Arvados - Bug #4562

[Documentation] Wiki page: explain appropriate use cases for arv-run vs. run-command vs. writing your own crunch script.

11/17/2014 09:20 PM - Tom Clegg

Status:	Resolved	**Start date:**	01/16/2015
Priority:	Normal	**Due date:**	
Assigned To:	Brett Smith	**% Done:**	100%
Category:	Documentation	**Estimated time:**	0.00 hour
Target version:	2015-02-18 sprint		

Description

These general approaches need to be introduced and explained between http://doc.arvados.org/user/tutorials/intro-crunch.html and http://doc.arvados.org/user/tutorials/running-external-program.html.

Executive summary

- arv-run makes sense for simple fan-out commands
- run-command makes sense when you already have a command line tool installed in a docker image, and you just want to invoke it as part of a compute workflow/pipeline
- writing your own crunch script makes sense if you want your code to be in revision control, you want more control of concurrency patterns, you need better performance, or you'd just rather write everything in python.
- third option (use run-command to wrap something that lives in your own git tree) isn't very well supported but as a workaround you could copy the run-command stuff into your own git tree.

This page should clearly explain the limitations of arv-run and run-command so that users know when to switch between the approaches to run things.

Subtasks:

Task # 5015: Review branch 4562-crunch-tools-docs-wip	**Resolved**

Related issues:

Related to Arvados - Feature #4561: [SDKs] Refactor run-command so it can be ...	**New**	
Related to Arvados - Feature #4743: [Crunch] Upgrade run-command regex or bas...	**Closed**	**12/08/2014**

Associated revisions

Revision 088bc7b9 - 01/30/2015 06:58 PM - Brett Smith

Merge branch '4562-crunch-tools-docs-wip'

Closes #4562, #5015.

History

#1 - 11/17/2014 09:48 PM - Tom Clegg

- Description updated

- Category set to Documentation

#2 - 12/31/2014 07:51 PM - Ward Vandewege

- Subject changed from [Documentation] Clarify the appropriate use cases for run-command vs. writing your own crunch script. to [Documentation] Clarify the appropriate use cases for arv-run vs. run-command vs. writing your own crunch script.

- Description updated

#3 - 01/06/2015 03:56 PM - Tom Clegg

- Subject changed from [Documentation] Clarify the appropriate use cases for arv-run vs. run-command vs. writing your own crunch script. to [Documentation] Wiki page: explain appropriate use cases for arv-run vs. run-command vs. writing your own crunch script.

#4 - 01/07/2015 08:30 PM - Tom Clegg

- Target version changed from Arvados Future Sprints to 2015-01-28 Sprint

#5 - 01/07/2015 08:38 PM - Brett Smith

- Assigned To set to Brett Smith

#6 - 01/16/2015 10:35 PM - Brett Smith

A draft wiki page is up. Right now it's not linked from anywhere, to minimize the chances of people stumbling on it prematurely. I'll fix that after it's gone through review.

#7 - 01/19/2015 09:08 PM - Peter Amstutz

Some general comments:

- Who is the audience? This should state that up front. Are readers expected to have already gone through the tutorial? I suspect the audience that will get the most out of a page like this are users who have run a few pipelines through workbench (gaining a passing familiarity with Arvados/Crunch) and have decided that now they want to start porting their own analysis.
- This is missing the "How" aspect of the title. It would greatly benefit from discussion and examples of how each approach could be applied to given situation and the trade offs that are involved.
- The descriptions of arv-run and run-command are not clear. Consider borrowing text from the user guide to summarize those tools in a few sentences. Possibly note that arv-run is actually an "interactive" frontend for run-command.
- It doesn't make sense to discuss run-command and crunch scripts without also discussing pipelines first. "Combining run-command and custom Crunch scripts in a pipeline" should be moved up.

#8 - 01/22/2015 10:22 PM - Brett Smith

Peter Amstutz wrote:

> - Who is the audience? This should state that up front. Are readers expected to have already gone through the tutorial? I suspect the audience that will get the most out of a page like this are users who have run a few pipelines through workbench (gaining a passing familiarity with Arvados/Crunch) and have decided that now they want to start porting their own analysis.
> - The descriptions of arv-run and run-command are not clear. Consider borrowing text from the user guide to summarize those tools in a few sentences. Possibly note that arv-run is actually an "interactive" frontend for run-command.

Done.

> - This is missing the "How" aspect of the title. It would greatly benefit from discussion and examples of how each approach could be applied to given situation and the trade offs that are involved.
> - It doesn't make sense to discuss run-command and crunch scripts without also discussing pipelines first. "Combining run-command and custom Crunch scripts in a pipeline" should be moved up.

