Pulp - Issue #5706

base_version can cause content to show as added, even though it was in the previous version

11/13/2019 03:43 PM - bmbouter

Status:	CLOSED - CURRENTRELEASE	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	fao89	**Estimated time:**	0:00 hour
Category:			
Sprint/Milestone:	3.0.1		
Severity:	2. Medium	**Groomed:**	No
Version:		**Sprint Candidate:**	No
Platform Release:		**Tags:**	
OS:		**Sprint:**	Sprint 63
Triaged:	Yes	**Quarter:**	

Description

This was originally reported on IRC by @gmbnomis.

Reproducer

1. Create empty RepositoryVersion, version 1.
2. Create a second RepositoryVersion by adding content unit B
3. Create a third RepositoryVersion, by adding content unit A AND use version 1 as the base_version

Issue

Observe that for RepositoryVersion 3, the added field shows A & B to be considered added, even though B was already present in RepositoryVersion 2.

Associated revisions

Revision 420dd1ff - 01/07/2020 04:45 PM - Fabricio Aguiar

added/removed content according to a given Version

https://pulp.plan.io/issues/5706 closes #5706

Revision baea1263 - 01/07/2020 07:32 PM - Fabricio Aguiar

added/removed content according to a given Version

https://pulp.plan.io/issues/5706 closes #5706

(cherry picked from commit 420dd1ff2f326db454b216f33d848d5267489dfb)

Revision bca447bd - 01/07/2020 09:16 PM - Fabricio Aguiar

added/removed content according to a given Version

https://pulp.plan.io/issues/5706 closes #5706

(cherry picked from commit 420dd1ff2f326db454b216f33d848d5267489dfb)

Revision 987e42de - 01/07/2020 09:16 PM - daviddavis

Renaming method for RepositoryContent relationships

ref #5706 https://pulp.plan.io/issues/5706

Revision 664bbe8c - 01/08/2020 07:34 PM - daviddavis

Renaming method for RepositoryContent relationships

ref #5706 https://pulp.plan.io/issues/5706

(cherry picked from commit 987e42de3da23ea07fb5a8f0d2bde06107d3844c)

Revision 62b6d781 - 01/10/2020 11:28 PM - daviddavis

Renaming method for RepositoryContent relationships

ref #5706 https://pulp.plan.io/issues/5706

(cherry picked from commit 987e42de3da23ea07fb5a8f0d2bde06107d3844c)

History

#1 - 11/13/2019 03:47 PM - bmbouter

- Description updated

#2 - 11/13/2019 09:44 PM - gmbnomis

bmbouter wrote:

> This was originally reported on IRC by @gmbnomis.

I got this one wrong and probably created lots of confusion. The reproducer is the same, but what I meant to say is:

Issue

Observe that for RepositoryVersion 3, the added field shows A to be considered added, and B as removed.

This is consistent with how the added and removed sets are defined (they show additions/removals with respect to the previous "latest" version (i.e. /2/ in this case)).

However, when a repo version finalizer looks at a repo version changes, it is interested in changes relative to the base version.

For example, assume we use a finalizer that removes dependent content for a removed content unit. Suppose that A depends on B. If the finalizer looks at the .removed set (containing B), the finalizer will remove A, which does not make sense.

Thus finalizers need other change sets than .added and .deleted. Currently, RepositoryVersion does not offer such sets comparing two (arbitrary) repo versions.

#3 - 11/15/2019 03:21 PM - daviddavis

- Tracker changed from Issue to Task

- % Done set to 0

#4 - 11/19/2019 04:20 PM - daviddavis

- Tracker changed from Task to Issue

- Severity set to 2. Medium

- Triaged set to No

#5 - 11/19/2019 04:36 PM - fao89

- Triaged changed from No to Yes

- Sprint set to Sprint 62

#6 - 12/06/2019 03:43 PM - rchan

- Sprint changed from Sprint 62 to Sprint 63

#7 - 12/09/2019 05:29 PM - daviddavis

Currently in RPM, we get the base (or last) version and find the content that has been added and removed. It's not a good solution but it's a workaround now at least.

My initial feeling is that added() and removed() should stay the same (given that 3.0 has released). But maybe we can either add an optional argument to alter their behavior or maybe we should introduce a new set of methods?

#8 - 12/13/2019 04:43 PM - fao89

I would go with an optional kwarg,
keeping added and removed as is for the default behavior, and changing it when kwarg is passed

#9 - 12/13/2019 04:51 PM - ipanova@redhat.com

One of the possible solutions would be to check for the optional kwarg which would be the base_version and if present then take **that** version instead of 'latest' version.

#10 - 12/15/2019 12:12 PM - gmbnomis

+1 for optional base_version parameter

#11 - 12/18/2019 07:00 PM - fao89

- Status changed from NEW to ASSIGNED

- Assignee set to fao89

#12 - 12/18/2019 08:30 PM - fao89

- Status changed from ASSIGNED to POST

https://github.com/pulp/pulpcore/pull/460

#13 - 01/07/2020 04:45 PM - Anonymous

- Status changed from POST to MODIFIED

Applied in changeset pulpcore|420dd1ff2f326db454b216f33d848d5267489dfb.

#14 - 01/07/2020 04:49 PM - daviddavis

Follow up PR to rename the method: https://github.com/pulp/pulpcore/pull/480

#15 - 01/07/2020 07:32 PM - Anonymous

Applied in changeset pulpcore|baea126391a5875c7de4cc57bc2af9de949f694f.

#16 - 01/07/2020 09:16 PM - Anonymous

Applied in changeset pulpcore|bca447bd85ff339f0cfa9461cb07d1341b84a057.

#17 - 01/16/2020 07:16 PM - bmbouter

- Sprint/Milestone set to 3.0.1

#18 - 01/16/2020 07:17 PM - bmbouter

- Status changed from MODIFIED to CLOSED - CURRENTRELEASE