Mozilla: Behind the Open Source Giant

Chelsea Carr

Mozilla began in 1998 as a side project of Netscape. It quickly became the largest open-source company where developers from all over the world contributed code for browsers, development tools and many other projects. Today, Mozilla is still one of the largest open-source projects around and wouldn't even exist if it weren't for the thousands of developers that commit their time to Mozilla's projects.

Servers

Mozilla manages millions of user's data on a regular basis and they have to use quite a storage system to do so. All of their servers are Linux based. They also use storage appliances such as NetApp and Equalogic, which aside from managing all of their data from day to day, also allows them to use virtual servers wherever applicable.

In-House Software

A lot of Mozilla's software was developed in-house. The main pieces of software they use are:

- **Gecko**
  This is the main layout engine that reads web content, such as XUL, HTML, CSS and JavaScript.
- **Necko**
  This is the main network library, which is comprised of APIs that help maintain the different network layers.
- **NSPR**
  Netscape Portable Runtime is an API used to maintain system-level functions.
- **NSS**
  Network Security Services is a library that supports cross-platform network security.
- **Rhino**
  An open-source implementation of JavaScript, written completely in Java.
- **SpiderMonkey**
  This is the JavaScript Engine used to power Gecko. It is written in C.
- **Tamarin**
  This is a JavaScript Engine written in C++.
- **XPCOM**
  This is a cross platform component object model.
- **XULRUNNER**

This is an environment that allows developers to build and write XUL applications.

One of the main, non-Mozilla created pieces of software is their source control, Mercurial. This system helps manage code changes and repository manipulation.

Open-Source Within Open-Source

Mozilla is an open-source company. Every piece of software they use is considered open-source.

Deployment

To prevent any bugs or broken code from going through, Mozilla has quite an extensive code review process. All added code must be submitted through a bug system in the form of a "patch". Patches are essentially text files with all the differences between the programmer's additions and the base revision. Mozilla engineers then review these patches to make sure they work how they are suppose to. If the patches get the go ahead, they are integrated into a test build that then must pass all of Mozilla's tests, unit tests and performance tests are examples, before it can than be moved to the production build to be released. New versions of existing browsers and other products are deployed every couple of months.

Security

As mentioned above, Mozilla writes a lot of their own network security. They also keep a running list of known security bugs that take priority over normal bugs. NSS helps to support SSL and other security standards. Being an open-source company does provide the risk of people committing malicious code, however their strict code check-in process makes it nearly impossible to let anything bad through.

Sources:

http://www.mozilla.org/en-US/firefox/technology/
http://www.mozilla.org/about/history.html
https://careers.mozilla.org/en-US/position/o97xWfwt
http://www.mozilla.org/projects/technologies.html