Elastic Axis

This plugin is a power up for the multi configuration jobs allowing you to configure jobs to run on all slaves under a single label.

Description

Ever wanted to have your own personal cloud and use Jenkins a cloud manager, allowing you to create distributed tasks to run over an array of computers? I'm not talking about just running integration tests, I'm talking about real **distributed tasks**, the sort where the more computers the faster you complete the task.

Then, this is the right plugin for you.

Jenkins is an excellent tool for several continuous integration tasks, but it is still in the early stages for helping you to create your own personal cloud of computers. The multi configuration job helps you running a task over a fixed number of computers, but the standard axis has several drawbacks for this kind of task. If you have tenths of computers to run several jobs in your personal array of computers, for example, you have to configure each job and include every node individually. And you'll have to do it again, when you create new jobs. If that's not tedious enough, when you add a new node to your cloud, you have to reconfigure every job to start receiving the new node's contribution.

Another serious drawback: if for some reason one of the nodes is offline, the job will not complete, even when the other nodes are capable of completing the task taking a longer time.

So, this plugin address these two flaws of the standard label axis.

Running over an Elastic Axis

When you create an Elastic Axis, you specify one or more labels where you want to run your job. That is, instead of specifying a fixed number of computers, you let the axis find out on the fly all the computers in that label to run your job. By doing this, you avoid the problem of the standard label, because new nodes are added to the label and not to the jobs. As soon as you add a new computer to the labels, jobs configured to used the Elastic Axis over this label will immediatly benefit.

Another issue addressed by the Elastic Axis is the offline nodes. Sometimes, the computers in your cloud might end up in maintenance or go offline for some reason. With the Elastic Axis, you can check an option instructing it to skip these nodes if they are not available, and therefore, your tasks may take longer, but the job will complete without hanging out because of offline nodes.

A Massive Test Case

In our company we have more than 25k functional tests that take 12 hours or so to run on a standalone desktop computer. But our test running engine was designed in such a way that you can configure more than one instance to run the set of tests. So, by using 20 modest computing units, we are able to run the whole set of tests in about an hour. By using Jenkins multiconfiguration jobs, automatically running the tests over 20 computer slaves becomes a reality.

Now, we have several production branchs of our product, all of them with the same 25k tests that should run as often as possible to make sure the fixes are not causing regression. And they all share the same grid of computers on jenkins to run the tests.

This is where the Elastic Axis plugin becomes a great tool. By using the plugin, we are able to easly manage the computing resources and change it without ever reconfiguring the jobs. The kind of task we run doesn't care whether you have 20 or 50 computers, it will complete even with a single unit. But by adding more computers, we can speed it and even the dauting task of running 25k functional tests that access databases turns out to run on acceptable time for continuous integration purposes.