Teraweb™: Web-based High Performance Computing

Joel Neisen, David Pratt and *Ola Bildtsén*
Network Computing Services, Inc. (MINN)
Minneapolis, MN 55415

http://www.networkcs.com/

ABSTRACT: *Using standard browsers, Teraweb provides a seamless and unified graphical environment for users of High Performance Computing (HPC) equipment. Teraweb can be used to run computational models, manipulate data files, print or visualize results, or display system usage information without having to learn cryptic command-line interfaces of operating system command.*

Introduction

Teraweb™ is a software system for web-bases remote computing. It provides users a complete set of tools to prepare, submit, and monitor work and to visualize results, all through a web browser. Accessing remote computing services through Teraweb's point-and-click interface allows users to avoid learning UNIX commands and to concentrate on what they want to do: simulation and analysis.
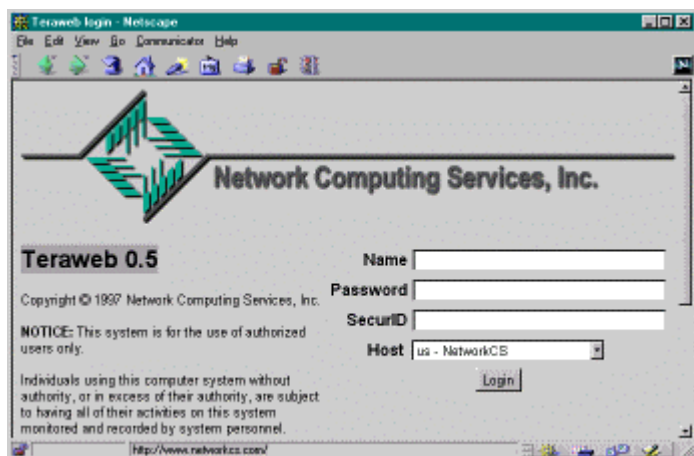

Figure 1

Teraweb also enables HPC administrators to provide services that hide the underlying differences between heterogeneous operating systems. Users see similar functionality across systems and need not remember differences between, for example, specific queuing or accounting systems. Administrators also use Teraweb to build and deliver customized, easy-to-use interfaces to the applications that are available on their systems.

Pre-History

SuperConductor: Aug 1992
SuperConductor[SM] is the name of the Graphical User Environment (GUE) service developed by Network Computing Services, Inc. (NetworkCS) for its HPC systems. The GUE consists of a set of Graphical User Interfaces (GUIs) that facilitate the use and management of files, applications, on-line documentation, help desk services, and account and resource allocations. SuperConductor runs in the X Window System environment using the OSF Motif widget package.

In many respects, SuperConductor's capabilities surpass those of Teraweb. X/Motif afforded the opportunity to use drag-and-drop gestures, context-sensitive help, and a wide variety of GUI elements. As it turns out, our experiences with SuperConductor had a large influence on Teraweb.

Wired: Feb 1996
Wired was a *think-tank* study the goal of which was to analyze the impact that new technologies would have on the HPC community. As the name implies, a large part of our efforts were devoted to understanding the implications of a ubiquitous network infrastructure.

SuperTrek: May 1996
SuperTrek was a program NetworkCS administered for the Minneapolis public school district to expose high school students to high-performance computing. A web interface was created so that students could prepare batch jobs without having to learn about the batch queuing system.

The CGI scripts would copy input files from the user's desktop (via *ftp*), create the batch input deck, and run the job. Interfaces were created for structural analysis,

computational fluid dynamics and ray-traced rendering applications.

SEG: Oct 1996
Beginning in Oct 1996, NetworkCS demonstrated interactive supercomputing at a number of trade shows. The most popular of these are the "Find the Oil" games at the Society of Exploration Geophysicists (SEG) annual meetings.

These demonstrations used JAVA and VRML programs in which a user could manipulate physical models to graphically construct an input file, submit and run the job on one of the HPC systems, and return and view the results as a VRML model.

