Distributed Net Applications Create Virtual Supercomputers

George Lawton

The whole is greater than the sum of its parts. That, in some ways, explains the theory behind one of the most important new technological opportunities created by the Internet: distributed applications.

In distributed Internet applications, functions are distributed online to dozens, thousands, or even millions of computers. These include distributed file-sharing applications, such as the controversial Napster, in which users share MP3 music files over the Internet, and Gnutella, an open-source application that lets users share all types of files.

One increasingly important type of distributed application links a group of computers over the Internet to create a virtual supercomputer.

The most computationally intensive long-term distributed Internet application on the planet, SETI@home, which analyzes signals in space looking for signs of intelligent life, runs on more than 2 million computers and processes an average aggregate of 12 teraflops, according to the project's chief scientist, Dan Wertheimer, an astronomer at the University of California, Berkeley's Space Sciences Laboratory.

Distributed Internet applications are important because they utilize unused computer time to conduct the type of attacks on complex computing problems that used to require supercomputers,

which are very expensive to use and inaccessible to most organizations.

Running a complex computational problem costs about 1 percent as much to run on a distributed Internet application as on a supercomputer, said Scott Kurowski, founder and executive officer of Entropia, a provider of commercial and noncommercial distributed-Internet-computing services.

The concept of distributed computing has been around for years but never took off because of past limitations in such areas as networking technologies.

Now, however, more powerful processors, as well as faster communications technologies, are making it easier to link computers to perform complex distributed applications.

And the increased demand for high-performance computing by a growing number of commercial and noncommercial organizations is driving demand for distributed applications.

While the technology offers many benefits, proponents also face several important challenges and concerns. For example, researchers must develop ways to cope with such problems as digital Internet applications' high bandwidth consumption and the way Internet congestion influences the technology's effectiveness.

In addition, security is a major concern for several reasons. For example, a network for rapidly distributing application code could also spread viruses and other malicious code.

Nonetheless, proponents say, the technology has enormous promise. In fact, it has the potential to redefine the way we use the Internet, said Vern Paxson, a senior scientist at AT&T's Center for Internet Research.

Eventually, the technology will become like a utility that people will depend on, said Adam Beberg, the head of Cosm, a research project designed in part to commercialize distributed Internet applications.

THE ROAD TO DISTRIBUTED INTERNET COMPUTING

The concepts behind distributed computing have been around for years. For example, the Network File System (NFS), developed by Sun Microsystems in 1984, is a client-server-based technology that lets a user access, store, and update files on a remote machine.

And the Distributed Computing Environment (DCE), a client-server-based software technology developed by the Open Software Foundation in 1989, permits computing and data exchange in a system of distributed machines.

However, the use of DCE-based technology has been limited by the cost of the applications and licensing fees. In addition, DCE did not integrate well with many programming languages and did not work well with multiple administrative domains, which means it is most useful only in applications run within a single company, said Andrew Grimshaw, associate professor of computer science at the University of Virginia, a leader of the university's Legion distributed-computing project, and president of Applied Meta-Computing (a company working on the commercialization of Legion technology).

All this left an unsatisfied demand for distributed Internet applications.

THE TECHNOLOGY

As shown in the figure on this page, distributed Internet applications can be coarse-grained or fine-grained.

In coarse-grained applications, clients communicate with master and proxy servers but not with each other. This approach is good for brute-force computations in which the application only has to divide calculations among different clients.

In fine-grained applications, participating clients must communicate with each other. This is necessary for such applications as simulations. As simulations progress, changes in situations handled by one client affect situations handled by others. Clients thus must be able to communicate these changes to each other.

Participating in distributed Internet applications

Individuals interested in participating in distributed Internet applications download and install the necessary core software. In some approaches, the core includes the code necessary for the client to run the application. In other approaches, the core includes only the application's operational infrastructure. In these cases the participant must separately download the code necessary to run the application.

