

Pulp - Story #3953

ContentUnitSaver should support creating content with related models inside its transaction

08/30/2018 04:31 PM - jortel@redhat.com

Status:	CLOSED - CURRENTRELEASE	Start date:	
Priority:	Normal	Due date:	
Assignee:	bmbouter	% Done:	100%
Category:		Estimated time:	0:00 hour
Sprint/Milestone:	3.0.0	Tags:	
Platform Release:		Sprint:	Sprint 42
Groomed:	Yes	Quarter:	
Sprint Candidate:	No		

Description

Problem

DeclarativeVersion (*stages*) currently supports creating content consisting of a single *Content* model. Many plugins such as RPM and Docker have complex content models. That is, each content (unit) is stored in multiple tables. The insert into these tables needs to be committed atomically to prevent storing incomplete content. Further, the inserts need to happen in relational (referential) order and be compatible with `bulk_create()`.

Solution

Add a hook to ContentUnitSaver that will pass the batch of DeclarativeVersion objects to a method called `save_related(batch)`. The base implementation of this will do nothing, but subclasses can perform `bulk_save` calls there. The call to `save_related` occurs after the content units have been saved, but during the same transaction.

Related issues:

Blocks RPM Support - Issue #3952: Using the ErrataRelatedModelSaver can resul...

CLOSED - CURRENTRELEASE

Associated revisions

Revision 1ce02a94 - 09/06/2018 10:22 PM - bmbouter

Add better support for related model saving

Adds `pre_save()` and `post_save()` handlers to ContentUnitSaver that plugin writers can override on subclasses.

<https://pulp.plan.io/issues/3953> closes #3953

Revision 1ce02a94 - 09/06/2018 10:22 PM - bmbouter

Add better support for related model saving

Adds `pre_save()` and `post_save()` handlers to ContentUnitSaver that plugin writers can override on subclasses.

<https://pulp.plan.io/issues/3953> closes #3953

History

#1 - 08/30/2018 04:37 PM - daviddavis

- Blocks Issue #3952: Using the ErrataRelatedModelSaver can result in creating incomplete content. added

#2 - 09/06/2018 04:44 PM - bmbouter

- Tracker changed from Task to Story

- Description updated

Rewriting based on discussion w/ @jortel.

#3 - 09/06/2018 05:33 PM - jortel@redhat.com

The proposed solution of calling `save_related()` after the batch has been bulk created only supports *one-to-many* related models (with FK to *Content*).

Changing this to:

```
def _create_batch(batch)
  <move batch create code block here>
```

plugin writer would override as:

```
def _create_batch(batch):
  <create models where Content has FK to>
  super()._create_batch(batch)
  <create models with FK to Content>
```

This is step toward functional decomposition of the *call* and supports both one-to-many and many-to-one.

That said, having Content with FK to other models could introduce orphan clean up problems and can't think of a good use case for it. But, thinking the API should be as flexible as possible to avoid being short sighted.

In either case the new method should be *protected* (leading `_`).

Thoughts?

#4 - 09/06/2018 06:29 PM - bmbouter

I can't think of a specific situation, but I can imagine someone wanting to do some saving within the transaction prior to the content unit being saved. I think the hooks are simpler for this specific pattern. The hooks would be:

```
class CustomContentUnitSaver(ContentUnitSaver):
```

```
    def save_before(batch):
        """
        """
```

```
    def save_safter(batch):
        """
        """
```

I think the hooks are less hazard prone because they don't have to call `super()`. Also it encourages them to separate their before/after code into distinct functions.

What do you think?

#5 - 09/06/2018 06:46 PM - jortel@redhat.com

Calling `super` when overriding a method is common (recommended) practice and I'd be surprised if it posed a *hazard*. I prefer the approach I suggested mainly because it's a pattern I'm used to seeing. But, what you're proposing in [#3953-4](#) will meet our needs as well.

#6 - 09/06/2018 07:06 PM - bmbouter

- Subject changed from *DeclarativeVersion should support creating content with related models.* to *ContentUnitSaver should support creating content with related models inside its transaction*
- Assignee set to *bmbouter*
- Groomed changed from *No* to *Yes*
- Sprint set to *Sprint 42*

OK sweet. I'm going to make this now so that we can resolve this Errata bug. I'll take that assigned also.

Grooming per the comment 5.

#7 - 09/06/2018 10:01 PM - bmbouter

- Status changed from *NEW* to *POST*

PR here: <https://github.com/pulp/pulp/pull/3621>

#8 - 09/06/2018 10:52 PM - bmbouter

- Status changed from *POST* to *MODIFIED*
- % Done changed from *0* to *100*

Applied in changeset [pulp1ce02a942c577be51a6e803a57eba73aed70e4fa](#).

#9 - 04/25/2019 06:45 PM - daviddavis

- *Sprint/Milestone set to 3.0.0*

#10 - 04/26/2019 10:34 PM - bmbouter

- *Tags deleted (Pulp 3)*

#11 - 12/13/2019 06:10 PM - bmbouter

- *Status changed from MODIFIED to CLOSED - CURRENTRELEASE*