Cognitive Interaction Toolkit - Support #266

Semantics of the "slaves" feature

2014-06-07 09:28 - Florian Lier

Status:	Resolved	**Start date:**	2014-06-07
Priority:	High	**Due date:**	
Assignee:		**% Done:**	100%
Category:	build-generator	**Estimated time:**	0.00 hour
Target version:		**Spent time:**	0.00 hour

Description

It is not completely clear to me how the "slaves" feature works.

```
"slaves":        [ "master" ],
"slaves.ubuntu": [ "@{slaves}" ],
"slaves.linux":  [ "@{slaves.ubuntu}" ],
```

This means, that the used distribution is build on "master" and *also* (?) on slaves labelled with ubuntu, or linux?

The problem is that *master* is fine for local building (aka mobile Jenkins) however, if we create jobs on ci.toolkit
we don't want the distro to be built on the master. Do one need to change this on the fly before actually adding jobs
to ci.toolkit?

Associated revisions

Revision 012181a8 - 2015-02-17 20:05 - Johannes Wienke

Adapt distributions according to new requirements

- Add patterns for orchestration job names, fixes #324
- remove slaves from distributions since they have reasonable defaults
  in the templates now, fixes #266
- remove bla-name variable, refs #241
- Clean up name and folder variables as much as possible, refs #228

Probably fixes #290

History

#1 - 2014-06-08 20:31 - Jan Moringen

- Status changed from New to Feedback

Explanation of Concrete Example and Current State

The snippet

```
"slaves":        [ "master" ],
"slaves.ubuntu": [ "@{slaves}" ],
"slaves.linux":  [ "@{slaves.ubuntu}" ],
```

should be a "user-level" (i.e. no special semantic in the generator) cascade of variables, slaves being the most generic variable, slaves.ubuntu being
the most specific one. The cascade is *achieved* using the variable substitution mechanism of the generator (See section in source:README#L49 for

some notes regarding the substitution syntax).

The idea is that the value can be overwritten at each level of specificity, for example "for all Linux build jobs" or "for all Ubuntu build jobs".

Future Work

The whole cascade and its use by templates is a "only" a matter *convention*. However, we should probably still improve and document this convention.

In my opinion, the goal would be having specific variables for each kind of deployment and a cascade of default values for convenience.

- For example, a kind of deployment could be "local Jenkins, distribution installation mode, Ubuntu Linux platform". This deployment could use a variable like slaves.local.distribution.ubuntu which may default to the value of slaves.local.distribution.linux and so on.
- The toolkit case could use something like slaves.toolkit.distribution.ubuntu.

Next Steps?

To stress the point again: this a matter of *organization, planning and convention*, not a current or future feature of the generator. As a result, we could start by collecting use-cases like the above example and start to flesh out the requirements and details of those. I would be happy to adapt the generator where necessary if we had this kind of use-case analysis.

#2 - 2014-06-10 18:18 - Johannes Wienke

- Assignee deleted (Johannes Wienke)

#3 - 2015-02-17 20:18 - Johannes Wienke

- Status changed from Feedback to Resolved

- % Done changed from 0 to 100

Applied in changeset projects|commit:012181a87a29a02758751d9c8d4595357a165be5.