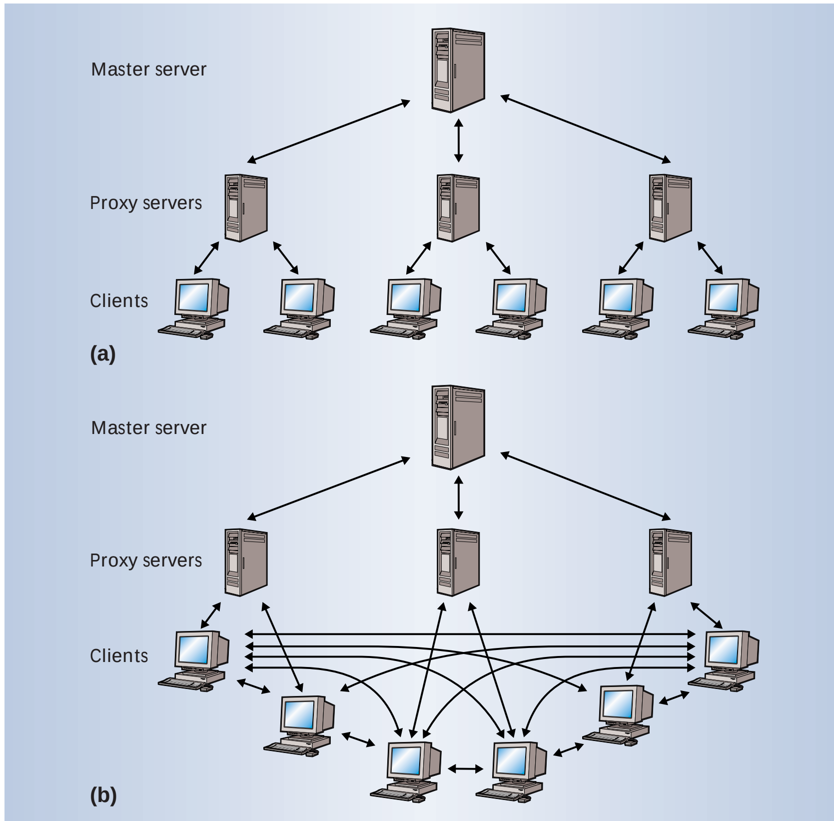
Some distributed applications run on a user's machine in the background, in connection with a screen saver. This way, the application runs only when the machine is not in use and stops running when users want to work on their machines again.

Distributed applications can also run in the open on a predetermined schedule on unused machines on a corporate network. Distributed applications could also run in the background while a user is working on a computer, but this would diminish performance.

Working across platforms

For maximum effectiveness, distributed computing technologies can run on a variety of platforms, including PCs, Macs, servers, mainframes, and even supercomputers.

In some cases, programmers may optimize the core software for a particular computing platform, such as Windows. To run on multiple platforms, developers must either write the core in a platform-neutral language, such as Java, or design a distributed platform that can run on different types of machines, as Applied MetaComputing has done.



Distributed Internet applications can be coarse-grained (a) or fine-grained (b). In coarse-grained applications, work is divided into independent pieces. The pieces are distributed to multiple clients, which return the results of their computations to a master server. Clients communicate only with a master server and perhaps proxy servers. Fine-grained applications, such as simulations, are more complex because clients must communicate with each other. This is necessary because the work done by one client affects the work done by others.

Architecture

Distributed computing applications typically use a hierarchical architecture in which a master server hands out tasks to various clients, which execute the tasks and send their work back to the server.

In some applications, such as SETI@home, a client finishes a task and requests more data from the server. In other applications, where client activity is highly predictable, servers push data on a set schedule.

In very large applications, proxy servers lighten the master server's load by handling many communication tasks. This permits scaling and load balancing, and helps distributed applications work around network problems.

Critical issues

Distributed Internet applications have had to address two critical technical issues, according to Kurowski.

Fault-tolerance and recovery. A master or proxy server monitors network communications. If a network connection goes down or a client crashes, the server sends the task to another client. To maximize fault tolerance, fine-grained applications can be run over a LAN or high-quality Internet connection, which makes recovery easier.

Communications. Distributed Internet applications have two primary communications needs. Applications such as SETI@home send a relatively small data file once a day, at the most. In other applications, such as simulations, in which different nodes frequently communicate their state to each other, a tremendous amount of communication is required.

