Pulp - Story #8687

DeclarativeVersion API doesn't accomodate metadata mirroring use cases

05/03/2021 10:13 PM - dalley

Status:	NEW		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:			**Estimated time:**	0:00 hour
Sprint/Milestone:				
Platform Release:			**Tags:**	
Groomed:	No		**Sprint:**	
Sprint Candidate:	No		**Quarter:**	

Description

Metadata mirroring is difficult to accomplish with the current state of the DeclarativeVersion API. The current workflow looks like this:

1. The plugin writer defines a "First Stage" with a "run()" method, which downloads metadata from the remote, processes it, and puts DeclarativeContent onto the "out" queue for further processing by subsequent stages.
2. The plugin writer creates an instance of the first stage and passes it to DeclarativeVersion
3. The plugin writer creates an instance of DeclarativeVersion and calls the "create()" method on it
4. "dv.create()" initializes a pipeline of stages and then runs them in an asyncio event loop until complete

There are a few problems:

- We can't create a Publication until after the Repository Version has been created
- The RepositoryVersion is created by DeclarativeVersion.create()
- The downloading of the metadata all happens inside the FirstStage

So all of the metadata downloading state is contained within the FirstStage, but FirstStage cannot save the PublishedMetadata, because there is no Repository Version at that point in time, and therefore can be no Publication.

A poor solution to this problem is to store the FirstStage state somewhere where it can later be used once the repository version has been created. See: https://github.com/pulp/pulp_file/pull/502

A better solution would be to either:

1. Drive the DeclarativeVersion machinery from the outside (keep the state outside to begin with rather than trying to pass it from inside FirstStage to the outside), or
2. Change DeclarativeVersion so that it can handle published metadata and published artifacts produced by FirstStage, and create a Publication from them. And determine what the API for passing that Publication back looks like.

Related issues:

Related to RPM Support - Story #6353: As a user, I can mirror RPM repository ...	**CLOSED - CURRENTRELEASE**

History

#1 - 05/03/2021 10:16 PM - dalley

- Description updated

#2 - 05/04/2021 04:46 PM - daviddavis

- Tracker changed from Issue to Story

- % Done set to 0

- Severity deleted (2. Medium)

- Triaged deleted (No)

#3 - 05/05/2021 05:46 PM - dalley

- Related to Story #6353: As a user, I can mirror RPM repository content and metadata added