CloudBees®

The Enterprise Jenkins Company

CloudBees Jenkins Platform Features

Introduction

The CloudBees Jenkins Platform enables Jenkins at enterprise scale, extending the same open source Jenkins you already know and love, with key CloudBees-developed enterprise features required by teams delivering mission-critical software applications. In addition, the CloudBees Jenkins Platform includes professional technical support by a team of Jenkins experts led by Kohsuke Kawaguchi, the creator of Jenkins.

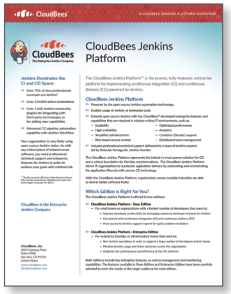
The CloudBees Jenkins Platform is made up two components:

- **CloudBees Jenkins Enterprise:** An enterprise-grade edition of Jenkins that includes advanced features to optimize usage of Jenkins.
- **CloudBees Jenkins Operations Center:** An operational dashboard for your entire Jenkins environment that provides advanced features for monitoring and managing Jenkins, as well as an analytics dashboard.

The enterprise features delivered with the CloudBees Jenkins Platform are provided in the form of CloudBees-developed plugins. The plugins and the features they provide can be grouped into two areas of scale:

- Those that enable developers and small teams to better utilize Jenkins
- Those that enable large organizations to run Jenkins at scale

Explore the CloudBees Jenkins Platform via this e-book. Once you are done, you can:



CloudBees CLOUDBEES JENKINS PLATFORM OVERVIEW

CloudBees The Enterprise Jenkins Company

CloudBees Jenkins Platform

Jenkins Dominates the CI and CD Space:

- Over 70% of Java professionals surveyed use Jenkins*
- Over 120,000 active installations
- Over 1,000 Jenkins community plugins for integrating with third-party technologies or for adding new capabilities
- Advanced CD pipeline automation capability with Jenkins Workflow

Your organization is very likely using open source Jenkins today. As with any critical piece of infrastructure software, you need professional technical support and enterprise features for Jenkins in order to achieve your goals with minimal risk.

* ZeroTurnaround's 2014 Java Tools Research Report: http://zeroturnaround.com/rebellabs/java-tools-and-technologies-landscape-for-2014/

CloudBees is the Enterprise Jenkins Company

CloudBees, Inc.
2001 Gateway Place
Suite 670W
San Jose, CA 95110
United States
www.cloudbees.com
info@cloudbees.com

The CloudBees Jenkins Platform™ is the proven, fully-featured, enterprise platform for implementing continuous integration (CI) and continuous delivery (CD) powered by Jenkins.

CloudBees Jenkins Platform

- Powered by the open source Jenkins automation technology.
- Enables usage of Jenkins at enterprise scale.
- Extends open source Jenkins with key CloudBees®-developed enterprise features and capabilities that are required in mission-critical IT environments, such as:
  - Scalability
  - High availability
  - Simplified administration
  - Role-based access control
  - Optimized performance
  - Analytics
  - Container (Docker) support
  - Distributed team management
- Includes professional technical support delivered by a team of Jenkins experts led by Kohsuke Kawaguchi, Jenkins founder.

The CloudBees Jenkins Platform represents the industry's most proven solution for CD and a critical foundation for DevOps transformations. The CloudBees Jenkins Platform allows IT organizations to accelerate application delivery by automating and orchestrating the application lifecycle with proven CD technology.

With the CloudBees Jenkins Platform, organizations across multiple industries are able to deliver better software faster.

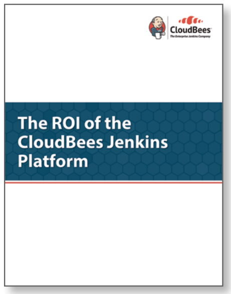
Which Edition is Right for You?

The CloudBees Jenkins Platform is offered in two editions:

- **CloudBees Jenkins Platform - Team Edition**
  - For small teams or organizations with a limited number of developers that want to:
    - Improve developer productivity by leveraging advanced developer features for Jenkins
    - Get started with continuous integration (CI) and continuous delivery (CD)
    - Have access to Jenkins support experts for quick problem resolution
- **CloudBees Jenkins Platform - Enterprise Edition**
  - For enterprise DevOps or infrastructure teams that want to:
    - Run Jenkins operations at scale to support a large number of developers and/or teams
    - Monitor Jenkins usage and share resources across the organization
    - Optimize the performance and efficiency of the CD platform

Both editions include key enterprise features, as well as management and monitoring capabilities. The features available in Team Edition and Enterprise Edition have been carefully selected to meet the needs of the target audience for each edition.

Download the datasheet



CloudBees The Enterprise Jenkins Company

The ROI of the CloudBees Jenkins Platform

Read the Return on Investment whitepaper

>> Jenkins for Teams

- Team Management
- Security
- Developer Productivity
- Continuous Delivery
- CloudBees Support

>> Jenkins at Scale

Jenkins for Teams

The CloudBees Jenkins Platform makes it easier for small teams or organizations to enhance their efficiency and productivity, leveraging even more value from Jenkins. Five groups of plugins are particularly relevant for these teams.