I don't think it's appropriate to include full examples, because it's hard to really get an apples-to-apples comparison from them. How can you compare an arv-run call to a pipeline that uses a combination of run-command jobs and custom Crunch scripts? Each section includes a paragraph that gives a high-level overview of what the tool's strengths and weaknesses, and then provides a link to more documentation detailing how to use it. If those discussions aren't helpful enough, then I think that needs to be tackled more directly. That's really the core of this story.

The point about pipelines is a good one. I liked it so much, I took the idea further. Now the basic presentation outline is, "You can run a pipeline with arv-run, or write your own pipeline template and run that. Here are the tools you can use when you write your own pipeline template." I think this helps clarify how the different pieces relate—it's reflected in the organization of the page.

If there's still a mismatch with the title, I feel like the title is at least partly to blame. Based on the story, I feel like the title very strictly ought to be something like, "Comparison of methods to run analysis work in Arvados," but that felt really unwieldy, and I ended up settling on this "How to" title. But I'm very open to better suggestions there.

Thanks.

#9 - 01/28/2015 06:41 PM - Peter Amstutz

@01/22/2015 05:10 pm

Much better.

Here's a brainstorm, how about a table that provides some kind of brief side-by-side summary? Some ideas:

	arv-run	run-command	crunch script
can set up entire run on the command line	yes	no	no
use files from keep	yes	yes	yes
automatically upload local files	yes	no	no
wrap existing tools	yes	yes	yes, using subprocess module
parallelize over list of files	yes	yes	must spawn parallel tasks explicitly
automatically upload output	yes	yes	no

supports control flow	no	no	yes
usable from workbench	no	yes	yes

#10 - 01/29/2015 03:24 PM - Brett Smith

Peter Amstutz wrote:

> Here's a brainstorm, how about a table that provides some kind of brief side-by-side summary?

I am the dude personally responsible for this monstrous reference table, and the experience has kind of left me sour on using tables to compare situations with nuanced differences. Because tables have to be brief, it's difficult for them to capture all the criteria that are relevant to different readers. Looking over what you're got here, questions that pop to mind are like, does it make sense to say Crunch script don't upload local output, when it just takes three lines of Python to do so (using CollectionWriter.write_directory_tree)? Does it make sense to say that run_command and Crunch scripts are usable from Workbench, when there's currently no Workbench UI for authoring pipeline templates?

More generally, the table collapses the distinction between pipeline-level tools and job-level tools, when the main thrust of the last revision was to clarify and emphasize that distinction.

It's a fair idea, and I could still be convinced, but I'm admittedly skeptical that the effort we put into it will pay off for our users.

#11 - 01/29/2015 03:37 PM - Peter Amstutz

I don't want to spend a lot of time wrangling over this, so I'll try to make this the last round of comments.

My concern with the current draft is that is a bit too wordy to be a "brief" summary, while at the same time not being detailed enough to actually illustrate the differences concretely (instead fobbing the user off onto the main documentation.) So it should either be tightened up to be more digestible (which is where my suggestion of adding a table came from) or expanded with small examples (which I suggested in my initial comments).

A really good way to illustrate the differences ("a picture is worth 1000 words") would be to write out the same task using arv-run, run-command, and a crunch script.

#12 - 01/29/2015 04:41 PM - Ward Vandewege

- Status changed from New to In Progress

#13 - 01/29/2015 07:41 PM - Brett Smith

- Target version changed from 2015-01-28 Sprint to 2015-02-18 sprint

#14 - 01/30/2015 04:28 PM - Brett Smith

Peter Amstutz wrote:

> My concern with the current draft is that is a bit too wordy to be a "brief" summary, while at the same time not being detailed enough to actually illustrate the differences concretely (instead fobbing the user off onto the main documentation.) So it should either be tightened up to be more digestible (which is where my suggestion of adding a table came from) or expanded with small examples (which I suggested in my initial comments).

Given what's specified in the story, I feel like it can't get any more brief—each tool just gets a couple of sentences explaining what it does, what it's good for, and what its limitations are. Those are all required by the description.

So let's make it longer. One of my concerns about including examples was documentation drift: the examples in the wiki getting out of date as the tool got updated. Following our IRC conversation with Tom, there's now a branch up for review that adds this page to the User Guide. It incorporates examples that already exist to briefly illustrate a good application of each tool.

I also trimmed some of the intro content, now that we can kind of rely on the larger context of the User Guide to provide that. The rest of the writing is the same as the previous wiki draft. Let me know what you think of this.

> A really good way to illustrate the differences ("a picture is worth 1000 words") would be to write out the same task using arv-run, run-command, and a crunch script.

I feel like that would run counter to the page's message. The whole idea here is that different tools are best suited to different tasks. Showing them all running the same task would undermine that message, and give users the wrong idea about their relative strengths and weaknesses.

Thanks.

#15 - 01/30/2015 07:00 PM - Brett Smith

- Status changed from In Progress to Resolved

- % Done changed from 0 to 100

Applied in changeset arvados|commit:088bc7b980536ee2b27c8abf4bfc09c348000589.