N news

Brian Hayes

Cloud Computing

As software migrates from local PCs to distant Internet servers, users and developers alike go along for the ride.

THE GREEK MYTHS tell of creatures plucked from the surface of the Earth and enshrined as constellations in the night sky. Something similar is happening today in the world of computing. Data and programs are being swept up from desktop PCs and corporate server rooms and installed in "the compute cloud."

Whether it's called *cloud computing* or *on-demand computing*, *software as a service*, or *the Internet as platform*, the common element is a shift in the geography of computation. When you create a spreadsheet with the Google Docs service, major components of the software reside on unseen computers, whereabouts unknown, possibly scattered across continents.

The shift from locally installed programs to cloud computing is just getting under way in earnest. Shrink-wrap software still dominates the market and is not about to disappear, but the focus of innovation indeed seems to be ascending into the clouds. Some substantial fraction of computing activity is migrating away from the desktop and the corporate server room. The change will affect all levels of the computational ecosystem, from casual user to software developer, IT manager, even hardware manufacturer.

In a sense, what we're seeing now is the second coming of cloud computing. Almost 50 years ago a similar transformation came with the creation of service bureaus and time-sharing systems that provided access to computing machinery for users who lacked a mainframe in a glass-walled room down the hall. A typical time-sharing service had a hub-and-spoke configuration. Individual users at terminals communicated over telephone lines with a central site where all the computing was done.

When personal computers arrived in the 1980s, part of their appeal was the promise of "liberating" programs and data from the central computing center. (Ted Nelson, the prophet of hypertext, published a book titled *Computer Lib/Dream Machines* in 1974.) Individuals were free to control their own computing environment, choosing software to suit their needs and customizing systems to their tastes.

But PCs in isolation had an obvious weakness: In many cases the sneaker-net was the primary means of collaboration and sharing. The client-server model introduced in the 1980s offered a central repository for shared data while personal computers and workstations replaced terminals, allowing individuals to run programs locally.

In the current trend, the locus of

computation is shifting again, with functions migrating outward to distant data centers reached through the Internet. The new regime is not quite a return to the hub-and-spoke topology of time-sharing systems, if only because there is no hub. A client computer on the Internet can communicate with many servers at the same time, some of which may also be exchanging information among themselves. However, even if we are not returning to the architecture of time-sharing systems, the sudden stylishness of the cloud paradigm marks the reversal of a long-standing trend. Where end users and corporate IT managers once squabbled over possession of computing resources, both sides are now willing to surrender a large measure of control to third-party service providers. What brought about this change in attitude?

For the individual, total control comes at a price. Software must be installed and configured, then updated with each new release. The computational infrastructure of operating systems and low-level utilities must be maintained. Every update to the operating system sets off a cascade of subsequent revisions to other programs. Outsourcing computation to an Internet service eliminates nearly all these concerns. Cloud computing also offers end users advantages in terms of mobility and collaboration.

For software vendors who have shifted their operations into the cloud, the incentives are similar to those motivating end users. Software sold or licensed as a product to be installed on the user's hardware must be able to cope with a baffling variety of operating environments. In contrast, software offered as an Internet-based service can be developed, tested, and run on a comput-

For most applications, the entire user interface resides inside a single window in a Web browser.

ing platform of the vendor's choosing. Updates and bug fixes are deployed in minutes. (But the challenges of diversity don't entirely disappear; the server-side software must be able to interact with a variety of clients.)

Although the new model of Internet computing has neither hub nor spokes, it still has a core and a fringe. The aim is to concentrate computation and storage in the core, where high-performance machines are linked by high-bandwidth connections, and all of these resources are carefully managed. At the fringe are the end users making the requests that initiate computations and who receive the results.

Although the future of cloud computing is less than clear, a few examples of present practice suggest likely directions:

Wordstar for the Web. The kinds of productivity applications that first attracted people to personal computers 30 years ago are now appearing as software services. The Google Docs programs are an example, including a word processor, a spreadsheet, and a tool for creating PowerPoint-like presentations. Another undertaking of this kind is Buzzword, a Web-based word processor acquired by Adobe Systems in 2007.

Another recent Adobe product is Photoshop Express, which has turned the well-known image-manipulation program into an online service.

Enterprise computing in the cloud. Software for major business applications (such as customer support, sales, and marketing) has generally been run on corporate servers, but several companies now provide it as an on-demand service. The first was Salesforce.com, founded in 1999, offering a suite of online programs for customer relationship management and other business-oriented tasks; the company's slogan is "No software!"

Cloudy infrastructure. It's all very well to outsource the chore of building and maintaining a data center, but someone must still supply that infrastructure. Amazon.com has moved into this niche of the Internet ecosystem. Amazon Web Services offers data storage priced by the gigabyte-month and computing capacity by the CPU-hour. Both kinds of resources expand and contract according to need. IBM has announced plans for the "Blue Cloud" infrastructure. And Google is testing the App Engine, which provides hosting on Google server farms and a software environment centered on the Python programming language and the Bigtable distributed storage system.

