

Python Support - Story #1882

Rebuild model to support all package types

04/29/2016 11:50 PM - amacdona@redhat.com

Status:	CLOSED - CURRENTRELEASE	Start date:	
Priority:	Normal	Due date:	
Assignee:	amacdona@redhat.com	% Done:	100%
Category:		Estimated time:	0:00 hour
Sprint/Milestone:		Tags:	Pulp 2
Platform Release:	2.13.0	Sprint:	
Target Release - Python:	2.0.0	Quarter:	
Groomed:	No		
Sprint Candidate:	Yes		

Description

The python model is currently built from the PKG-INFO file. It's unit key is (name, version) which is a problem for wheels because there can be many files for each name and version. Filename is a better key because name, version, platform, pyversion, and filetype are all present in a predictable, collision proof way defined in <https://www.python.org/dev/peps/pep-0427/> for wheels. The other formats should only have one file each (double check this).

The new model must:

- be able to be instantiated (and later saved to support lazy) from the JSON
- be able to update fields that are unavailable from JSON which can be populated from format (except .exe) using twine at download_success time
- can be instantiated from a package metadata (necessary for upload).

Additionally, it is important to consider that there are multiple versions of python metadata currently supported on pypi (1.0, 1.1, and 2.0). I have written a script to pull down packages and list the metadata keys (including the nested keys for each file) for each metadata version, the results are below.

To complete this task:

- Determine which fields are necessary for sync
- determine which fields may not play nice with all package types
- implement the model
- implement instantiation from a package
- add twine to requirements

```
{  u'1.0': set([  u'author',
                u'author_email',
                u'classifiers',
                u'comment',
                u'description',
                u'download_url',
                u'filetype',
                u'home_page',
                u'keywords',
                u'license',
                u'maintainer',
                u'maintainer_email',
                u'md5_digest',
                u'metadata_version',
                u'name',
                u'obsoletes',
                u'obsoletes_dist',
                u'platform',
                u'project_urls',
                u'provides',
                u'provides_dist',
```

```
u'pyversion',
u'requires',
u'requires_dist',
u'requires_external',
u'requires_python',
u'summary',
u'supported_platform',
u'version']],
u'1.1': set([
u'author',
u'author_email',
u'classifiers',
u'comment',
u'description',
u'download_url',
u'filetype',
u'home_page',
u'keywords',
u'license',
u'maintainer',
u'maintainer_email',
u'md5_digest',
u'metadata_version',
u'name',
u'obsoletes',
u'obsoletes_dist',
u'platform',
u'project_urls',
u'provides',
u'provides_dist',
u'pyversion',
u'requires',
u'requires_dist',
u'requires_external',
u'requires_python',
u'summary',
u'supported_platform',
u'version']],
u'2.0': set([
u'author',
u'author_email',
u'classifiers',
u'comment',
u'description',
u'download_url',
u'filetype',
u'home_page',
u'keywords',
u'license',
u'maintainer',
u'maintainer_email',
u'md5_digest',
u'metadata_version',
u'name',
u'obsoletes',
u'obsoletes_dist',
u'platform',
u'project_urls',
u'provides',
u'provides_dist',
u'pyversion',
u'requires',
u'requires_dist',
u'requires_external',
u'requires_python',
u'summary',
u'supported_platform',
u'version']],
'json': set([
u'_pypi_hidden',
```

```
u'_pypi_ordering',
u'author',
u'author_email',
u'bugtrack_url',
u'cheesecake_code_kwalitee_id',
u'cheesecake_documentation_id',
u'cheesecake_installability_id',
u'classifiers',
u'comment_text',
u'description',
u'docs_url',
u'download_url',
u'downloads',
u'filename',
u'has_sig',
u'home_page',
u'keywords',
u'license',
u'maintainer',
u'maintainer_email',
u'md5_digest',
u'name',
u'package_url',
u'packagetype',
u'path',
u'platform',
u'python_version',
u'release_url',
u'requires_dist',
u'requires_python',
u'size',
u'summary',
u'upload_time',
u'url',
u'version']}]}
```

Related issues:

Blocks Python Support - Story #1883: As a user, I can sync and publish all pa...

CLOSED - CURRENTRELEASE

Blocks Python Support - Story #1884: As a user, I can lazily sync python pack...

MODIFIED

Associated revisions

Revision 54687513 - 05/13/2016 04:15 PM - Austin Macdonald

Create new model for python packages

This commit does 4 things: 1. Rename `_filename` to `filename`. 2. Switch unit key of `Package` to `filename`. 3. Remove unnecessary fields from the model. 4. Add the machinery to create instances of new model from the PyPI JSON metadata and directly from the packages.

The `filename` contains the name, version, platform, and python versions. This means that `filename` is guaranteed to be unique on PyPI. This is necessary because name and version will no longer guarantee uniqueness when we are supporting multiple package types. For instance, two separate packages, a wheel and and `sdist` can and would share the same name and version.