- Team Management
- Security
- Developer Productivity
- Continuous Delivery
- CloudBees Support

Team Management Plugins

Organize your Jenkins jobs by team with the Folders plugin. >>

Recover more quickly from outages by automatically backing up your Jenkins environment with the Backup Scheduling plugin. >>

Document jobs more easily and safely with the WikiText Descriptions plugin. >>

>> Jenkins for Teams
- Team Management
  - Folders Plugin
  - Backup Scheduling Plugin
  - WikiText Descriptions Plugin
- Security
- Developer Productivity
- Continuous Delivery
- CloudBees Support

>> Jenkins at Scale

<< Return to Introduction

Folders Plugin

Folders help to create hierarchies or custom taxonomies to better manage large numbers of jobs.

Challenge

As the number of projects and teams grow, so does the corresponding number of jobs running on Jenkins. Users find that they need to organize their projects and jobs to make management of them easier. Jenkins does not provide built-in functionality that addresses this need.

Solution

The Folders plugin allows you to organize jobs into hierarchical folders, much like how files are organized within directories on your computer. This allows you to group related jobs together. For example, you could set up specific folders by department, projects or jobs.

Folders can also define security permissions (with the Role-based Access Control plugin) on a per folder basis. Roles can be inherited (or filtered out) within the nested folder hierarchy. For example: the engineering department folder can give broad permissions to all engineers, but the project Foo sub-folder only allows certain engineers to modify it.

Folders then allow you to rapidly clone a folder with its children intact. So you can create, for example, a template folder that can be copied and allocated to a new group coming onboard, yet it retains the same folder hierarchy.

Finally, folders are namespace-aware. Therefore, you can have jobs with the same name that sit under different hierarchies.

More details are available on the CloudBees Jenkins Platform documentation website.

>> Jenkins for Teams
– Team Management
– Folders Plugin
– Backup Scheduling Plugin
– WikiText Descriptions Plugin
– Security
– Developer Productivity
– Continuous Delivery
– CloudBees Support
>> Jenkins at Scale

<< Return to Introduction

Backup Scheduling Plugin

Use Jenkins to backup Jenkins. No more cron jobs or error-prone custom scripts.

Challenge

Running backups is crucial. Usually users write scripts or cron jobs to perform backups for Jenkins. Maintaining the backup scripts is cumbersome and error-prone.

Solution

The Backup Scheduling plugin, available with the CloudBees Jenkins Platform, greatly simplifies performing backups. You simply create a new type of job called Backup Jenkins. You can choose to backup job configurations, build records, system configurations or any combination thereof. In addition, since a backup job is a Jenkins job, you can easily relocate the configuration to a different system if required — no porting of shell scripts!

More details are available on the CloudBees Jenkins Platform documentation website.

‹‹ Return to Introduction

WikiText Descriptions Plugin

Prevent potential XSS attacks due to HTML descriptions in Jenkins.

Challenge

Jenkins allows users to enter description fields in HTML for projects. There is a potential for cross-site scripting attacks when using plain HTML. To alleviate these concerns, teams have policy preferences towards writing wiki text, rather than HTML.

Solution

The CloudBees Jenkins Platform WikiText Descriptions plugin allows you to write the description in wiki markup syntax. This addresses concerns about cross-site scripting attacks. The plugin supports multiple wiki markup languages, fitting in with the current wiki solutions typically found in enterprises.

The WikiText Descriptions plugin currently supports the following wiki markup languages:

- Confluence
- MediaWiki
- Twiki
- Textile
- TracWiki

More details are available on the CloudBees Jenkins Platform documentation website.

>> Jenkins for Teams
– Team Management
– Folders Plugin
– Backup Scheduling Plugin
– WikiText Descriptions Plugin
– Security
– Developer Productivity
– Continuous Delivery
– CloudBees Support
>> Jenkins at Scale

<< Return to Introduction

Security Plugins

Secure your jobs with fine-grained permissions using the Role-based Access Control plugin. >>

Isolate team-sensitive agents to ensure performance and compliance with the Folders Plus plugin. >>

>> Jenkins for Teams

- Team Management
- Security
  - Role-based Access Control (RBAC) Plugin
  - Folders Plus Plugin
- Developer Productivity
- Continuous Delivery
- CloudBees Support

>> Jenkins at Scale

Role-based Access Control (RBAC) Plugin

Set-up sophisticated authorization policies to manage Jenkins.

Challenge

You need to have a sophisticated authorization strategy to control access to Jenkins jobs. At the same time, you desire flexibility in setting permissions that go beyond just the job level. For example, you may want to set permissions at a project or department level. You may also want to define secret projects that are only visible to the people permitted to view them.

There are multiple authorization plugins available today but each comes with some limitations. The Role-based Access Control (RBAC) plugin, provided with the CloudBees Jenkins Platform, provides a very sophisticated authorization strategy that exceeds the functionality of other available options.

Solution

The Role-based Access Control plugin gives a Jenkins administrator the ability to define various security roles for the system they administer. Once roles have been defined, the Jenkins administrator can assign those roles to groups of users. The assignment of roles can take place either at the global level or can be limited to specific objects within the system. Additionally, the Jenkins administrator can even delegate the management of groups of specific objects to specific users.

continued >>

Role-based Access Control (RBAC) Plugin (cont.)

The Role-based Access Control plugin combines with the Folders plugin to provide a powerful solution for managing a Jenkins installation shared by multiple teams of users. The Jenkins administrator can create folders for each of the teams and then create groups in those folders for each of the roles that team members are assigned. By delegating the management of the group membership (but not the management of the roles assigned to groups) to the team leaders, the Jenkins administrator can empower the team leads to manage the permissions of their team, while reducing the administrative overhead.

