on the search option. You will have to type in the administrator password. If you do not know where to find administrator it is located on the tool bar at the top of the screen under system.

At this screen move to the search window and type in `gcc`. Check everything that deals with `gcc` and click install. Go back to the terminal window and type in `./configure--with-x-toolkit=xaw`. From there type in `config.log` following that `sudo make install`, `sudo ./configure`, and `sudo ./make`. Once the terminal window has come to a stop everything should have installed itself.

IV. MYSQL

A. Description

MySQL is a key part of LAMP, and a fast growing open source enterprise software stack. More and more companies are using LAMP as an alternative to expensive proprietary software stacks because of its lower cost and freedom from lock-in. It was first released internally on May 23, 1995, but it wasn't until January 8, 1998 that the version for Windows 95 and NT was released. More releases of the beta version have been made and distributed to various locations. The latest version is version 5.1. Libraries for accessing MySQL databases are available in all major programming languages with language-specific API's. In addition, an ODBC interface called MyODBC allows additional programming languages that support the ODBC interface to communicate with a MySQL database, such as ASP or Cold fusion. The MySQL server and official libraries are mostly implemented in ANSI C.

The primary goals that have been set for MySQL include: to be the best and most used database in the world, to be available and affordable for all, easy to use, to be continuously improved while remaining fast and safe, to be pleasurable to use, and free from bugs.

MySQL is a multithreaded, multi-user SQL database management system. It has more than 10 million installations. MySQL is owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, which holds copyright to most of the code base. The company develops and maintains the system, selling support and service contracts, as well as proprietary-licensed copies of MySQL, and employing people all over the world who collaborate via the Internet. MySQL AB was founded by David Axmark, Allan Larsson, and Michael Widenius. The CEO is Marten Mickos.

MySQL is popular for web applications and acts as the database component of the LAMP, MAMP, and WAMP platforms and for open-source bug tracking tools like Bugzilla. Its popularity as a web application is closely tied to the popularity of PHP, which is often combined with MySQL. PHP and MySQL are essential components for running the popular Word Press blogging platform.

MySQL works on many different platforms. The platforms include Mac OS X, Solaris, SunOS, Windows XP, Windows Vista, etc. MySQL is written in C and C++. It has an embedded database that allows enterprises to bring their applications and solutions to the market faster. MySQL provides standards-based drivers for JDBC, ODBC, and .Net enabling developers to build database applications in their language of choice. In addition, a native C library allows developers to embed MySQL directly into their applications. The MySQL Migration Toolkit is a powerful framework that enables a user to quickly migrate their proprietary databases to MySQL. Using a Wizard-driven interface, the MySQL Migration Toolkit implements a proven methodology and walks the user through the necessary steps to successfully complete a database migration project.

B. Procedure

- 1.) Go to <http://www.mysql.org/>
- 2.) Click on Downloads on the far-left side
- 3.) Go down until you see MySQL Community Server and click on download
- 4.) Out of the list choose [Ubuntu 6.06 LTS \(Dapper Drake\)](#)
- 5.) Then beside **Client (x86)** click *pick a mirror*
- 6.) Follow the screen downloads by clicking Next

V. PHP

A. Description

PHP is a programming language that was originally designed for producing dynamic web pages. PHP is used mainly in server-side scripting, but can be used from a command line interface or in standalone graphical applications. The main implementation is produced by the PHP group and released under the PHP License. The Free Software Foundation considers PHP to be free software.

The Danish programmer Rasmus Lerdorf wrote PHP as a set of CGI binaries in the C programming language in 1994. It was made to replace a small set of Pearl scripts he had been using to maintain his personal homepage. His original reason for designing PHP was to use

it to display his resume and to collect certain data about the number of people that viewed his web page. PHP (Personal Home Page Tools) was released on June 8, 1995 after Lerdorf combined it with his own form interpreter to create PHP/FI.

In 1997 Zeev Suraski and Andi Gutmans rewrote the parser and formed the base of PHP3 changing the language's name to the suit it a Hypertext Preprocessor. The development team officially released PHP/FI 2 in November of 1997 after months of beta testing. Public testing of PHP 3 began and the official launch came in June of 1998. Suraski and Gutmans then started a new rewrite of PHP's core producing the Zend Engine in the year of 1999. They also founded Zend Technologies, which actively manage the development of PHP.

