

Shared Leadership IN THE Apache Project

Roy T. Fielding

The Apache Project [1] is a collaborative software development effort aimed at creating and maintaining a robust, secure, efficient, extensible, and open-source implementation of an HTTP (Web) server. The project is managed by the Apache Group, a geographically distributed set of volunteers who use the Internet and Web to communicate, develop, and distribute the server and its related

documentation. In addition, hundreds of users have contributed ideas, code, and documentation to the project. According to the Netcraft survey of November 1998 [4], the Apache server and its derivatives are used by over 57% of publicly accessible Web sites, more than double its nearest competitor.

Unlike most open-source projects, Apache has not been organized around a single person or primary contributor. When the project began in February 1995, the most popular server software on the Web was the public domain HTTP daemon developed by Rob McCool at the National Center for Supercomputing Applications (NCSA). However, development of NCSA httpd had stalled after Rob McCool left, and many Web masters had developed their own extensions and bug fixes that were in need of a common distribution. A small group of these Web masters gathered together via private email for the purpose of coordinating their changes (in the form of “patches”). Brian Behlendorf volunteered the use of his server as a shared information space, providing logins for the eight core developers and hosting a public mailing list for communication. That left two problems to be solved before the development could proceed: decision making and coordination.

Apache has always been a multinational

project, with core developers located in the U.S., Britain, Canada, Germany, and Italy. Collaboration within the group is hindered by variation in work schedules and the effect of cross-Atlantic network latency. Each Apache Group volunteer has (at least) one other “real” job, usually related to either Web services or protocol research. We collaborate on producing and supporting the Apache server out of enlightened self-interest: by pooling our efforts, the resulting product is much more functional and robust than anything we could have produced alone. However, this also means that none of us can devote large blocks of time to the project.

In order for the project to succeed, our development process (the procedures by which we make decisions and coordinate our efforts) had to reflect this globally distributed and volunteer organizational environment. There was no Apache CEO, president, or manager to turn to for making decisions. Instead, we needed to determine group consensus, without using synchronous communication, and in a way that would interfere as little as possible with the project progress.

What we devised was a system of voting via email that was based on minimal quorum consensus. Each independent developer could vote on any issue facing the project by send-

ing mail to the mailing list with a "+1" (yes) or "-1" (no) vote. For code changes, three positive votes and no negative votes (vetoes) were required before the change would be allowed in the final source. For other decisions, a minimum of three "+1" and an overall positive majority was required. Anyone on the mailing list could vote, thus expressing their opinion on what the group should do, but only those votes cast by the Apache Group members were considered binding.

The voting system had a number of interesting properties. By setting the minimum at three positive votes, only a subset of the group had to be involved in any decision. This allowed participation in the project to be bursty, focused into periods of two or three days a week by each developer, without blocking overall progress on any given day. At the same time, the minimum vote enforced a high degree of peer review over all code changes. Veto power was used sparingly and only when backed by a good explanation, primarily as a means of preventing software bloat (the addition of too many features to the core code) and ensuring that no change would knowingly break a significant supported platform. However, it also had its drawbacks. During periods of rapid and focused development, voting would become a barrier and a frequent source of friction among the developers. What should have been a fun project would sometimes become bogged down in the details of decision by committee.

We have since improved our development process as new tools became available and as the context of the project and membership of the Apache Group changed over time. In 1996, we switched to using CVS [2] to manage our shared information space, including product distributions and Web site content. CVS allows a central repository to maintain consistency and simplify the task of committing changes from the remote workspaces of each developer. Changes to the repository are summarized as a set of differences and sent to a mailing list, thus notifying the other developers of the content being changed. As a result, we now review changes after they are committed to the source, thus streamlining the approval of simple fixes. Anonymous access to the current source code base is provided to the public, enabling a wider testing audience and satisfying those who like the cutting edge.

The Apache project is a meritocracy—the

more work you have done, the more you are allowed to do. New members of the Apache Group are added when a frequent contributor is nominated by one member and unanimously approved by the voting members. A flurry of interest was generated when the IBM Corporation joined the Apache project [3] and began contributing code fixes and features back to the open-source base. IBM is treated the same as any other member of the Apache Group, albeit one with deeper pockets and more developer resources to spare. They have been of considerable help to the group in supporting our first in-person Apache Group meetings and providing access to corporate lawyers. IBM deserves credit for joining the project on Apache's terms, based upon their merit as a contributor, and in continuing to promote both the spirit and the actions of open-source development.

Although the Apache Group makes decisions as a whole, all of the actual work of the project is done by individuals. The group does not write code, design solutions, document products, or provide support to our customers; individual people do that. The group provides an environment for collaboration and an excellent trial-by-fire for ideas and code, but the creative energy needed to solve a particular problem, redesign a piece of the system, or fix a given bug is almost always contributed by individual volunteers working on their own, for their own purposes, and not at the behest of the group. Competitors mistakenly assume Apache will be unable to take on new or unusual tasks because of the perception that we act as a group rather than follow a single leader. What they fail to see is that, by remaining open to new contributors, the group has an unlimited supply of innovative ideas, and it is the individuals who chose to pursue their own ideas who are the real driving force for innovation. ■

REFERENCES

1. Apache HTTP Server Project; www.apache.org
2. Concurrent Versions System (CVS); www.cyclic.com/cvs/info.html
3. IBM chooses "open-source" Apache server. *IEEE Internet Computing* 2, 4 (Jul.–Aug. 1998).
4. Netcraft Web Server Survey, November 1998; www.netcraft.com/Survey/

ROY T. FIELDING (fielding@ics.uci.edu) is a doctoral candidate in Information and Computer Science at the University of California, Irvine, and cofounder of the Apache Project.