T-ReX: May 1997
Transparent Remote eXecution (T-ReX) is a Java *application* which facilitates simple remote execution of applications on HPC systems.

A collection of Java classes, corresponding to a specific third-party application, are designed to "know" what kinds of parameters, and input and output files each applications uses. Running as a stand-alone application, that is, outside of the browser *sandbox*, T-Rex could automatically copy input files from the desktop machine to the HPC system for execution and copy output files back when the application had finished.

To simplify matters of software distribution (and other administrative matters), only a very small kernel of T-Rex needed to be installed on the user's desktop system. This application performs the necessary user authentication and log-in and then dynamically loads application-specific classes from a central server.

Teraweb

Impetus
With the advent and increasing use of workstations, NetworkCS recognized the evolution of *bi-modal* computing. That is, the alliance of big-iron and desktop systems into what is known today as the enterprise solution. As desktop systems increased in number and capability, we observed two related phenomena: 1) increasing reliance on the desktop system for pre- and post-processing analysis and 2) a growing number of *casual* HPC users; that is, users who were unfamiliar with the traditional command line interfaces and other aspects of HPC operating systems.

The pre-history of Teraweb shows that we have addressed this issue for some time. That is not to say this trend escaped the notice of others. Cray Research, Inc. pursued a number of projects to accomplish the same end; some were more application focused like MPGS, others more function-oriented frameworks such as AIT (Application Integration Toolkit).

These approaches, though successful for their intended
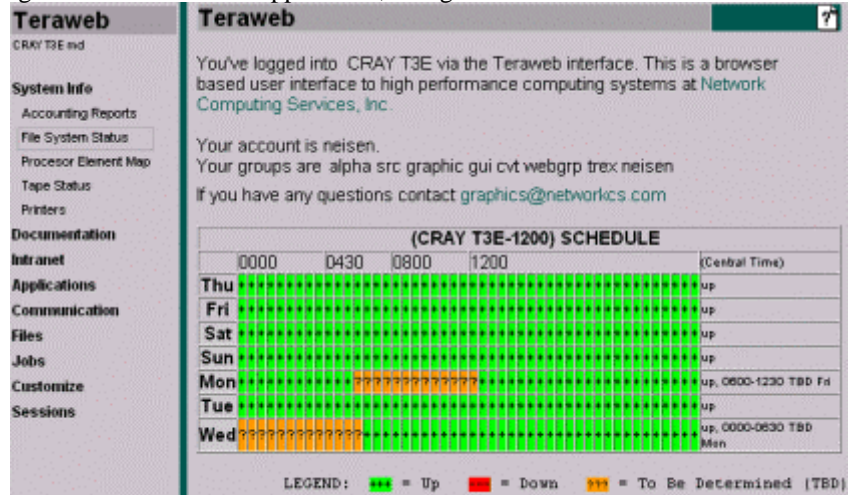

Figure 2

purpose, didn't have exactly what we wanted for broad-based remote computing framework and solid performance for real-world day-to-day activity.

Another drawback of these methods was the need to install custom client-side software. We wanted to be able to avoid this issue completely and so looked at the common capability desktop systems. We were convinced that the web-browser was the key. The browser is already installed on desktop systems, the point-and-click model is easy to use and required little or no training, and, finally, it is a "cool" technology -- people like to "surf the web".

Design
The primary concern for anyone thinking about delivery of web-based services is that of security and access protections. An oft-repeated mantra is that Teraweb shouldn't be able to do anything that the user couldn't do in a normal telnet session. We thought it essential that Teraweb preserve the natural security and access mechanisms of the underlying operating system.

A second important concern is maintaining state. To provide a sense of session coherence, in an inherently stateless protocol, the Teraweb server includes a state vector. It records a variety of selected variables to track user identity, working directory, environment variables and so forth.

We were successful in implementing an approach which meets these requirements. This approach is the subject of a pending patent application.