In May of 2000 another version of PHP was released. The Zend Engine powered PHP 4. On July 13, 2004 PHP 5 was released and the Zend Engine 2 powered it. It had new features like a support for object-oriented programming, performance enhancements, better support for MySQL, and support for SQLite.

PHP also provides a command line interface for developing shell and desktop applications or any other system administration task. PHP is increasingly used on the command line for tasks that have traditionally been the domain of Pearl, Python, or shell scripting. It primarily acts as a filter. The program takes input from a file or stream containing text and special PHP instructions and outputs another stream of data for display.

PHP includes a large assortment of free and open source libraries with the core build. PHP is an Internet aware system with modules built in for accessing file transfer protocol servers, database servers, and other libraries like those of MySQL and SQLite. PHP allows developers to write extensions in the C language to add functionality to the PHP language. These can then be compiled into PHP or loaded dynamically at runtime. The PHP language is kept as human-readable source code.

B. Procedure

Before installing PHP to the Linux operating system, there are certain system packages that have to be installed. After successfully acquiring the packages needed, the Apache web server software was installed in order to run and operate PHP safely. The latest version of PHP (PHP 5.2.3) was then downloaded and obtained from <http://www.PHP.net>. The software package of the PHP 5.2.3 was unzipped and extracted to another folder once found on the computer. PHP had to be installed through the command line terminal of the Linux operating system.

In the command terminal a series of commands had to be entered to install PHP. The first command the must be enter is `$ sudo apt-get install php5`. This will run a series of tests to make sure the computer has all the packages it needs. The next command is `$ sudo apt-get install libapache2-mod-php5`. This is an apache library that also needs to be installed for the proper running of the software. The following command should be `$ sudo /etc/init.d/apache2` restarted in order to restart apache. After the installation, the PHP needs to be tested by using the command `$ <?php phpinfo();?>`. Once tested, the recommended PHP modules XSLT, GD and PEAR should be installed by use of the command `$sudo apt-get install`

php5-xsl php5-gd php-pear. Once the modules are installed issue a restart to Apache for all changes to take effect.

VI. MEDIA WIKI

A. Description

MediaWiki derives from the word “wiki”, which is a web application designed to allow multiple authors to create, remove, and edit web page content using a web browser. The multiple author capacity makes them effective tools for mass collaborative authoring. One of the best-known is the online encyclopedia called Wikipedia. Also, the term refers to the collaborative software used to create such a website, known as Wiki Software, which is usually implemented as a script that runs on a web server with the Wiki content stored in a relational database management system such as MySQL. The first Wiki Software was created back in 1995.

MediaWiki is free server-based software, Wiki Software package which is licensed under the GNU General Public License and developed by contributors around the world. The name “MediaWiki” was thought of by Wikipedia contributor Daniel Mayer as a joke towards Wikipedia. It is designed to be run on a large enterprise server farms for a website that gets millions of hits per day. It is extremely powerful and scalable software. Some companies use it as a content management system.

The main benefits from MediaWiki are: free open source software, the ability to edit web pages online and check the results immediately (no need to upload or use FTP), and it is an excellent collaboration tool allowing different users to work on the one/same project.

B. Requirements

MediaWiki is built on top of an AMP environment. It typically runs in a LAMP environment using GNU/Linux operating system but it is also possible to run MediaWiki correctly under Microsoft Windows and Apple Max OS X. The software operates using one of the above operating systems, the Apache httpd (version 1.3, 2.0 or above) web server for delivering web pages, the relational database management system MySQL daemon (version 4.x or above) as a database storage type for storing the content, the scripting language PHP (version 4.3 or above) for the application logic that glues together these components, and some type of web browser. Everything needed is available as open source software.

The hardware recommended is i386 architecture (Intel, AMD, etc.) but others are possible (PowerPC w/Mac OS X), memory should be about 96 MB of RAM for best performance, a few megabytes of disk space, and a Network Interface Card.