Distributed Internet applications tend to use TCP over WANs, particularly those covering long transmission distances, because the protocol offers lower latency. However, TCP also requires more management. Therefore, over LANs (whose shorter transmission distances leave less opportunity for latency), distributed Internet applications tend to use UDP with additional control functionality to ensure that data is sent reliably.

HIGH-PROFILE PROJECTS

There are several high-profile research projects based on distributed Internet applications.

SETI@home

SETI@home (http://setiathome.ssl.berkeley.edu/) is based at UC Berkeley's Space Sciences Laboratory and is affiliated with the school's Search for Extraterrestrial Intelligence (SETI) project. The project collects data from signals generated by radiation and other sources in space, and analyzes the data for patterns that could indicate signals sent by intelligent life. The signals are received by the US National Astronomy and Ionosphere Center's Arecibo Radio Telescope in northwestern Puerto Rico. Data is sent to UC Berkeley and then to SETI@home participants.

Researchers began developing SETI@home in 1996 and launched it on 17 May 1999. According to Wertheimer, the rapidly growing project already works with 2 million computers in about 200 countries and consumes about one-third of all public Internet bandwidth at UC Berkeley. Volunteer participants have donated a quarter of a million years of computer time so far.

Great Internet Mersenne Prime Search

GIMPS (http://www.mersenne.org/prime.htm) conducts brute-force calculations to find new Mersenne prime numbers. These numbers—named after 17th century French monk, mathematician, and theologian Marin Mersenne—are primes that fit the formula $2^x - 1$. Only 36 Mersenne primes were known before software engineer and entrepreneur George Woltman started GIMPS in January 1996.

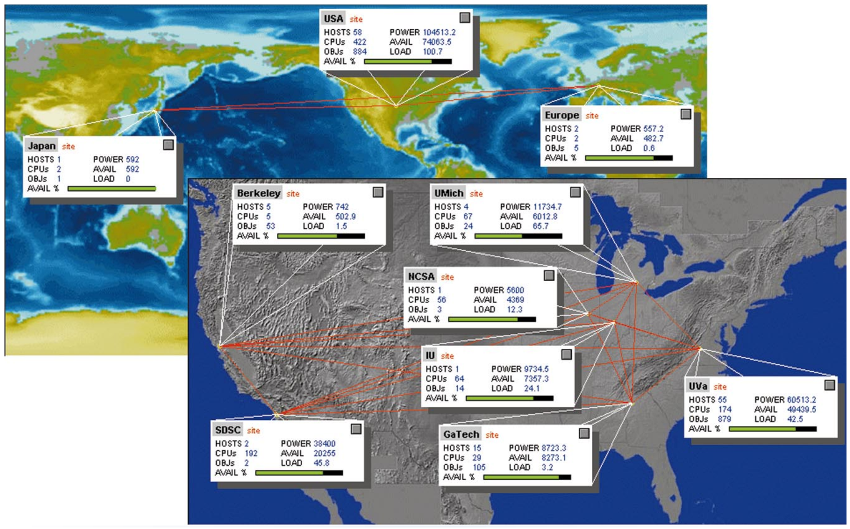
The initial GIMPS application was not distributed. In 1997, however, Entropia's

Kurowski worked with Woltman to distribute the application over the Internet. Kurowski said he did this in part to demonstrate the power of distributed Internet applications.

About 30,000 machines from about 120 countries are participating at an aggregate computational rate of about 1 teraflops.

Overall, GIMPS has found four new Mersenne primes, the longest being $2^{6,972,593} - 1$, which contains 2,098,960 digits (see the number at ftp://entropia.com/gimps/prime4.txt). According to Kurowski, finding this number without distributed-Internet-application technology would have taken GIMPS 6,500 machine years.

The first two new Mersenne primes were found before Entropia's involvement. However, Kurowski said, they were easier to find because they were so much smaller than the last two.



The Legion Status Monitor, a technology used in Applied MetaComputing's distributed Internet applications, shows information about North American and global distributed networks, including power, load, speed, and host availability. This technology provides an organization with considerable information about and control over a distributed Internet application, thereby enhancing its flexibility and functionality.

