Arvados - Feature #4561

[SDKs] Refactor run-command so it can be used as an SDK by scripts in a git tree

11/17/2014 09:10 PM - Tom Clegg

Status: New
Priority: Normal
Assigned To: SDKs
Category: SDKs
Target version: Arvados Future Sprints

Start date: 
Due date: 
% Done: 0%
Estimated time: 0.00 hour

Description
Currently you have two main options for running a job:

1. Put the program you need in a docker image. Use run-command from the arvados tree to wrap it as a crunch job.
2. Write a native crunch script in your git tree.

The first option forces you to save a new docker image in order to run a new version of your program. Unacceptable!

There are several features of run-command that make it convenient and attractive to beginners. However, it forces you to use a totally different approach to developing and running scripts, an approach which prevents you from doing some important things:

• Keeping your code in revision control,
• Using the same code in more than one pipeline template,
• Writing jobs in the programming language of your choice.

There is no migration path from a simple run-command job to a non-trivial program, so the developer is forced to choose: live with run-command's custom JSON-based programming language, or abandon the existing pipelines and all of run-command's advantages, and rewrite everything in a normal language like Python.

This can be addressed by refactoring run-command as a set of utilities and features, rather than a programming language that can only be used inside crunch jobs.

• The "run-command language" interpreter should be runnable like other interpreters (#!whatever-the-language-is-to-be-called).
• Convenience features like "store output dir contents in Keep and set success=true at end of task" should be ported to other SDKs too (most obviously bash), so authors can migrate from the JSON language to a normal language.
• It should be possible to provide a crunch script in any language by copying the script itself into the job record. (This makes it possible to run jobs without touching git.)
  • The whole point of the run-command language is to already be JSON-encoded, which means it should be provided in a serialized attribute like script_parameters. (Other languages are just text, but that can also fit in a serialized field.)
  • The name of the "script" attribute already suggests that you can put a script in it. This could change type (from varchar(255) to text) and become a serialized field capable of containing a string or a hash. This means we can't rely on "script" to be a short name suitable for displaying in UI (a problem we already have with run-command jobs: the program has no name, so we display the name of the language instead).
  • We could support passing the name of an interpreter in "script" (e.g., "run-command" or "python") and passing the program itself in script_parameters[stdin]. We would have to treat string and hash/array cases differently: if stdin is a string, pass the string value, but if stdin is a hash or array, pass its JSON encoding.
• It should be possible to move your run-command program into a git tree and run it from there.
  • Currently, this can be done awkwardly by copying some version of run-command into your own git tree.
  • With #4027, we can make run-command's features available through the SDK.
  • Then we just need #whatever-the-language-is-called to work, or some other way to invoke run-command from the installed SDK, rather than requiring it to be in $CRUNCH_SRC/crunch_scripts/.

Related issues:
Related to Arvados - Bug #4562: [Documentation] Wiki page: explain appropriat... Resolved 01/16/2015
Related to Arvados - Story #3820: [Crunch] Jobs need code from multiple git r... Closed 09/05/2014
Related to Arvados - Story #3603: [Crunch] Design good Crunch task API, inclu... Closed 08/27/2014

History
#1 - 11/17/2014 09:18 PM - Tom Clegg
  - Description updated
  - Category set to Crunch
#2 - 01/20/2015 07:56 PM - Tom Clegg
- Subject changed from [Crunch] Support using run-command to wrap a script in a separate git tree. to [SDKs] Refactor run-command so it can be used as an SDK by scripts in a git tree
- Description updated
- Category changed from Crunch to SDKs
- Target version set to Arvados Future Sprints
- Story points changed from 1.0 to 2.0

#3 - 02/04/2015 03:47 PM - Tom Clegg
- Description updated

#4 - 02/04/2015 03:55 PM - Tom Clegg
- Description updated

#5 - 02/04/2015 04:01 PM - Peter Amstutz
(comments from IRC)

Now that we have the "deploy SDK into container" feature, moving most of run-command's functionality into the SDK is a good idea. The main reason I didn't do that was because the deployment cycle for the SDK was much less convenient than for a crunch script. In the long term I'd prefer to deprecate run-command in favor of OWL (and send run-command to the scrapyard for spare parts).