White Paper

Case Study in Scientific Application Streaming

A Case Study in Scientific Application Streaming at the Harvard School of Engineering and Applied Sciences

Executive Summary

The School of Engineering and Applied Sciences (SEAS) serves as the connector and integrator of Harvard's teaching and research efforts in engineering, applied sciences (including computer science), and technology.

Within SEAS, the office of Computing and Information Technology's (CIT) CyberInfrastructure Labs (CI Labs) supports faculty, researchers, students, and staff by deploying and maintaining up-to-date, effective computing technologies – and serves as a technology adviser and collaborator to the broader Harvard community.

With its collaborative computing focus and emerging technology test beds, the CIT CI Labs is an ideal environment for the implementation of scientific and engineering application streaming.

The goal of the scientific application streaming project, as outlined in this white paper, is to simplify the deployment of engineering and scientific applications to a user population of around 1,000 students and faculty.

Large, complex scientific and engineering applications are being deployed over a heterogeneous network architecture to a highly diverse client population. Initial results show install times decreasing from hours to minutes, as well as fewer problems caused by human error during complex installation and licensing procedures.

As innovators among Harvard's IT community, the CI Labs anticipates wider implementation of application streaming, both within its user base and across Harvard.

Table of Contents

About Harvard University and SEAS

Located in Cambridge, Massachusetts, Harvard University has some 40,000 computer users at 12 different schools, operating under a decentralized information technology model.

SEAS serves as the connector and integrator of Harvard's teaching and research efforts in engineering, applied sciences, and technology. SEAS offers graduate and undergrad programs in Applied Mathematics, Applied Physics, Bioengineering, Electrical Engineering, Environmental Sciences and Engineering, Mechanical Engineering, and Computer Science.

SEAS deploys, administers, and licenses engineering and scientific applications used for class assignments and research, such as MATLAB,* Mathematica,* COMSOL,* and Abacus.*

Although SEAS has approximately 75 faculty and 750 students, the actual number of users supported by the CI Labs is closer to 1,000 because of visiting students and faculty, dual appointments, and students at other Harvard schools and universities in the region who take classes at SEAS.

As SEAS students, faculty, and staff are technically savvy, the school provides a fertile environment in which to test new technologies. More broadly, through such projects, CIT CI Labs will be able to better position itself as a innovative leader in IT and become an increasingly important, and central resource for the entire Harvard University community.

> "Not many schools and universities are able to test new technology. We're fortunate to have an environment that's conducive to experimenting with emerging technologies: a very flexible audience, along with a set of staff engineers who are open-minded and looking for new ways of doing business."
>
> Joy Sircar, Associate Dean for Computing and Information Technology and CIO, Harvard University SEAS

Usage Scenario

Unlike a typical corporate IT group, the CI Labs does not necessarily control the client environment. There are several typical usage scenarios:

- Students and faculty use their own laptop or desktop computers, running various versions of Linux,* Microsoft Windows,* or Mac OS* and connected to the network either via Wi-Fi or a wall jack
- Clusters of Linux* machines owned by SEAS are used on campus
- Remote access from off-campus or Wi-Fi via VPN

Applications are currently installed on an ad hoc basis. Usage doesn't necessarily start at the beginning of term, but rather as research or assignments require it. Students and faculty come looking for applications as they are needed throughout the semester, with the peak need for engineering and scientific application installations several weeks into the term.

It would be inefficient and prohibitively expensive for the CI Labs to provide all required applications to everyone. Instead, they need to provide efficient, cost-effective on-demand access to applications, and to practice active license management and harvesting.

End users find the current installation process confusing. Typical questions focus on where to find the software, how to install it, and who pays for it. Resolving these issues can cause multiple interruptions for students, faculty, and staff.

Once they've figured out where to go, users typically come to the SEAS CIT Service Desk with their laptops and are walked through the installation process by a technician. Alternatively, a service desk technician may make an office visit, especially for faculty.

These are large, computationally intensive engineering and scientific applications with complicated, idiosyncratic licensing and installation procedures. Installation may take several hours and can be problematic even for the service desk to learn. It's easy to get a simple configuration setting wrong, resulting in application errors and repeat visits.