You can find more information on all security-related plugins and how they compare to each other in the Role-based Access Control webinar. The Role-based Access Control plugin allows the administrator to create a role, which is a set of permissions, then leave it up to team leads and other authorized people to control who gets those roles on any given project. This separation makes it easier for teams to set the right access control, without the tediousness of clicking a large number of checkboxes.

Aside from picking up group information from external systems, such as Active Directory, the CloudBees Jenkins Platform lets you define groups locally, even at the folder level. This allows each team that shares the same CloudBees Jenkins Platform instance to rapidly add/remove members of the team, without requiring coordination with either a corporate IT group or the Jenkins administrator.

More details are available on the CloudBees Jenkins Platform documentation website.

<< Return to Introduction

Folders Plus Plugin

Dedicate agents to a project/team by restricting agents to a particular folder.

›› Challenge

Restricting jobs to specific agent is impossible with open source Jenkins. For example, an operations team that stores credentials on a agent and hence wants to prevent other teams from using the agent, cannot do so today.

›› Solution

The Folders Plus plugin, available with the CloudBees Jenkins Platform, allows administrators to tie specific agents to folders (and jobs underneath the folder). Jobs in other folders will not be able to access this particular agent.

More details are available on the CloudBees Jenkins Platform documentation website.

›› Jenkins for Teams
– Team Management
– Security
– Role-based Access Control (RBAC) Plugin
– Folders Plus Plugin
– Developer Productivity
– Continuous Delivery
– CloudBees Support
›› Jenkins at Scale

Developer Productivity Plugins

Broaden collaboration and save time by pre-testing pull requests with the Pull Request Builder GitHub plugin. >>

Improve developer efficiency by avoiding bad Git pushes with the Validated Merge plugin. >>

>> Jenkins for Teams

- Team Management
- Security
- Developer Productivity
  - Pull-Request Builder for GitHub Plugin
  - Validated Merge Plugin
- Continuous Delivery
- CloudBees Support

>> Jenkins at Scale

<< Return to Introduction

Pull-Request Builder for GitHub Plugin

Allows you to configure Jenkins to verify pull-requests on a GitHub project and check that the proposed changes validate continuous integration criteria set on your Jenkins instance.

>> Challenge

Swamped by the need for constant manual code review every time a change is proposed? Having difficulty identifying any potentially negative impacts to a project from proposed changes? Development teams using GitHub rely on the pull-request mechanism for code-review before changes are adopted and merged into the mainstream development branch. This is a very convenient process, as GitHub provides a nice UI to discuss changes and annotate modified lines in the code.

However, manual code review can't catch all possible mistakes. In fact, it may even be a waste of time when the proposed changes have a negative impact on the project and it was missed in the review process. Jenkins jobs can detect issues that will negatively impact the project - before they do.

>> Solution

Use the Pull-Request Builder for GitHub plugin to take advantage of GitHub APIs. You'll be notified when a pull-request is created or updated. The Pull-Request Builder plugin can trigger a build to automatically check any proposed changes, producing a build status report. This allows the reviewer to ignore an invalid pull-request, while the requester can check the log and fix his/her changes. The result: fewer missed issues and less manual time involved to identify and triage them.

More details are available on the CloudBees Jenkins Platform documentation website.

Validated Merge Plugin

Solve the issue of broken builds due to bad commits. With this plugin, Jenkins only merges code that does not break the build.

Challenge

A bad check-in breaks the code repository and causes downtime. This is a very common problem during development and one that causes a lot of lost productivity.

If a bad commit makes it into the repository, an administrator/developer fixes it by backing out the change. Though a frustrating problem for all development projects, the downtime is exacerbated for large and distributed teams.

Teams get around this problem by setting up artificial processes, like asking a developer to run tests before committing a change. If the test suite is huge, it discourages developers from making frequent commits.

More details are available on the CloudBees Jenkins Platform documentation website.

Solution

The Validated Merge plugin, available with the CloudBees Jenkins Platform, prevents downtime due to bad commits. Jenkins acts as an intermediary Git repository. Developers check-in code to this repository instead of the tip, then Jenkins merges the code with the tip and runs tests. If, and only if, the merge and tests are successful is the code pushed to the tip.

This means that bad commits impact only the developer who is committing them — never the whole team, leading to tremendous productivity boosts.

Additionally, developers can now get rid of artificial testing processes. Removing this repetitive step allows developers to focus on developing new code and encourages them to commit more often, thus improving productivity.

The Validated Merge plugin functionality works with Git repositories.

>> Jenkins for Teams
- Team Management
- Security
- Developer Productivity
  - Pull-Request Builder for GitHub Plugin
  - Validated Merge Plugin
- Continuous Delivery
- CloudBees Support

>> Jenkins at Scale

Continuous Delivery Plugins

Build complex delivery pipelines with the Pipeline plugin. >>

>> Jenkins for Teams

- Team Management
- Security
- Developer Productivity
- Continuous Delivery
  - Pipeline Plugin
- CloudBees Support

>> Jenkins at Scale

Pipeline Plugin

A programmatic solution using the Groovy DSL that allows you to create one job to build your pipelines.

›› Challenge

Software delivery pipelines can't be easily managed and adapted to support complex and non-sequential enterprise pipelines. Teams may even need the ability to manage long-running jobs that can take days to complete.

›› Solution

With the Pipeline plugin, you can support simple or complex pipelines alike, using the Groovy DSL. By providing functionality to both pause and restart jobs, Pipeline allows for recovery from both master and agent failures. The ability to recover from a checkpoint successfully passed prior to the failure can potentially save days of time, if your job is one that takes multiple days to run.

