Plugin Selection for the Setup Dialog

This is a draft document for Jenkins 2.0. None of this is final.

Based on this mailing list post.

Introduction

Jenkins 2.0 will include a setup dialog that pops up when Jenkins starts for the first time. It intends to help people new to Jenkins to get started quickly, and present them with a reasonably sized set of plugins to choose from for customizing their instance, of they so choose. To this end, not every plugin will be presented here. While the extensibility of Jenkins is one of its strengths, exposing new users to the 1200+ available plugins before they've even started using Jenkins isn't a reasonable approach.

Selection Criteria

Plugins qualify for inclusion in the list based on a number of criteria:

- Plugin popularity (both in absolute numbers and growth rate), which can serve as a stand-in for plugin usefulness unless we believe this to be inaccurate for specific reasons (e.g. CVS Plugin and Monitor External Job being installed a lot, but only because they are bundled).
- Does the plugin integrate well with Jenkins? Does it work e.g. with concurrent builds, in secured Jenkins, and distributed Jenkins (master + slave setup)? People new to Jenkins should not be surprised by a plugin offered here undermining or not working with these features.
- Is the plugin more than just superficial themeing/UI? While those have their place and are popular, advertising plugins changing the UI out of the box would be very weird (Simple Theme Plugin, Green Balls)
- Are the features it provides useful to someone fairly new to Jenkins? If the plugin requires someone to e.g. be familiar with Jenkins internals before they can use it successfully, or to have multiple Jenkins instances, it may not be a good choice for this dialog. For example, if the script console were a plugin, it would probably not be on here.
- Even if it's not popular, is it a reasonably common SCM? SCMs are so fundamental to successful Jenkins use that these shouldn't be as heavily restricted as some of the other categories.
- The plugin most work without requiring a restart of Jenkins.
- The plugin must be discoverable when installed, and at least allow users to get started without outside documentation. (This disqualifies Build Token Root Plugin.)
- The plugin is hosted in the `jenkinsci` GitHub organization
- The plugin must be maintained reasonably well (e.g. was updated in the last 18 months or so, and ideally has a dedicated maintainer)

Categories

Given the diverse feature set of Jenkins plugins, it is difficult to define a set of categories that can be used to group plugins without misrepresenting them (and having every plugin only how up once in the list, rather than multiple times). The following categories appear to adequately provide a high-level view of plugin functionality in Jenkins:

- **General**: Anything that has no better category. Depending on contents, these may result in the creation of new categories. Possible examples are Rebuilder Plugin, Timestamper Plugin, or Translation Assistance Plugin.
- **Organization and Administration**: Plugins that help Jenkins administrators accomplish their goals, and that help keep order. Examples are Monitoring Plugin, Antisamy Markup Formatter, or Disk Usage Plugin.
- **Build Features** are plugins that provide features to builds/jobs typically beyond an SCM, build step or publisher. SSH Agent Plugin, AnsiColor Plugin, and Timestamper.
  - TODO: Kind of a catch-all bucket for these plugins. Can we find something better?
- **Build Tools** for plugins providing build tool integration, e.g. NodeJS Plugin, Maven Plugin, Gradle Plugin.
- **Build Analysis and Reporting**: Plugins in this category process reports or data generated during builds, such as JUnit Plugin, Build Failure Analyzer, or Checkstyle Plugin.
- **Build Pipelines and Continuous Delivery** is for plugins that allow creating more complex chains of actions that would reasonably be possible in a single freestyle job. Examples are Copy Artifact Plugin, Parameterized Trigger Plugin, or Workflow Plugin.
- **SCM** provides a selection of plugins that integrate with SCM tools, e.g. Subversion Plugin, GitHub Plugin, or GitHub Pull Request Builder Plugin.
- **Distributed Builds ~~and Containers~~** are plugins that provide different build environments to builds, such as Docker Plugin, Windows Slaves Plugin, or Matrix Project Plugin.
  - TODO: Container creation is different from this, so probably doesn't belong here.
  - Currently no container plugins made the cut.
- **User Management and Security** contains all plugins related to securing Jenkins, such as Matrix Authorization Plugin, LDAP Plugin, or Secure Requester Whitelist Plugin.
  - Some overlap with Organization and Administration, e.g. Antisamy Markup Formatter. The difference is that these plugins are mostly useless when security is disabled.
- **Notifications and Publishing** are plugins that notify other systems about builds and build results, or publish data, typically on remote systems. Examples are JIRA Plugin, Mailer Plugin, Email Extension Plugin, or Slack Plugin.
  - Likely overlap with Build Analysis and Reporting. The difference is mostly that plugins in this category typically require specific other systems or software to work.

Any of these categories will be present as necessary for the plugin selection.