The cloud OS. For most cloud-computing applications, the entire user interface resides inside a single window in a Web browser. Several initiatives aim to provide a richer user experience for Internet applications. One approach is to exploit the cloud-computing paradigm to provide all the facilities of an operating system inside a browser. The eyeOS system, for example, reproduces the familiar desktop metaphor—with icons for files, folders,

---

Gartner's Seven IT Grand Challenges

What are the most important IT challenges for the next 25 years? At the recent Gartner Emerging Trends Symposium/ITxpo, Gartner analysts identified seven IT grand challenges that, if met, will have profound economic, scientific and societal impacts. They are:

▶ Eliminate the need to manually recharge wireless devices
▶ Parallel programming applications that fully exploit multicore processors
▶ Non-tactile, natural computing interfaces
▶ Automated computer-to-human speech translation

▶ Reliable, long-term digital storage
▶ Increase programmer productivity by 100 percent
▶ Identify the financial consequences of IT investments

"IT leaders should always be looking ahead for the emerging technologies that will have

a dramatic impact on their business, and information on many of these future innovations are already in some public domain," says Gartner VP Ken McGee. To find such information, Gartner suggests examining relevant research papers, patents, and production prototypes.

and applications—all living in a browser window. Another solution would bypass the Web browser, substituting a more-capable software system that runs as a separate application on the client computer and communicates directly with servers in the cloud. This is the idea behind AIR (formerly Apollo) being tested by Adobe Systems. OpenLaszlo, an open-source project, works in much the same way.

For those deploying software out in the cloud, scalability is a major issue—the need to marshal resources in such a way that a program continues running smoothly even as the number of users grows. It's not just that servers must respond to hundreds or thousands of requests per second; the system must also coordinate information coming from multiple sources, not all of which are under the control of the same organization. The pattern of communication is many-to-many, with each server talking to multiple clients and each client invoking programs on multiple servers.

The other end of the cloud-computing transaction—the browser-based user interface—presents challenges of another kind. The familiar window-and-menu layer of modern operating systems has been fine-tuned over decades to meet user needs and expectations. Duplicating this functionality inside a Web browser is a considerable feat. Moreover, it has to be done in a comparatively impoverished development environment. A programmer creating a desktop application for Windows or one of the Unix variants can choose from a broad array of programming languages, code libraries, and application frameworks; major parts of the user interface can be assembled from pre-built components. The equivalent scaffolding for the Web computing platform is much more primitive.

A major challenge of moving applications to the cloud is the need to master multiple languages and operating environments. In many cloud applications a back-end process relies on a relational database, so part of the code is written in SQL or other query language. On the client side, program logic is likely to be implemented in JavaScript embedded within HTML documents. Standing between the database and the client is a server application that might be written in a scripting language (such as PHP, Java, and Python). Information exchanged between the various layers is likely to be encoded in some variation of XML.

Even though the new model of remote computing seems to reverse the 1980s "liberation" movement that gave individual users custody over programs and data, the shift does not necessarily restore control to managers in the corporate IT department.

To the extent that cloud computing succeeds, it represents an obvious competitive challenge to vendors of shrink-wrap software. Ironically, the open-source movement could also have a tough time adapting to the new computing model. It's one thing to create and distribute an open-source word processor competing with Microsoft Word; not so obvious is how a consortium of volunteers would create a Web service to compete with Google Docs.

Finally, cloud computing raises questions about privacy, security, and reliability—a major subject of discussion at a workshop held last January at the Center for Information Technology Policy at Princeton University. Allowing a third-party service to take custody of personal documents raises awkward questions about control and ownership: If you move to a competing service provider, can you take your data with you? Could you lose access to your documents if you fail to pay your bill? Do you have the power to expunge documents that are no longer wanted?

The issues of privacy and confidentiality are equally perplexing. In one frequently cited scenario, a government agency presents a subpoena or search warrant to the third party that has possession of your data. If you had retained physical custody, you might still have been compelled to surrender the information, but at least you would have been able to decide for yourself whether or not to contest the order. The third-party service is presumably less likely to go to court on your behalf. In some circumstances you might not even be informed that your documents have been released. It seems likely that much of the world's digital information will be living in the clouds long before such questions are resolved. ▣

Brian Hayes writes about science and technology from Durham, NC.

Virtual Reality
A Fly's Life

A team of Swiss and U.S. researchers have developed an interactive virtual-reality display system that enables them to better understand fruit flies' behavior and movement in response to their visual environment, *New Scientist* reports.

Led by Steven Fry of the Institute of Neuroinformatics in Zurich, the Swiss-U.S. team built a wind tunnel in which changing scenes or images are projected onto its walls. A camera tracks a fruit fly in 3D, making the scenes or images move in response to the animal's activity inside the wind tunnel. Previous research had involved tethered flies which, Fry said, "is very unnatural and it becomes very difficult to interpret the data because of the strong interference by the experimenter."

The team's research, which has implications for animal behavior and biomimetic design control, can be readily reproduced, according to Fry. "Being based on standard hardware and software techniques, our methods provide an affordable, easy to replicate, and general solution for a broad range of behavioral applications in freely moving animals," he says.

Information Technology
Wireless Conductor Paths?

The conductor paths in sensor systems have traditionally consisted of thin wires—until now. Researchers at the Fraunhofer Institute for Manufacturing Engineering and Applied Materials Research in Bremen, Germany, have developed a new technique that prints conductor paths, using a contactless aerosol ink with nano-sized silver particles. In tests conducted with the Institute for Microsensors, Actuators and Systems at the University of Bremen, the printed conductor paths have proven to be nearly 500 times thinner than wire bonds, and the sensors provide significantly more accurate measurements.