DeployHub Plugin

DeployHub’s Plug-In for Jenkins

DeployHub is a continuous deployment tool which automates software releases integrated into your continuous integration workflow. It is offered as both an Open Source solution with a Pro upgrade for security, and audit reporting. Using Jenkins with DeployHub improves the software deployment step of your continuous integration process allowing testing and production teams to more easily adopt the developer defined Jenkins Workflow.   The DeployHub Jenkins Plugin is called by Jenkins as a step in a job to perform the software deployment, without the need for Jenkins 'slaves' at multiple end-points or one-off deploy scripts. DeployHub then tracks a Build to a component and an end target (physical, virtual, cloud, container.) This enables you to utilize Jenkins to orchestrate your Continuous Integration process, while DeployHub does the heavy lifting of packaging the application stack and deploying it across your testing and production environments. The DeployHub Jenkins Plug-in includes a "Build Engine" object which can be connected to a Jenkins server. Each "Build Engine" can have one or more "Build Jobs" which relate to a particular Jenkins Project on that Jenkins Server. These "Build Jobs" can be linked to one or more components in DeployHub.

Plugin Information
View DeployHub on the plugin site for more information.

Older versions of this plugin may not be safe to use. Please review the following warnings before using an older version:

- Credentials stored in plain text

How the DeployHub Jenkins Plugin Works

The linking of Components to a Jenkins Build Jobs allows DeployHub to track which Builds have been performed against which component. The Jenkins plug-in can be set to simply "notify" DeployHub that a Build has been performed. Since DeployHub knows which Jenkins Project is linked to which Build Job and which Component is linked to which Build Job, it can track Builds for each Component with an associated Jenkins Project.

Should a deployment be performed, DeployHub takes the Jenkins Build number into account when determining if a component should be deployed. Even if the Component Version has not changed, a new Build will result in a re-deployment. Jenkins Build Logs can be viewed directly from inside the DeployHub dashboard, minimizing the amount of browser activity required to view Build details. For example, a user can see a Build Artifact on a Target Server and open the details of the generating Build directly from the Server *Details* page. This information is then tracked in a Continuous Feedback Loop providing a clear map of all artifacts associated to a deployment.

Configuration of the DeployHub Jenkins Plugin

The DeployHub plug-in is displayed in the Jenkins Configure window, and provides a way to configure what is required from DeployHub. The basic configuration, which shows under the title “Use DeployHub to run a deployment”, includes the following:

Username and Password	Determines security access to the various objects within DeployHub, including the *Applications* and *Components* that are available from the Jenkins Plug-In.
Target Environment	A DeployHub *Environment* contains all of the *Servers* that will be deployed to.
Application	A DeployHub *Application* contains all of the *Components* that make up a deployment. It is deployed against an *Environment*.
Wait for Deployment to Complete	This tells Jenkins to wait until DeployHub is finished with the deployment before moving on to the next step in the Job. If this option is chosen, the remaining steps in the Job will only run if the deployment was successful. If left unchecked, Jenkins will continue to the next step in the Job as soon as DeployHub begins the deployment, and ignore the success or failure of the deployment.
Use Advanced Version Selection	If this is checked, other options will appear that will allow you to take advantage of the capabilities that are available with Applications within DeployHub. These include: • Find Latest Version - If Latest Version is approved, create a new version • Use Component Selection • Set Component Attributes • Set Application Attributes