More details are available on the CloudBees Jenkins Platform documentation website.

›› Jenkins for Teams

– Team Management

– Security

– Developer Productivity

– Continuous Delivery

– Pipeline Plugin

– CloudBees Support

›› Jenkins at Scale

CloudBees Support Plugins

Get faster resolution to issues in your Jenkins environment with the CloudBees Support plugin. >>

>> Jenkins for Teams

- Team Management
- Security
- Developer Productivity
- Continuous Delivery
- CloudBees Support
  - CloudBees Support Plugin

>> Jenkins at Scale

CloudBees Support Plugin

Get faster resolution of issues by automatically capturing commonly-requested Jenkins information when support tickets are entered.

›› Challenge

One of the key features of a CloudBees Jenkins Platform subscription is the expert Jenkins support provided by CloudBees.

However, when a support ticket is filed, the CloudBees support staff usually has to ask for additional information about the Jenkins environment to help in debugging the issue. This can lead to some back-and-forth between the support engineer and the customer, before the support engineer has the complete information needed to triage and resolve the issue.

›› Solution

The CloudBees Support plugin creates a bundle of various Jenkins-related information required to diagnose issues. The plugin periodically captures snapshots of information and stores it on the Jenkins instance. It can then be sent to CloudBees when a failure occurs.

More details are available on the CloudBees Jenkins Platform documentation website.

›› Jenkins for Teams

– Team Management

– Security

– Developer Productivity

– Continuous Delivery

– CloudBees Support

– CloudBees Support Plugin

›› Jenkins at Scale

>> Jenkins for Teams

>> Jenkins at Scale

- Master Server and Builder
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds

Jenkins at Scale

In addition to enhancing the productivity of developers and teams, the CloudBees Jenkins Platform provides advanced features that enable the use of Jenkins at enterprise scale. The following groups of plugins deliver enterprise-scale functionality.

>> Master Server and Builder

>> Optimized Utilization

>> Enterprise Management

>> Enterprise Analytics

>> Enterprise Continuous Delivery

>> Enterprise Security

>> VMware vSphere Builds

Master Server and Builder Plugins

Automatically recover from master server failures with the High Availability plugin. >>

Quickly restart aborted builds upon master recovery with the Restart Aborted Builds plugin. >>

Enable jobs to survive master failures with the Long-Running Build plugin. >>

Enable jobs to survive both master and agent failure with the Pipeline plugin. >>

>> Jenkins for Teams

>> Jenkins at Scale

- Master Server and Builder
  - High Availability Plugin
  - Restart Aborted Builds Plugin
  - Long-Running Build Plugin
  - Pipeline Plugin
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds

High Availability Plugin

Unplanned downtime can cost the equivalent of days of lost project time. With the High Availability plugin, automatic detection of failure and recovery ensures development activities continue despite a crash.

Challenge

When a Jenkins master fails due to software or hardware crashes, it results in *extended* downtime for the entire product team. Administrators detect these failures manually or through homegrown scripts. Once a failure is noticed, administrators scramble to get the master up as quickly as possible. This process is often manual and can easily take more than a few hours. On larger projects, the downtime experienced from a failure can be the equivalent of several days of lost project time.

Solution

The High Availability plugin, offered within the CloudBees Jenkins Platform, eliminates downtime due to master failures. Multiple Jenkins masters act as backups waiting for a primary master failure. Once a failure is detected, a backup master automatically boots up and acts as a failover. *The best part? Project work continues.*

With this feature, failure detection and recovery is automatic and requires no intervention from administrators or their home-grown scripts. Additionally, this feature can be used to do rolling upgrades of Jenkins.

The Restart Aborted Builds plugin complements the functionality offered within the High Availability plugin.

›› Jenkins for Teams

›› Jenkins at Scale

- Master Server and Builder
  - High Availability Plugin
  - Restart Aborted Builds Plugin
  - Long-Running Build Plugin
  - Pipeline Plugin
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds

More details are available on the CloudBees Jenkins Platform documentation website.

<< Return to Introduction

Restart Aborted Builds Plugin

View and restart jobs after a Jenkins master crash recovery.

›› Challenge

When a Jenkins master crashes, builds that are running at the time of the crash are lost. In a moderately large environment, it is quite a challenge for the administrator to know which ones to restart.

›› Solution

The Restart Aborted Builds plugin, available with the CloudBees Jenkins Platform, presents a list of jobs that were running when Jenkins crashed and provides an easy way to restart them.

This plugin complements the High Availability plugin. The High Availability plugin automatically recovers from a Jenkins crash, allowing the administrator to quickly restart all of the builds that were running when the crash occurred.

More details are available on the CloudBees Jenkins Platform documentation website.

›› Jenkins for Teams

›› Jenkins at Scale

- Master Server and Builder
  - High Availability Plugin
  - Restart Aborted Builds Plugin
  - Long-Running Build Plugin
  - Pipeline Plugin
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds

Long-Running Build Plugin

The Long-Running Build plugin helps you recover from build crashes without restarting from the beginning.

Challenge

We all know what happens to project builds that normally take hours, or perhaps even days, to run when Jenkins crashes. Or, what happens to builds that are running when Jenkins crashes and it is not restarted in Safe mode (waiting for running builds to complete).

The unacceptable answer is, builds need to be restarted — and there goes your project schedule.