As a fully compliant web server, the underlying capability of Teraweb is extended using scripts written in HTML, Perl, C, and Java, just as you'd expect for any web server. In fact, it has proved to be a straightforward matter to convert existing CGI scripts for use in Teraweb.

Quick tour
Users interact with HPC systems via "Teraweb sessions" that are controlled by a web browser. Each session

represents a connection to a system. Sessions are established by opening the Teraweb Login page. Users enter a User ID and password, select a machine to connect to and press the "Login" button to connect. The Teraweb login facility can be adapted to accept other sorts of authentication information, such as SecurID card numbers.

Teraweb's functionality is accessed primarily through the menu frame at the left side of the browser window. The menu is divided into Groups and Applications. Each Group has a number of related Applications associated with it. Groups can be either expanded or collapsed by clicking on the label. Applications are run when the user clicks on its corresponding label.

The right frame, or the working area, contains the output of the application. Hyperlinks, buttons, and forms are used to access related features or to drill down into the application.

Default Menu Items
System Info Accounting Reports, File System Status, CPU Status, Tape Status

Documentation User Guide, Message of the Day, System news, Man Pages, Search Index, Cray Docs

Applications Shell Command, Gaussian, GAMESS-UK, Chemistry Visualizer

Files Directory List , File Editor, Find Files , File Upload

Jobs Current Batch Jobs , Current Processes , Users Logged In

Customization Application Editor, Application Path, Menu Editor

Feedback Survey, Problem Reports, Contact Support

Sessions Welcome Screen , List Sessions , X Authentication, Environment , Password , Teacher mode , Log Off

Help System
Our goal was to make all Teraweb pages self-obvious. We didn't want to make a system that required user training. Unfortunately, there are a number of circumstances in which the user, particularly the novice user, might need additional information.

To address these cases we introduced the help icon. Located in the upper right hand corner of the screen, this icon is a hyperlink to additional information. Quite often this is a reference to a man page or other existing on-line document.

Augmented Reports
HTML provides a fairly rich set of formatting directives. Teraweb takes advantage of this by augmenting the ASCII output with tables, font style and sizes, and color. These items make reports much more readable, and help direct the attention of the user to special information.

HTML can be used to direct the browser to automatically download a specific URL. Teraweb uses this feature to automatically refresh documents. This is very useful for users when they are monitoring their own jobs, and for operators or system administrators as they monitor aggregate use of system resources (e.g. batch queues, CPU utilization, disk use, network status or printer jobs).

Hyperlinks are a very powerful feature. Teraweb uses hyperlinks to expand the depth of its functionality. Hyperlinks in Teraweb output provide a means of progressive disclosure. The base output provides information of general utility while links provide access to variations or more detailed information, or access to related but different commands. A very simple example of this is in the display of man pages; items in the "See Also" section are hyper-linked to those man pages.

Functionality

File Access
Part of the work involved in doing simulation and analysis is in managing files, including the need to move files between local workstations and the HPC server. Teraweb provides the ability to manage files on the server system, as well as to move files between desktop and HPC systems.
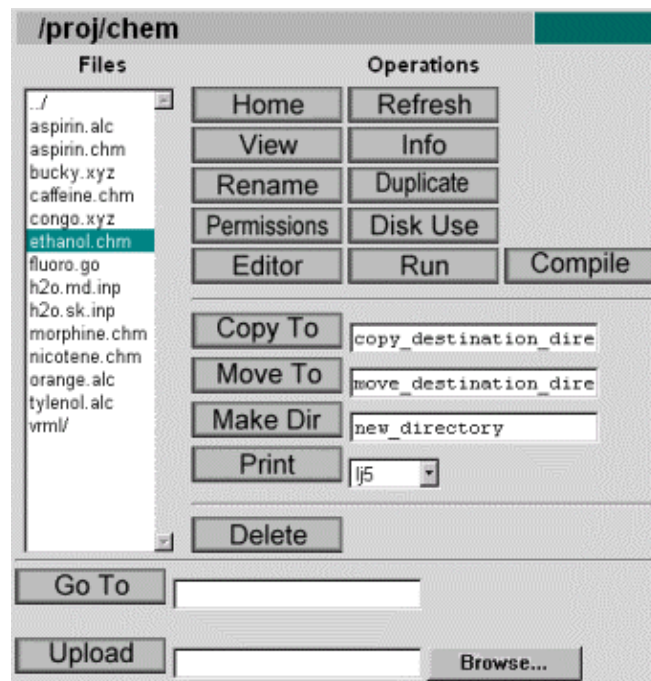