C. Procedure

1. Go to MediaWiki's homepage (<http://www.mediawiki.org/wiki/MediaWiki>).
2. Click on Download (<http://www.mediawiki.org/wiki/Download>).
3. Click on mediawiki-1.10.0.tar.gz. (<http://download.wikimedia.org/mediawiki/1.10/mediawiki-1.10.0.tar.gz>). This is the current version.
4. Save it to disk (Desktop).
5. Now you must uncompress the file. Go to Applications -> Accessories -> Terminal

6. On Ubuntu/Linux OS to untar (uncompress) the file, type in the command: `tar xvzf mediawiki-*.tar.gz` (the * stands for all folders within the file)

Now, you must make a directory file for MediaWiki, change permissions, and copy the folders extracted from the downloaded file into the new directory.

1. In the Terminal, type in: `cd /var/www`
2. `sudo mkdir wiki` (mkdir is make directory, and wiki is what I named the file that MediaWiki was going to go into; this command resulted in creating the file "wiki")
3. `sudo chown -R apache:apache /var/www/wiki/*` (changes permissions for the config directory; owner to apache)
4. `sudo cp -r /home/administrator/Desktop/mediawiki-1.10.0/* /var/www/wiki` (copies folders into wiki file; it is very important to use the * because if you don't then it will likely not copy all the files/folders to the wiki)
5. Minimize the Terminal window and go to: <http://localhost/wiki>
6. It should show the MediaWiki icon and the version being used. It said I needed to set up the wiki first (and provided a link to the setup process).

To check to make sure the database is running properly you must sign in with the root password.

1. Terminal and typed in the command, `sudo mysql -u root -p`
2. Type in password: `ure_mysql`
3. When it gave me the prompt of `mysql ->` I typed in `status` and hit Enter
4. It showed that the database was working.

Setting up MediaWiki.

1. Go to Terminal. Enter command: `cd /var/www/wiki`
2. `sudo chmod a+w config` (changes configurations so setup process can be done)
3. Go back to <http://localhost/wiki> and click set up.
4. It ran a process of checking the environment and showed this after it was done:
 - Don't forget security updates! Keep an eye on the [low-traffic release announcements mailing list](#).

Checking environment...

Please include all of the lines below when reporting installation problems.

- PHP 5.2.1 installed
- Found database drivers for: MySQL
- PHP server API is apache2handler; ok, using pretty URLs (index.php/Page_Title)
- Have XML / Latin1-UTF-8 conversion support.
- Session save path appears to be valid.
- PHP's `memory_limit` is 128M.

- Couldn't find Turck MMCache, eAccelerator, or APC. Object caching functions cannot be used.
- Found GNU diff3: /usr/bin/diff3.
- Found GD graphics library built-in, image thumbnailing will be enabled if you enable uploads.
- Installation directory: /var/www/wiki
- Script URI path: /wiki
- Environment checked. You can install MediaWiki.

Next, setup the site configurations (on the same page, below the environment information).

5. Name: ECSUGridWiki
6. Contact e-mail: jeaimehp@gmail.com
7. Language: English
8. Copyright/License: No license metadata
9. Admin username: ECSUGridWiki
10. Password: uregrid
11. No caching.
12. For all of the e-mail, I enabled all of the options.
13. Database configuration:
 - a. Type: MySQL
 - b. Server: localhost
 - c. Name: wikidb
 - d. User: wikiuser
 - e. Password: uregrid
 - f. Port: 5432
14. Superuser account
 - a. Username: root
 - b. Password: ure_mysql
15. MySQL Specific Options
 - a. Database Table Prefix: ecsuwiki

To edit the configuration file for MediaWiki the following commands are required to access it:

1. cd /var/www/wiki
2. sudo gedit LocalSetting.php

VII. HOW TO USE MEDIA WIKI

To access the MediaWiki go to: <http://10.24.5.58/wiki> . It takes you to the homepage (http://10.24.5.58/wiki/index.php/Main_Page).

Pages are created by MediaWiki's wikitext format. Wikitext (also known as wiki markup) allows users to easily modify pages, temporarily publish random sentences (to see how they look), and even temporarily delete a page in a wiki. It helps users edit these pages without any knowledge/skill of programming (XHTML and CSS is used).