Solution

The Long-Running Build plugin, available with the CloudBees Jenkins Platform, solves this problem. This plugin offers a Long-Running project type. The configuration is almost the same as for a standard free-style project, with one difference: the part of your build that you want to be able to survive a restart should be configured as a (UNIX) shell or (Windows) batch step. Of course, this script could, in turn, also run Maven, Make or other tools.

More details are available on the CloudBees Jenkins Platform documentation website.

Pipeline Plugin

A programmatic solution using the Groovy DSL that allows you to create one job to build pipelines. (The Pipeline plugin is also included as a part of the Continuous Delivery Plugins earlier in this E-book.)

Challenge

Software delivery pipelines can't be easily managed and adapted to support complex and non-sequential enterprise pipelines. Teams may even need the ability to manage long-running jobs that can take days to complete.

Solution

With the Pipeline plugin, you can support simple or complex pipelines alike, using the Groovy DSL. By providing functionality to both pause and restart jobs, Pipeline allows for recovery from both master and agent failures. The ability to recover from a checkpoint successfully passed prior to the failure can potentially save days of time, if your job is one that takes multiple days to run.

More details are available on the CloudBees Jenkins Platform documentation website.

Optimized Utilization Plugins

Utilize lightly-loaded agents with the Even Scheduler plugin. >>

Create faster builds across geographically dispersed agents with the Fast Archiver plugin. >>

Improve your master performance under heavy load with the NIO-SSH Slaves plugin. >>

Throttle builds during heavy loads with the Label Throttle Build plugin. >>

Stop running builds for a specified period of time with the Skip Next Build plugin. >>

>> Jenkins for Teams

>> Jenkins at Scale

- Master Server and Builder
- Optimized Utilization
  - Even Scheduler Plugin
  - Fast Archiver Plugin
  - NIO SSH Slaves Plugin
  - Label Throttle Build Plugin
  - Skip Next Build Plugin
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds

» Jenkins for Teams
» Jenkins at Scale
– Master Server and Builder
– Optimized Utilization
– Even Scheduler Plugin
– Fast Archiver Plugin
– NIO SSH Slaves Plugin
– Label Throttle Build Plugin
– Skip Next Build Plugin
– Enterprise Management
– Enterprise Analytics
– Enterprise Continuous Delivery
– Enterprise Security
– VMware vSphere Builds

Even Scheduler Plugin

Change the default agent allocation algorithm of Jenkins to allocate jobs to free machines.

Challenge

The default Jenkins algorithm for allocating a machine directs Jenkins to always try to use the same node for the same job, unless it's not available. If the node is not available, the job will build elsewhere. However, as soon as the preferred node is available, the build comes back to it. This locality is useful in a large number of cases. For example, when performing SCM updates it is preferable to a fresh checkout, and some tools (like Maven) use local caches to speed up builds.

However, this pipeline may result in a scenario where a number of builds run on the same machine, while other machines sit idle. In situations like this, it is beneficial to be able to change the default behavior of Jenkins to target idle machines. This allows your builds to run faster and enables you to optimize available system resources.

Solution

The Even Scheduler plugin, available with the CloudBees Jenkins Platform, offers a different scheduling algorithm, one that we refer to as an "even load strategy." With this strategy, the scheduler prefers idle nodes absolutely over nodes that are doing something. The strength of this algorithm is that you are more likely to get a fully idle node. Quite simply, executing a build on a fully idle system is faster than executing the same thing on a partially loaded system.

More details are available on the CloudBees Jenkins Platform documentation website.

Fast Archiver Plugin

Faster builds with faster upload of archives from agents to masters.

Challenge

When a job running on a agent archives a build, the artifact is uploaded to the Jenkins master. The upload inflates the overall build time and consumes network bandwidth. Bigger archives have larger build times and consume more bandwidth.

Solution

The Fast Archiver plugin, available with the CloudBees Jenkins Platform, optimizes the default archiver behavior. The plugin compresses the delta between subsequent archives and transfers the compressed bits from agents to masters. This functionality speeds up builds and saves network bandwidth.

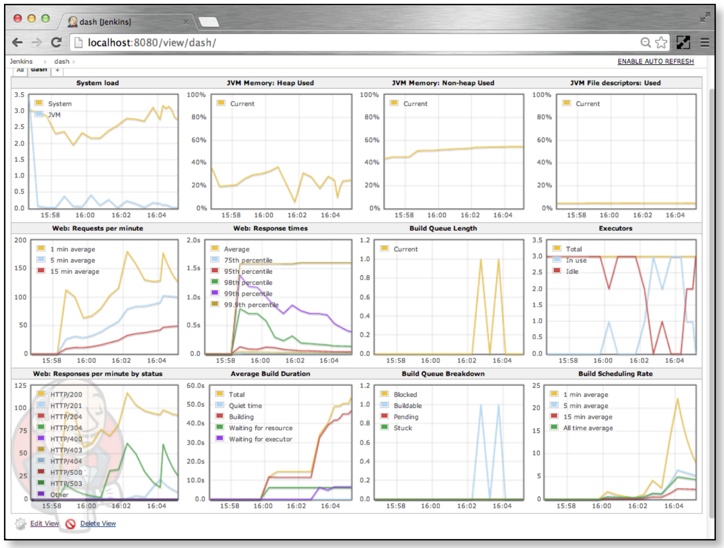
More details are available on the CloudBees Jenkins Platform documentation website.

