



CHALLENGE

Adopt CI across a multinational development organization to increase the pace of development for Android software

SOLUTION

Use CloudBees Jenkins Enterprise™ to establish role-based access control, streamline backups, gain access to expert support and maximize the benefits of Jenkins in an enterprise environment

RESULTS

- » Build times cut by more than 85%
- » Days of backup maintenance time eliminated
- » Cross-enterprise collaboration improved

PRODUCTS

- » CloudBees Jenkins Platform™ - Enterprise Edition

Leading Semiconductor Manufacturer Accelerates Mobile Application Delivery and Improves Developer Collaboration in a Distributed Enterprise Environment

Teams developing software for Android and other mobile operating systems are expected to deliver much faster and more frequently than their counterparts working on traditional online or desktop applications. Maintaining this accelerated development pace can be difficult, particularly for an enterprise with large, geographically distributed teams. "The world of Android moves really fast and it's moving faster every year," says Jeremy Capps, software services and infrastructure design lead at a multinational semiconductor manufacturer. "That can be a problem for us because we're so large, and our size can slow us down."

Developers at the semiconductor manufacturer began using Continuous Integration (CI) with Jenkins to keep up with the speed of Android development, and the company now relies on CloudBees Jenkins Enterprise™ to maximize the advantages of Jenkins CI. "Groups that were doing two drops a year are now doing about 20 drops a year with Jenkins and the overall quality of the software has improved," says Capps. "CloudBees Jenkins Enterprise lets our teams securely share their work with other teams, reduces maintenance overhead and gives us access to expert support when we need it."

"Instead of discovering an issue the day before release, with Jenkins CI we discover it when it first gets into the code. CloudBees Jenkins Enterprise enables our large, distributed team to capitalize on the advantages of CI—all developers are involved in the process, bottlenecks are removed, and the quality of software shoots through the roof."

Jeremy Capps

Software Services and Infrastructure Design Lead
at a Multinational Semiconductor manufacturer

CHALLENGE: MAXIMIZE THE BENEFITS OF JENKINS IN A DISTRIBUTED ENTERPRISE ENVIRONMENT

Developers at the company needed a way to streamline multi-configuration Android builds. Recognizing the advantages of CI for this work, individual developers initially set up their own Jenkins instances. Although this ad hoc approach can work with small groups, it had several significant drawbacks in this large organization. First, it lacked consistency across teams. Builds that worked for one team on one version of Jenkins broke when another team tried to build it in production for release. Second, backups were not being performed reliably, and the maintenance overhead for backup activities was increasing, with developers writing and maintaining their own backup scripts. Third, developers had no dependable way to rapidly resolve problems that cropped up. “In some cases, I’d spend an hour or more looking through open source forums to find an answer to a problem we were having, only to find that the person who provided that answer did not know what they were talking about,” says Capps.

Aside from these difficulties, the biggest challenges the company faced with open source Jenkins were centered on access control. “It was common for one group to deny access to their builds to other groups, because to grant access to one small portion meant they had to grant full access to everything. The choice was all or nothing; there was no fine-grained access control,” says Capps. “Our team grew in one year from about 60 engineers in two locations to more than 450 in three locations, and with a team that size effective access control is essential.”

SOLUTION: A SOLUTION FOR CI ACROSS THE ENTERPRISE

The semiconductor manufacturer subscribed to CloudBees Jenkins Enterprise to support CI across its geographically distributed development organization.

The team started by deploying the Jenkins Long Term Support (LTS) release with CloudBees Jenkins Enterprise plugins. They set up jobs for running multiple parallel builds for a matrix of different Android versions, platforms and configurations. However, when a fix was needed for a command-line issue that was available in the open source software (OSS) version of Jenkins before it was available in the LTS release, the team switched to OSS Jenkins. “I was delighted to learn that I could simply switch to OSS Jenkins, update frequently and still use the CloudBees® plugins that we rely on,” says Capps.

Capps and his colleagues deployed several CloudBees Jenkins Enterprise plugins, starting with the Role-based Access Control

plugin. They established roles, including project administrator, developer and reviewer, as well as a daemon role for automated background processing. User and group information stored in Active Directory are accessed via the LDAP plugin.

Used together with the Folders plugin, the Role-based Access Control plugin enabled the team to provide the fine-grained access control they needed—where authorized users were given access to individual folders, instead of granting them access to an entire file server.

The team replaced Python scripts and other informal backup mechanisms with the Backup Scheduling plugin available with CloudBees Jenkins Enterprise. They created a backup job in Jenkins and established a uniform system for performing offsite backups across all instances.

Now, to resolve situations that previously may have required hours of searching on forums and tracking issues, the team contacts CloudBees support. “CloudBees engineers have been very responsive. Even if I raise a question that isn’t necessarily a bug, they investigate it right away,” says Capps.

The company plans to begin using the High Availability plugin to eliminate downtime, even when upgrading Jenkins and plugins

RESULTS

Build times cut by more than 85%.

“A year ago we it would take four hours to complete builds for the four or five platforms we were working with,” says Capps. “Today, we have 15 platforms but with Jenkins, we’ve cut our build times to 30 minutes. The ability to do multiple configuration builds is essential, and we couldn’t have achieved those results with other CI products.”

Days of backup maintenance time eliminated.

“Over a year, the Backup Scheduling plugin saves me about a week of work,” says Capps. “It has eliminated the hassles we previously faced with backup administration caused by different scripts and policies that were used by different groups.”

Cross-enterprise collaboration improved.

“With the Role-Based Access Control plugin, teams can share what needs to be shared while still maintaining control of their builds,” Capps notes. “That has cut down on the email between teams in Bangalore and Oregon, and the time spent communicating about jobs and builds has been cut in half. On a weekly basis we’ve saved five to seven hours of work per developer.