The software is licensed using a network licensing server, with a pool of concurrent licenses shared across the user population. Due to the heterogeneous client environment, there hasn't been a good way to authenticate licenses, and managing shared licenses is an ongoing issue.

Goals

In evaluating scientific and engineering application deployment technologies, the CI Labs had these goals:

Simplify application deployment for users. Above all they wanted to make software installation self-service and easy, reducing the time it takes users to get up and running.

As Joy Sircar observes, "Four years [at Harvard] seems like a long time, but it actually goes by in a flash. We don't want students wasting time at a service desk on stuff like software installation in the short time that they're here."

Lower technology deployment costs and staff time. A small number of staff needs to support a relatively large number of users, without sacrificing accuracy. The goal was to reduce problems and repeat visits caused by installation errors.

Validate new technology for potential use across the Harvard system. The CI Labs saw an opportunity to take advantage of their school's technical orientation and small size to pilot a new technology that may eventually benefit all of Harvard.

Says Robert Heywood, Associate Director of CIT Platforms and Services: "We're looked upon to take more risks. We have both the capability and the inclination for innovation, and we're small so we can be more nimble. We try to capitalize on that. The community expects it of us."

About the CI Labs and the Solution

Project history

In February 2008, a new group called the CyberInfrastructure Labs (CI Labs) formed within SEAS CIT. Part of its mission: to test new computing technologies and deploy them in a way that's cost-effective and easy for users.

The CI Labs began searching for ways to make application deployment more nimble and efficient. Approaches considered included low-level web services, high-level portal technology, and distributed file systems, as well as thin client, remote desktops, server-based and cloud computing.

Server-based desktop virtualization was one option that was considered in detail. With this model, the desktop runs as a virtual machine on a server. However, this approach involves a costly up-front investment in server-side computing capacity.

The CI Labs ultimately decided on application streaming, due in part to its light server footprint. Server resources are needed only when downloading, instead of all the time, as with virtual desktop infrastructure (VDI). Plus, the startup cost is small—a plus in tight economic times—and it's easily scalable.

With application streaming, applications are streamed on demand from the data center to the client, where they are executed locally.

Although the terms "streaming" and "application virtualization" are sometimes used interchangeably, streaming refers to the delivery model of sending the software over the network for execution on the client. Streamed software can be installed in the client OS locally or it can be virtualized.

Virtualized software runs on an abstraction layer and does not install in the OS registry or system files. This simplifies the interactions between the streamed application and the OS, greatly reducing software conflicts and image management problems.

System architecture

The CI Labs piloted Altiris Software Virtualization Solution Professional (SVS Pro)* as an application streaming solution.

The application streaming portal and software packages were located on a single server. However, the application server could be split among multiple machines for scalability if needed (if they had 10,000 users, for example). Application packaging was done on a separate workstation.

Additionally, a separate development environment was also used to create packages, deploy them on a test server, and access them using a range of test clients, before running them on the production server.