NIO SSH Slaves Plugin

Define the bare-metal limits for virtual machines (VMs) being used as agents. This helps in faster runs when multiple builds are running multiple VMs on a single bare-metal machine.

›› Challenge

Master performance (for example, UI responsiveness) can degrade under a heavy load of builds, if those builds are serviced by SSH build agents.



›› Solution

The NIO SSH Slaves plugin based on Java non-blocking I/O provides much better throughput than the SSH agents available with Jenkins. This plugin uses fewer threads per agent as compared to SSH agents and enables better responsiveness on the part of the Jenkins master. In addition, when CPU resources aren't available, this plugin slows the build down to maintain Jenkins master responsiveness, as compared to SSH agents where the connection is lost, resulting in a build failure.

More details are available on the CloudBees Jenkins Platform documentation website.

<< Return to Introduction

Label Throttle Build Plugin

Define the bare-metal limits for VMs being used as agents. This helps in faster runs when multiple builds are running multiple VMs on a single bare-metal machine.

Challenge

When agents are really virtual machines (VMs), you may have multiple agents that actually share the same physical resources underneath. Whenever multiple jobs are fired off, they may execute on different VMs, but the VMs are on the same underlying machine. This results in slower builds due to thrashing of the VMs.

Solution

The Label Throttle plugin, available with the CloudBees Jenkins Platform, allows you to define a limit for your builds. You can group agents together, then assign a limit that specifies how many concurrent builds can be performed on the agents that belong to that group. In this way, the CloudBees Jenkins Platform avoids overloading your hypervisor host machine.

Thus, when you have a single-system hypervisor, such as VMware ESXi or VirtualBox, Jenkins might think that you have 10 agents with two executors each, but in reality 20 concurrent builds cannot really be executed. You can only run up to, say, four. This plugin will allow you to define four (in this case) as a limit for your builds, ensuring that the VMs do not thrash. The end result: faster builds, overall.

This plugin is very complementary in combination with the VMware Auto-Scaling plugin.

More details are available on the CloudBees Jenkins Platform documentation website.

- Jenkins for Teams
- Jenkins at Scale
  - Master Server and Builder
  - Optimized Utilization
    - Even Scheduler Plugin
    - Fast Archiver Plugin
    - NIO SSH Slaves Plugin
    - Label Throttle Build Plugin
    - Skip Next Build Plugin
  - Enterprise Management
  - Enterprise Analytics
  - Enterprise Continuous Delivery
  - Enterprise Security
  - VMware vSphere Builds

Skip Next Build Plugin

Stop running builds for a specified period of time.

>> Challenge

You need to skip building a job for a short period of time. This typically happens when:

- You are going to take external resources that the build utilizes offline for maintenance, and you don't want to be annoyed by all the build failure notices, or
- You are merging a major feature branch and you want to prevent builds until after the merge is completed.

While you can skip a build by disabling the project from the project configure page, you then need to remember to re-enable the project once the maintenance or merge activities are complete. Need we say more?

>> Solution

The Skip Next Build plugin, available with the CloudBees Jenkins Platform, allows you to skip building a project for a specified period of time. The build automatically turns on after the elapsed time period. The plugin helps administrators ignore a lot of false negatives and save time when managing resources.

More details are available on the CloudBees Jenkins Platform documentation website.

Enterprise Management Plugins

Manage team-specific plugins with the Custom Update Center plugin. >>

Reuse best practices with the Templates plugin. >>

Achieve enterprise-scale agent management with the Nodes Plus plugin. >>

>> Jenkins for Teams

>> Jenkins at Scale

- Master Server and Builder
- Optimized Utilization
- Enterprise Management
  - Custom Update Center Plugin
  - Templates Plugin
  - Nodes Plus Plugin
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds

Custom Update Center Plugin

Host your update centers easily. Ensure only approved versions of binaries and plugins are shared between teams.

Challenge

Today, it is very hard to have consistent governance and ensure compliance of plugins and binaries across teams. Teams developing products often end up working with different versions of plugins or binaries, especially if they are using different Jenkins masters.
The version mismatch results in subtle failures in the application that are discovered late in the development cycle. Debugging and solving these issues is difficult and time consuming.

Teams get around this problem by documenting the approved version of plugins or binaries. However, this solution is rarely enforceable. The more advanced solution is to host an update center; this is a cumbersome and error-prone process.

Solution

The Custom Update Center plugin, offered within the CloudBees Jenkins Platform, allows administrators to easily create their own update centers. Administrators set up their update centers and optionally inherit from approved update centers. They can upload plugins and optionally specify the version of the plugin that is available to downstream Jenkins instances. With this plugin, administrators have an enforceable mechanism to share plugins/ binaries between teams.

Thus, this plugin provides an enforceable governance mechanism to ensure compliance of plugins and binaries used across projects and teams.

More details are available on the CloudBees Jenkins Platform documentation website.

>> Jenkins for Teams
>> Jenkins at Scale
– Master Server and Builder
– Optimized Utilization
– Enterprise Management
– Custom Update Center Plugin
– Templates Plugin
– Nodes Plus Plugin
– Enterprise Analytics
– Enterprise Continuous Delivery
– Enterprise Security
– VMware vSphere Builds

Templates Plugin

Capture configuration information in one place that can then be propagated out to all dependent configurations.

Challenge

You have common configuration scenarios that are replicated in multiple places. Making changes to any of the scenarios requires you to go to each configuration and change it manually. The manual process is error-prone and cannot be replicated easily. Thus, you cannot easily share best practices within various projects or jobs.