Figure 3

The file browser displays the files in the current working directory of the Teraweb session. File operations, such as copying, removing, or printing are performed by pressing buttons in the file browser that represent these actions.

An example of a common file operation is the setting of file permissions. The corresponding UNIX command is an example of a command with a particularly complicated

syntax (e.g., "chmod u=rwx,go=u-w file"). The Teraweb file permission screen presents the user with a list of files and a series of check-boxes for setting the various file permission options.

Teraweb also provides an interface to commands unique to the flexible file sharing and protection mechanisms of the distributed file system, AFS. Similar to the UNIX file permission screen, the AFS Access Control List (ACL) screen provides check-boxes for setting (for a directory) the various access control rights associated with users and groups.

Resource Monitoring

Once the user is ready to submit work, Teraweb provides tools to look at the availability of system resources and to submit and monitor jobs.
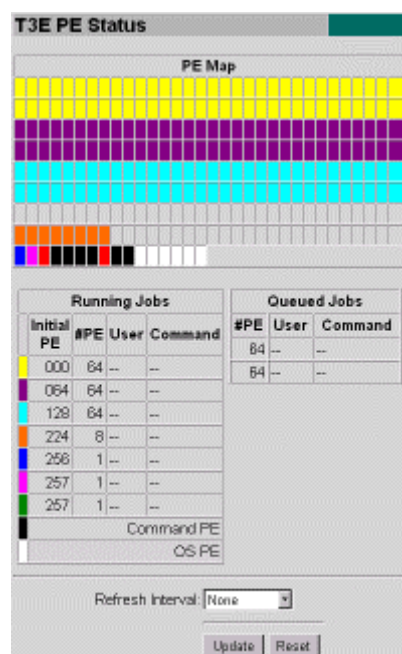
Figure 4

One example of a Teraweb system monitoring tool is the "PE Map" display that can be made available on the CRAY T3E. This display takes the output of a text-based monitoring command (grmview) and provides a graphical display of the availability of the processing elements (PEs) on the system. Teraweb uses color coding to identify individual running jobs, mapping them to the PEs running them. The "PE Map" also displays jobs that are queued and waiting to run.

Figure 7 is an example of process monitoring. It displays currently running processes according to certain parameters. Many of the fields are hyper-linked; clicking on a process id (PID), for example, adds that process to the "PID list" field. The user can select a signal to send to those processes. Clicking on the values of other fields produces a new report of processes associated with that user, parent process, or terminal session (TT).

In addition to the resources already mentioned, Teraweb is able to display CPU utilization on PVP or SMP systems, batch queues, tape drive use, printer queues, and accounting information. These provide resource availability information so that the user can make a more informed decision about how and where to run their work.

Applications

Teraweb provides customized interfaces to user applications on the HPC system. An example is the interface to the computational chemistry program Gamess, shown in the figure 5. This screen allows the user to configure a Gamess job to be run, specifying input and output files, and other job parameters such as the time limit and the number of processors to use.

Figure 5

Once the job is submitted by clicking on the "Submit" button, the user can monitor its progress using the tools previously described in the Resource Monitoring section.

The details of the underlying batch queuing system are hidden from the user. As a result the user sees the same screen across a variety of HPC systems or sites.

Teraweb front-ends for Gaussian, Gamess, XMol, Amber and several user-developed applications already exist; others are under development.