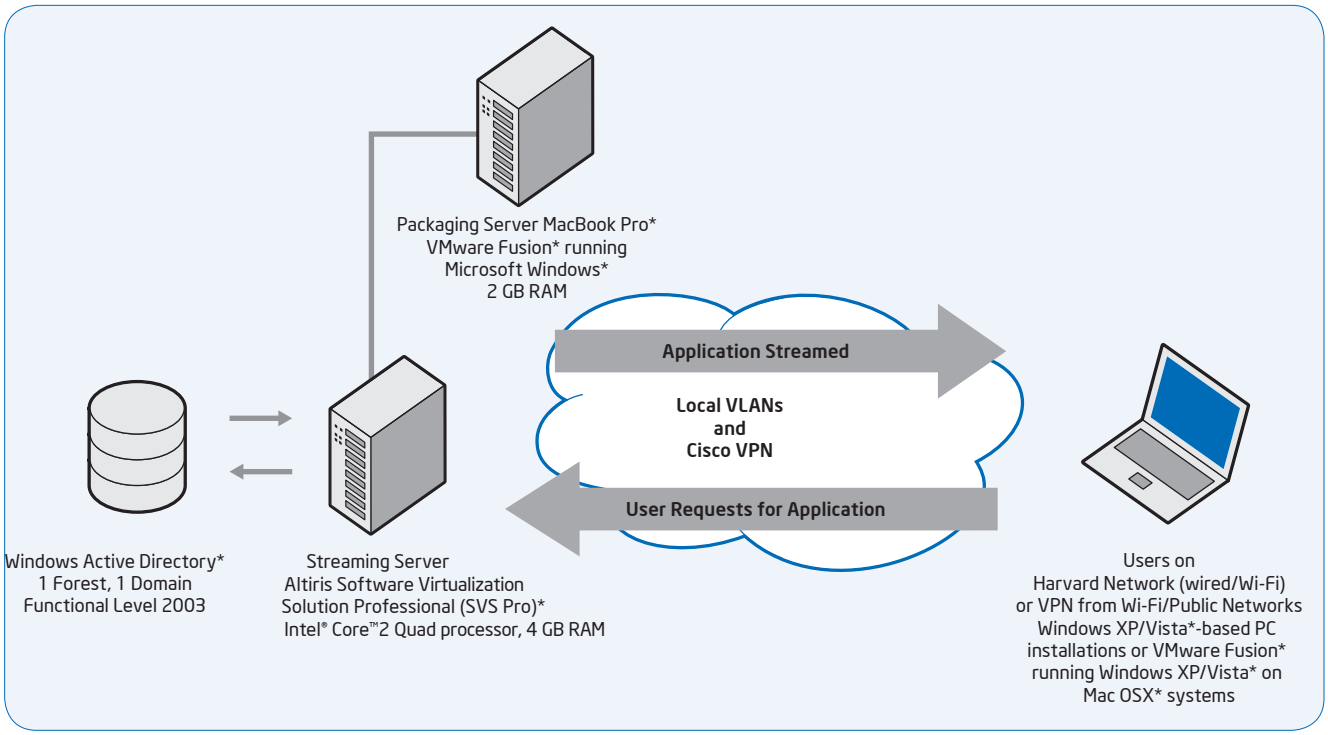

Figure 1. Architecture diagram for Harvard SEAS CI Labs scientific and engineering application streaming pilot.

User experience

To access scientific and engineering applications, users simply followed a web link, which took them to a login page, if necessary. They were then presented with a menu of applications.

The streaming agent is the client software that communicates with the streaming server. The first time users connect, the agent is downloaded and installed automatically. From then on, every time users click on an application, streaming starts automatically.

Within a few minutes, users can interact with the application. The program isn't actually all there yet, but as they use it, more of the application is streamed down. This approach works well for large, complex scientific applications, whose extensive functionality is required by few users. Alternatively, users can cache the entire application for offline use.

Pilot testing

The initial deployment of scientific and engineering application streaming involved packaging four applications. The CI Labs started by picking its most complex applications: MATLAB,* Mathematica,* COMSOL,* and Abacus.*

At time of writing, 39 users had installed and streamed applications successfully. Despite the small initial user population, the CI Labs felt that application streaming is being validated due to the complexity of the applications and computing environment.

Concerns

The CI Labs had a few concerns going into the pilot:

Application performance. There were concerns about how long it would take to install and run these large complex applications, especially over wireless and VPN connections.

Network access. Harvard has a big, sprawling network space, of which SEAS is a customer rather than an owner. As traffic travels around the network, it may encounter security boundaries at any hop, causing unexpected application errors.

> "We can now deploy applications in a very simple way. We do all the complicated stuff once, and then it's easy, instead of having to train the service desk to perform several different installation procedures. Capturing each installation procedure and deploying the result to users in a uniform way is a great administrative advantage.
>
> Jack Yatteau, Technical Lead

Results

Concerns about the impact of streaming technology on the user experience were addressed in part by positive initial timing results of first application use (installation) and subsequent use. For example, the time elapsed from installation start to the point at which the user could first interact with the COMSOL* application was reduced from 25 minutes (via DVD install) to two minutes (via clicking the application package from the application streaming server menu) over a wireless connection. From that point on, application startup times using the streamed version and the DVD-installed version were similar, as expected.