Solution

The Templates plugin, available with CloudBees Jenkins Platform, captures the sameness of configuration in multiple places. Administrators define templates of jobs/build steps/publishers and replicate them while creating new jobs. Changes are made in one central location and are reflected in all dependent configurations.

The Templates plugin lets you define four types of templates:

Builder

Usually, developers have to create build steps as shell scripts to address enterprise requirements, such as those for: custom deployments, packaging, notification and QA integration. These scripts are used for a large set of jobs and only differ by a few attributes. Maintaining such scripts and avoiding copy/paste mistakes can be a nightmare for large Jenkins installations. This template creates a new type of builder and avoids the need for users to copy scripts from one job to another.

continued >>

Templates Plugin (cont.)

Job

In many enterprise contexts, it is common for two projects to differ only by the project name. Similarly, source code management, repositories or QA servers only differ by a sub-directory in a URL. For example, code may be hosted at svn.mycompany.com/project/trunk, a dedicated Maven repository may be set at maven.mycompany.com/repositories/project, a QA Sonar instance may run at project-sonar.mycompany.com/ and so forth. Job templates can be used to represent this model in an enterprise development environment by letting projects only define the minimal variable configuration to apply to the model. Thus, a job template offers a higher level of abstraction as it templates the definition of a whole job.

Folder

Administrators can configure a folder template that sets specific properties on a particular type of folder. Every time a new folder is created, it inherits those properties. Thus, the folder template is a way to define a standard structure for a folder so that it is applied with few parameters and in a simplified configuration.

The folder template can be used to define a set of jobs (that can themselves be defined by templates) to follow enterprise rules and best practices. Such a template defines the activities that a user will have to follow when using the template. You can then create a full wizard for a user to create the adequate folder and jobs structure for a new project.

Some examples of jobs might include:

- An SCM continuous integration job
- A continuous inspection nightly job to populate a Sonar QA server
- A continuous deployment job to push successful builds on a demo server

Auxiliary

Auxiliary templates only define attributes. They are not expected to be used by end-users, but are useful to encapsulate a reusable set of attribute definitions that can used by other templates.

More details are available on the CloudBees Jenkins Platform documentation website.

>> Jenkins for Teams
>> Jenkins at Scale
– Master Server and Builder
– Optimized Utilization
– Enterprise Management
– Custom Update Center Plugin
– Templates Plugin
– Nodes Plus Plugin
– Enterprise Analytics
– Enterprise Continuous Delivery
– Enterprise Security
– VMware vSphere Builds

Nodes Plus Plugin

Assign agent machine owners, who are then automatically notified of changes in agent machine availability.

›› Challenge

In a medium-to-large Jenkins installation, agents often belong to different groups or individuals. An agent machine failure is usually discovered accidentally. What follows the discovery can be a challenge: how to identify the owner of the agent machine, so the owner can get it running again.

›› Solution

The CloudBees Jenkins Platform solves this issue with the Nodes Plus plugin. This plugin allows the Jenkins administrator to assign an owner to the agent machine. The owner is notified of changes in build node availability so that machine failures can be more quickly addressed.

More details are available on the CloudBees Jenkins Platform documentation website.

›› Jenkins for Teams
›› Jenkins at Scale
– Master Server and Builder
– Optimized Utilization
– Enterprise Management
– Custom Update Center Plugin
– Templates Plugin
– Nodes Plus Plugin
– Enterprise Analytics
– Enterprise Continuous Delivery
– Enterprise Security
– VMware vSphere Builds

Enterprise Analytics Plugins

Get insight into plugin usage on a master with the Plugin Usage plugin. >>

>> Jenkins for Teams

>> Jenkins at Scale

- Master Server and Builder
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
  - Plugin Usage Plugin
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds

Plugin Usage Plugin

Track overall usage of plugins and associate them with specific jobs.

Challenge

Typically, administrators managing Jenkins have no insight into how many jobs are using a particular plugin. Thus, administrators don't know how many jobs/teams will be affected if a plugin is upgraded or deleted.

Solution

The Plugin Usage plugin, available with the [CloudBees Jenkins Platform](), scans the Jenkins configuration to produce a list of plugins and the jobs associated with them. Administrators can proactively manage the plugin installations by notifying job owners if performing an upgrade. They can also remove plugins that are not being used.

More details are available on the CloudBees Jenkins Platform documentation website.

<< Return to Introduction

Enterprise Continuous Delivery Plugins

See pipeline performance and developer feedback with the Pipeline Stage View plugin. >>

Restart builds from checkpointed locations when master and agent failures occur with the Checkpoints plugin. >>

Get easier navigability across chained jobs with the Consolidated Build View plugin. >>

>> Jenkins for Teams

>> Jenkins at Scale

- – Master Server and Builder
- – Optimized Utilization
- – Enterprise Management
- – Enterprise Analytics
- – Enterprise Continuous Delivery
  - – Pipeline Stage View Plugin
  - – Checkpoints Plugin
  - – Consolidated Build View Plugin
- – Enterprise Security
- – VMware vSphere Builds

Pipeline Stage View Plugin

Pipeline visualization allows workflow errors to be easily pinpointed.

>> Challenge

You know you've had a failure, but where did it occur in the pipeline? How far has your commit progressed? You can't easily tell — and time is wasted trying to figure it all out.

>> Solution

Pipeline Stage View allows you to readily pinpoint errors. Developers can see how far their commits have gone in the pipeline. Managers can see how effectively the overall pipeline process runs.