Since Teraweb is web-based, it can deliver non-traditional reports to the user. For instance, rather than looking at the ASCII text output from some application, a web-aware back-end can be created. Browsers come complete with a full repertoire of viewers and plug-ins, thus offering a rich variety of text, image, video, and audio formats that can be used to analyze the data. For example, XMol-style representation of the output of computational chemistry

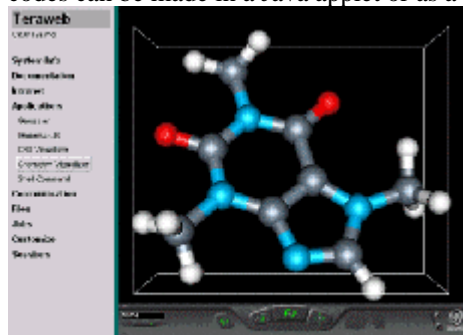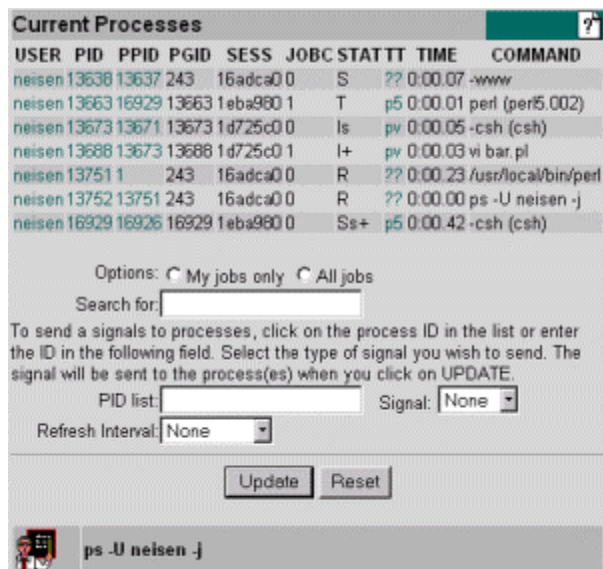codes can be made in a Java applet or as a VRML file.

Figure 6

Teacher Mode

One of the main features of Teraweb is its ability to let users run commands and applications without needing to learn UNIX commands. However, for those users who wish to learn those commands and their options, Teraweb provides a "Teacher Mode," which, when enabled, echoes the UNIX commands that it executes "behind the scenes."

Figure 7

Once "Teacher Mode" is turned on, a special entry is included at the bottom of the Teraweb page with the corresponding UNIX command. In figure 7, the user's current processes are displayed, the ps command used to generate this information is included next to the teacher icon at the very bottom of the page.

User Extensibility

From the beginning, we wanted Teraweb to be flexible enough for users to tailor it according to their own needs and preferences. To accomplish this, users have full control over the menu and are able to add their own CGI scripts.

The Teraweb Menu

The menu is actually a Java applet. The menu consists of an ordered set of groups, each group containing applications. Applications are associated with a URL that provides function.

Internally, the menu is modeled after a dictionary. When a menu item is selected, the semantics are determined by its current definition. The dictionary is initialized at start-up time by reading a system-wide configuration file. The menu applet then scans for personal menu definitions and adds those to the dictionary. Thus the user can define new groups or applications or re-define existing items. Personal menu files can include other files, so that groups of users can share from a common menu file.

To facilitate customization, Teraweb provides interfaces for users to create and edit personal menu files.

CGI Scripts

The menu only organizes Teraweb functionality, it does not supply the functionality. Just as with all web servers, the bulk of Teraweb's features are implemented as CGI scripts. Teraweb users can specify multiple locations for scripts. Thus, specific behavior can be created by the user in their own directories, or shared with other users.

Teraweb comes with a visual interface to create HTML form-based CGI scripts.