As for network access, testing from known network points enabled them to exercise the most difficult access cases first.

According to Robert Heywood, "So far there haven't been any serious issues—and they usually show up early."

Benefits

The CI Labs saw these early benefits of application streaming:

Quicker startup for users. Users can start using a new application in minutes rather than hours, with reduced confusion about where to source the software and fewer service desk visits. Plus, the installation process itself is quicker, as is application startup.

Installations are more consistent. With manual installation, the outcome depends on the time, place, and knowledge of the installer. But with streaming, the process is standardized.

Safer deployment of new application versions. Scholars are in the middle of experiments and labs may not want to install a new application version, since a change in the environment would require verifying results to make sure the software itself isn't skewing the output. With application virtualization, both versions can be streamed without interference and outputs can be easily compared.

Streaming Windows* applications to Mac OSX*. Scientific and engineering applications are now more accessible to Mac*-based workstations and laptops. Successful streaming of Windows*-based applications to Mac OSX* systems running VMware Fusion* offers a unique hybrid solution, allowing early accessibility to streaming services in a heterogeneous computing environment.

Better software usage tracking. Because user identity is validated before streaming, the CI Labs now has better data about application installation and usage.

CI Labs lessons learned

The CI Labs has these tips for IT management considering application streaming:

Learn the basics. Streaming technology is evolving and improving rapidly. In order to get the most out of your project, you need enthusiastic, capable staff that can come up to speed fast—plus good relationships with your user communities and vendors.

Know your network. Especially in a complex environment like Harvard's, detailed knowledge of the environment is essential. If that knowledge is not available, you'll need to build time into the schedule to acquire it.

Have enough capacity for packaging
Make sure to dedicate enough disk space and memory on the packaging server.

For example, to package a 2 GB application, they started out with 4 GB but kept running out of disk space, and ended up needing 12 GB of disk space and 512 MB of RAM.

> "Overall, it worked better than we expected. I was surprised how easily they could create the packages. That was a major unknown for these big scientific applications."
>
> Robert Heywood, Associate Director of CIT Platforms and Services

Ongoing issues

The CI Labs also discovered some ongoing issues with streaming technology:

Support for Macs. The current technology doesn't run natively under Mac OSX,* which are increasingly common in higher education computing environments.

They look forward to being able to deliver applications in the same way to all their users, which will open the door to replacing their licensing server with Active Directory*-based authentication.

Handling networked licenses. Until platform issues are resolved and all users are receiving applications via streaming, the CI Labs must use their existing licensing solutions.

Said David Gipstein, Director of SEAS Computing & Information Technology: "We did encounter some issues with licensing. This technology may present challenges when managing tightly bound network licenses, unless you have complete control of your user environment. However, in a more homogeneous environment, such as you might find in a corporation, licensing might be easily managed by integrated streaming components."

Future plans

The CI Labs has identified some directions for future pilots in their deployment of scientific and engineering application streaming:

Using Intel® vPro® technology advanced management capability, including improved license management capabilities and the ability to push out patch upgrades and roll them back.

Explore server virtualization. The CI Labs wants to experiment with hosting the application and packaging servers on VMware ESX*, which already hosts their primary server virtualization infrastructure, along with VMotion* technology to balance and move virtual machines between physical hosts on the cluster.

Conclusion

Encouraged by initial results, the CI Labs plans to pilot other applications among its users, including other licensed applications as well as public domain software that doesn't require licensing. As a test bed for new technologies with a flexible audience, the CI Labs is well positioned to experiment and pilot these new technologies, such as scientific and engineering application streaming, and to share lessons learned across Harvard.

Says David Gipstein: "Streaming is a great collaborative technology. It allows flexibility for remote access, but it's also secure. If a group has specialized or standard applications that they wish to deploy, they can be provisioned and hosted as an application service in either a managed or delegated model that is both flexible and scalable. The desire is to enable and allow faculty, students, and staff to focus on learning, the educational environment, and the academic mission of the university."

Printed in USA 0109/WW/OCG/XX/PDF Please Recycle 321190-001US