Pipeline visualization allows you to:

1. See stages executed in a run
2. See stages paused for human input
3. Gain easy access to generated artifacts
4. Restart the checkpointed runs

More details are available on the CloudBees Jenkins Platform documentation website.

<< Return to Introduction

Checkpoints Plugin

Eliminate schedule slips due to master and agent failures.

›› Challenge

Long-running software pipelines that fail on day six of a seven-day build cycle can have a dramatic, negative impact on software delivery schedules.

›› Solution

The Checkpoints plugin allows pipelines to be restarted at checkpoints, eliminating schedule delays due to master and agent failures. With checkpoint recovery, long-running pipelines can survive both Jenkins and infrastructure failures, while at the same time eliminating productivity losses incurred when a pipeline has to be restarted from the beginning.

More details are available on the CloudBees Jenkins Platform documentation website.

Consolidated Build View Plugin

Visualize results of downstream jobs that make up an overall project, with ability to navigate to the console log of each job.

Challenge

It is fairly common to model a pipeline through multiple downstream jobs that are triggered by an upstream job. The problem with this popular approach is that it is hard to visualize which builds were run and to easily navigate to the console of the builds.

Solution

The Consolidated Build View plugin, available with the CloudBees Jenkins Platform, helps to easily visualize the pipelined jobs, navigate to the job log console and obtain information about where the jobs have been run.

More details are available on the CloudBees Jenkins Platform documentation website.

Enterprise Security Plugins

Secure inter-master artifact exchange with the Secure Copy plugin. >>

>> Jenkins for Teams

>> Jenkins at Scale

- Master Server and Builder
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
  - Secure Copy Plugin
- VMware vSphere Builds

Secure Copy Plugin

Copy artifacts securely between teams on the same or different instances of Jenkins.

Challenge

Teams in different development groups — usually running different Jenkins masters — need to share artifacts.

They do so by directing the upstream job to place the artifacts in a common area. A team member then manually notifies the team working on the downstream job that the artifacts are available. The result of using a process such as this is that the artifacts are shared manually and insecurely.

Solution

The Secure Copy plugin, available with the CloudBees Jenkins Platform, allows teams to transfer artifacts securely, automatically and without manual notifications. A key-pair is generated out of band and set up in the sender and receiver jobs. Subsequently, every time an artifact is generated, it is picked up by the downstream job. The artifact cannot be downloaded by anyone else.

More details are available on the CloudBees Jenkins Platform documentation website.

›› Jenkins for Teams

›› Jenkins at Scale

- Master Server and Builder
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
  - Secure Copy Plugin
- VMware vSphere Builds

VMware vSphere Builds Plugins

Auto-scale your agents on vSphere with the VMware ESXi/vSphere Auto-Scaling plugin. >>

>> Jenkins for Teams

>> Jenkins at Scale

- Master Server and Builder
- Optimized Utilization
- Enterprise Management
- Enterprise Analytics
- Enterprise Continuous Delivery
- Enterprise Security
- VMware vSphere Builds
  - VMware ESXi/vSphere Auto-Scaling Plugin

VMware ESXi/vSphere Auto-Scaling Plugin

Better leverage existing VMware resources by using machines in VMware pools as agents.

>> Challenge

You have builds starved for agent machines, while elsewhere within your VMware vSphere/ESXi installation you have multiple machines that may be running idle. You would like to "rent" the excess capacity of these unused VMs for your build jobs.

>> Solution

The VMware vCenter Auto-Scaling plugin, available with the CloudBees Jenkins Platform, allows you to create agent machines that are VMware ESXi/vCenter VMs. You assign a virtual machine from VMware ESXi/vCenter to your build, allowing your build to interact with that VM.

This offers better resource utilization of your machines and substantial savings to organizations as they reuse machines instead of buying new machines to fill the demand for resources.

The basic idea of this plugin is as follows. On one side, you configure pools of VMs. Each pool consists of multiple, identical VMs — that is, those VMs in the same pool are exchangeable for the sake of builds/tests that you run. On the other side, you have Jenkins jobs that require a virtual machine. What this plugin essentially does is to rent a VM to a build that requires resources while it's building. Jenkins keeps track of which VM is used where, and makes sure two builds don't end up colliding on the same VM at the same time.

continued >>

VMware ESXi/vSphere Auto-Scaling Plugin

(cont.)

In addition to this basic bookkeeping, Jenkins can perform setup/teardown actions to VMs, when a VM is assigned to a build and returned to the pool, respectively.

The actions include:

- Power on
- Power off/suspend
- Revert to the last snapshot

A very typical example of this functionality is to have Jenkins revert the VM to the golden snapshot and power on in the beginning of the build, then have Jenkins shut down the VM at the end of it. This allows you to always run your tests in a consistently clean state.

The VMware vCenter plugin allows you to use folders in vCenter to simplify managing large numbers of VMs. You only have to point to a vApp folder and all machines are automatically picked up — allowing users to dynamically add/remove machines without changing configurations.

This plugin is very complementary when used in combination with the Label Throttle plugin.

More details are available on the CloudBees Jenkins Platform documentation website.

The CloudBees Jenkins Platform is built on top of open source Jenkins, an independent community project. Read more at: www.cloudbees.com/jenkins/about

CloudBees, Inc.
2001 Gateway Place
Suite 670W
San Jose, CA 95110
United States
www.cloudbees.com
info@cloudbees.com

<< Return to Introduction