Web-aware Applications & Visualization

An important key to Teraweb's future is accessing third party applications for the remote HPC user. This, after all, is Teraweb's *raison d'être*. Scientific applications (e.g. Gaussian) were included into Teraweb early on to demonstrate its utility in this area. To facilitate our efforts in this area, we built, as part of Teraweb, tools for "wrapping" existing applications.

The wrapping for an application can done to different degrees of integration. At the simplest end, the wrapping may be little more than a simple, batch-like collection of parameters, and presentation of the output files as text. A more complicated version may post process the output files to take better advantage of the browser's multi-media capabilities. We call these "web-aware" applications.

One simpler example of a "web-aware" application is the chemistry visualizer in figure 6. Another application, created by Andrew Johnson of the Army High Performance Computing Resource Center (AHPCRC). provides a full feature, interactive, fluid-flow visualization tool called "CFD-Viz".

We have integrated several third party applications for Teraweb users and we expect to add many more applications as we go forward.

Operational Systems

To date, Teraweb has been ported to and installed on CRAY T3E, CRAY J90, SGI Origin 2000, CRAY C90, CRAY Y-MP, and CRAY-2 systems at NetworkCS. It has been installed on CRAY T3E, IBM SP, SGI Onyx and SGI Challenge systems at the Army High Performance Computing Resource Center.

Additionally, Teraweb has been licensed for use at other HPC sites.

Glossary

CGI Common Gateway Interface
Programs or scripts that are run by a web server to perform some task. These are usually addressed with URL's that begin /cgi-bin.

Cookie A mechanism which web servers (or their CGI scripts) can use to store and retrieve information in the client's browser.

HTML Hypertext Markup Language
A method for describing the logical format of text documents that can include hyperlinks and references to other multimedia files. For more information see the HTML web page (http://www.w3.org/MarkUp/).

HTTPD Hypertext Transfer Protocol Daemon
The application-level protocol used by browsers and web servers. The protocol is defined in RFC 1945.

HPC High Performance Computing, High Performance Computer

MPGS Multi-Purpose Graphics System
A distributed engineering visualization application originally developed at Cray Research, Inc., now owned by Computational Engineering International.

PVP Parallel Vector Processors

Sandbox A security model for JAVA applets designed to limit access to system resources by that applet, particularly for untrusted applets downloaded by a browser from the Internet. For instance, applets running in the context of a browser are not permitted to read, write, or otherwise modify local files.

SecurID A product of Security Dynamics, Inc. that provides one time passwords.

SMP Symmetric Multiprocessing

URL Uniform Resource Locator
A world wide web address.

VRML Virtual Reality Modeling Language
File format standard for 3D multimedia and shared virtual worlds on the Internet. For more information see the VRML FAQ (http://www.vrml.org/about/).

XMol An application developed at NetworkCS that allows researchers to view (on any X11 or OpenGL display server) 3D molecular models produced by other software packages, and to print the molecular displays in variety of formats. Molecular models can be manipulated in a variety of ways. Animations of multi-step datafiles are possible, as are the calculations of atom-to-atom distances, bond angles, and torsion angles.

Author Biographies

Joel Neisen has a MS degree in computer science/artificial intelligence and a BS in computer science/computer graphics. He is currently Director of NetworkCS's Network Interfacing department where he has led the development of several advanced visualization systems including Teraweb; XMol, a computational chemistry visualization tool; and SuperConductor, an X-based graphical interface to supercomputer systems.

David Pratt received a BS in computer science from the University of Minnesota in 1988 and a BS in chemical engineering from Michigan State in 1985. Currently a member of the Network Interfacing department, he is responsible for the design, implementation, and ongoing support of scientific visualization services, including XMol, T-ReX, and the SuperTrek project. Mr. Pratt is the primary webmaster at NetworkCS and resident Perl expert.

Ola Bildtsén has a BS degree in computer science from Amherst College and is in a graduate program in software engineering at the University of Minnesota. Mr. Bildtsén's expertise includes C and Java programming and web design. He was co-author of T-ReX.

Copyrights