COMMERCIALIZING THE TECHNOLOGY

Several vendors—including Porivo Technologies, Applied MetaComputing, Cosm, and Entropia—recently launched plans to create businesses based on distributed Internet applications.

Companies could use these applications to conduct different types of complex calculations. Entropia's Kurowski said distributed Internet applications could conduct complex simulations and solve difficult problems in such industries as aerospace, petroleum, and entertainment. For example, pharmaceutical companies might want to run complicated chemical simulations to search for new drugs.

Vendors plan to generate revenue by charging either licensing fees for providing their technology to other companies or broker fees for running distributed Internet applications for other companies.

Porivo Technologies

Porivo (http://www.porivo.com) is developing a Java-based distributed-computing platform and plans to act as a service bureau, setting up applications that will let customers buy processing power obtained from participating computer owners.

Cofounder and CEO William Holmes said Porivo's platform will work with many different types of industries. One of the company's first applications is being designed for the healthcare market, to work with data used in such areas as clinical trials.

Holmes said that because participating clients would receive only a small part of an application, their owners could not interpret or reconstruct medical records or other sensitive data.

Meanwhile, business partners could use Porivo to integrate joint operations across a distributed application, with each performing a different part of a task.

Porivo hopes to release a beta version of its platform this summer and to ship a product in the fall.

Porivo is considering the release of a software developer's kit. Organizations could thus build applications on Porivo's platform. Porivo would then let nonprofit and public-interest organizations post their projects on a portal on its Web site, where people who want to donate computer time could sign up.

Applied MetaComputing

Grimshaw founded Applied MetaComputing (http://www.appliedmeta.com) in 1998 to commercialize the technology developed by the University of Virginia's object-based Legion research project (http://legion.virginia.edu).

A number of corporations, government agencies, and researchers are using Applied MetaComputing's technology, according to David Kopans, vice president of business development.

He said the company's technology, unlike that used in some other distributed Internet applications, provides its own OS-like environment. This gives users more control over a distributed application, he said, which is very important for complex, fine-grained applications.

CONCERNS

Distributed Internet applications must address several key challenges and concerns.

Security

Developers must minimize the chances that distributed Internet applications can accidentally distribute malicious code to participants or enable denial-of-service attacks. A massive DoS attack in 1997 helped end a major distributed-computing project.

With this in mind, Beberg said, Cosm distributed-computing technology includes DoS prevention measures.

Meanwhile, distributed applications now typically use authentication to ensure that people distributing code are authorized to do so. Applications also frequently use encryption to protect the data being transmitted and keep it from being corrupted or changed.

Bandwidth consumption

Because of the amount of communication between clients, fine-grained distributed Internet applications can consume considerable client bandwidth. Broadband Internet-access technologies, such as cable modems, would help speed transmissions but would not address quality-of-service issues.

Internet congestion

With fine-grained applications, network congestion can cause problems by disrupting the frequent transmissions between clients. This is not a big problem with coarse-grained applications, which primarily involve only short, occasional transmission bursts between servers and clients.

FUTURE

According to Kurowski, the use of distributed Internet applications could grow very quickly. He said, "This is going to be the grid of the future. Network bandwidth will increase, processing power will only increase, and the number of machines available will only increase."

Meanwhile, the growing use by consumers and businesses of computers with more powerful processors and of fast Internet-access technologies—such as digital subscriber lines, cable modems, and fiber—could improve distributed Internet applications' performance and drive their increased use.

In the future, Kurowski said, there will be more and increasingly varied distributed Internet applications that will be accessible to individuals and organizations for working with difficult problems they could not otherwise solve. This will shift high-performance computing's emphasis away from supercomputers, he said.

Porivo's Holmes said, "This is the next extension of the Internet as a computing platform, enabling communities of PC users across the globe to participate in projects that impact our businesses and lives. With ever more powerful PCs at our disposal and the growing availability of high-speed Internet access, for the first time we can truly aggregate this unused resource for valuable applications." ✷

George Lawton is a technology writer based in Brisbane, California. Contact him at glawton@best.com.

Editor: Lee Garber, *Computer*, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; l.garber@computer.org