To create a page:

1. Go to log in/create account.
2. Log in as the Administrator (ECSUGridWiki).
3. Go to the address bar and behind Main_Page, type in a colon (":" , without the parenthesis and no spaces). For example: Main_Page:URE

To edit a page:

1. After you have created the page, click on edit this page.
2. Simple editing tips
 - To create a link to a webpage already on the wiki, you need to type the name of the webpage on wiki first with two brackets before it, and put the | in-between it and the name of the page, ending with two brackets and no spaces. For example: [[Main_Page|Main page]]
 - To create a web link outside of the wiki, you need to type/copy & paste the URL with a bracket before it, and then enter a space between the URL and the name of the page. For example: [http://www.yahoo.com Yahoo]

To access a page simple type in the name of the page you are looking for in the search bar. For example: MediaWikiCheckingEnvironment and it will bring up the page.

When a user submits an edit to a page, MediaWiki writes it to the database, but without deleting the previous versions of the page. This allows the user and/or administrator to easily go back and refer to the older content, incase spamming or vandalism has occurred. This content is found on the Recent Pages link on the left corner.

MediaWiki manages image and multimedia files, which are stored in the filesystem. For the larger wikis with a lot of users, MediaWiki supports caching and can easily be coupled with Squid proxy server software.

Some of its key features include links, namespaces, subpages, categories, templates, and media content. Links are created by surrounding words with double square brackets, and any spaces between them are left intact. Namespaces are prefixes before a page title (ex: User: or Talk:) which allows a page to exist under multiple names, but serve a different purpose depending upon their prefix. For example: [[The Man]] could be describing the movie with Samuel L. Jackson and Eugene Levy but [[User:The Man]] could be a profile describing a user who goes by this nickname. Each page has an associated Talk: page which can be used to discuss its contents. Subpages are a simple feature that provides automatic back-links from a page of the pattern [[Page title/Subpage title]] to the component before the slash (ex: Page title). Categories are user-created, similar to tags used in many web applications by the ones in MediaWiki are hierarchical and descriptive. The templates are text blocks that can be loaded inside another page whenever that page is requested. A template "tag" is simple a special link in double curly brackets (ex: {{corrupted}}) which calls the template (ex: Template:Corrupted) to load where the tag is. Templates support parameters, meaning parts of the text can be substituted for each specific use case. Media content (which is what MediaWiki suggests) includes a wide variety of uploaded media files, and image galleries and thumbnails can be created with swift easiness if everything is set up properly.

VIII. FUTURE WORK

Recommendations for future work include setting up a four-node, Condor-based GRID using the documentation server created to document future procedures. Work has already begun on this 4-node GRID. As of July 12, 2007, the TeraGRID team currently has a three-node mix platform grid. In terms of condor, nodes are referred to as "workers"

and servers referred to as “masters.” This platform grid consists of two Red Hat Linux nodes and one OSX node with a Red Hat master. All systems are using Condor 6.8.5 as a job distribution package.

IX. ACKNOWLEDGMENTS

Our project wouldn't have been a success without the help of our advisor and mentors. Thank you to Dr. Linda Hayden for the opportunity to participate into the Undergraduate Research Experience/GRID program. Thank You to Je'aime Powell and Keisha Wilkin for mentoring us through the program. Also to those who assisted us, Jeff Wood, Joal Hathaway, and Lee Hayden. Last but not least, Marlon Pierce for the understanding of the Media Wiki and documentation server and GRID specialty, thank you.

X. REFERENCES

- [1] “Lamp- How To”, Retrieved from the World Wide Web <http://lamphowto.com/>
- [2] “Ubuntu” Retrieved from the World Wide Web <http://www.ubuntu.com/>
- [3] “The Apache HTTP Server Project” Retrieved from the World Wide Web <http://httpd.apache.org/>
- [4] “The Apache Software Foundation (ASF)” Retrieved from the World Wide Web <http://httpd.apache.org/>
- [5] “MySQL AB”, Retrieved from the World Wide Web, <http://mysql.com/>
- [6] “PHP Hypertext Processor”, Retrieved from the World Wide Web <http://www.php.net/>
- [7] “MediaWiki”, Retrieved from the World Wide Web, <http://www.mediawiki.org/wiki/MediaWiki>
- [8] “Blogger”, Retrieved from the World Wide Web <http://www.blogger.com/>
- [9] “Del.icio.us”, Retrieved from the World Wide Web, <http://del.icio.us/>
- [7] “Globus Toolkit”, Retrieved from the World Wide Web, <http://www.globus.org/toolkit/>