There are two reasons why I chose to drop the extra fields. Firstly, the expected use case will be that users will specify which packages they want to install from a pulp python repository, they will not need to search the repo to figure out which packages to install. This means that the fields on the model that are not necessary for sync or publish are not very useful. The second problem is that if we are to implement lazy sync for python, we need to be able to create the units from the metadata available on PyPI. This metadata combines some fields like `license` and `home_page` for all versions, platforms, and package types of a given package name. This could present an issue where some of this information changed over time, but the newest version of that data is applied to all units. Since it is not really necessary in the first place, it is preferred to leave the information out than to possibly present inaccurate information.

The field `summary`, though not strictly necessary, provide helpful information and was left for convinence, despite the possibility that it will change over time.

closes #1882

mor minimal model

mor docs

Revision 2b951967 - 10/07/2016 07:37 PM - Austin Macdonald

Create new model for python packages

This commit does 4 things: 1. Rename `_filename` to `filename`. 2. Switch unit key of `Package` to `filename`. 3. Remove unnecessary fields from the model. 4. Add the machinery to create instances of new model from the PyPI JSON metadata and directly from the packages.

The filename contains the name, version, platform, and python versions. This means that filename is guaranteed to be unique on PyPI. This is necessary because name and version will no longer guarantee uniqueness when we are supporting multiple package types. For instance, two separate packages, a wheel and a sdist can and would share the same name and version.

There are two reasons why I chose to drop the extra fields. Firstly, the expected use case will be that users will specify which packages they want to install from a pulp python repository, they will not need to search the repo to figure out which packages to install. This means that the fields on the model that are not necessary for sync or publish are not very useful. The second problem is that if we are to implement lazy sync for python, we need to be able to create the units from the metadata available on PyPI. This metadata combines some fields like license and home_page for all versions, platforms, and package types of a given package name. This could present an issue where some of this information changed over time, but the newest version of that data is applied to all units. Since it is not really necessary in the first place, it is preferred to leave the information out than to possibly present inaccurate information.

The field summary, though not strictly necessary, provide helpful information and was left for convenience, despite the possibility that it will change over time.

closes #1882

mor minimal model

mor docs

History

#1 - 04/29/2016 11:52 PM - amacdona@redhat.com

- Blocked by Story #1883: As a user, I can sync and publish all package types added

#2 - 04/29/2016 11:52 PM - amacdona@redhat.com

- Blocked by deleted (Story #1883: As a user, I can sync and publish all package types)

#3 - 04/29/2016 11:52 PM - amacdona@redhat.com

- Blocks Story #1883: As a user, I can sync and publish all package types added

#4 - 04/29/2016 11:55 PM - amacdona@redhat.com

- Blocks Story #1884: As a user, I can lazily sync python packages added

#5 - 05/03/2016 10:46 PM - mhrivnak

Did this come from breaking up #135 into smaller pieces? Or is this work that has not previously been put on a sprint? One way or another, we need this to go through the sprint planning process and get it accepted onto sprint 2.

#6 - 05/06/2016 06:31 PM - amacdona@redhat.com

- Tracker changed from Task to Story

- Sprint Candidate changed from No to Yes

This is the result of the reorganization of python stories.

#7 - 05/13/2016 04:18 PM - Anonymous

- Status changed from ASSIGNED to MODIFIED

- % Done changed from 0 to 100

Applied in changeset [54687513a426a1bdec567452a52ff103555f7efe](#).

#8 - 05/27/2016 08:51 PM - amacdona@redhat.com

- Platform Release set to 2.9.0

#9 - 05/27/2016 09:51 PM - amacdona@redhat.com

- Target Release - Python set to 2.0.0

#10 - 06/15/2016 07:30 PM - semyers

- Platform Release deleted (2.9.0)

#11 - 06/21/2016 09:23 PM - semyers

- Status changed from MODIFIED to POST

#12 - 10/10/2016 05:57 PM - Anonymous

- Status changed from POST to MODIFIED

Applied in changeset [2b9519671e44ebf36fdb5eb1ec53fd98fc37700](#).

#13 - 10/25/2016 06:41 PM - semyers

- Platform Release set to 2.12.0

#14 - 01/18/2017 12:42 AM - semyers

- Status changed from MODIFIED to 5

#15 - 02/01/2017 09:30 PM - semyers

- Status changed from 5 to MODIFIED

- Platform Release deleted (2.12.0)

This issue has been removed from the 2.12.0 release, and returned to its MODIFIED state for inclusion in a future release of Pulp.

#16 - 03/10/2017 05:05 PM - semyers

- Status changed from MODIFIED to POST

I'm moving this back to POST to flag this bug as untestable in pulp-smash. This status is also appropriate because while the changes are merged to the 2.0-dev branch of pulp_python, that entire branch was deemed unreleasable for pulp 2.12, and the situation has not improved since then.

#17 - 03/16/2017 04:44 PM - semyers

- Status changed from POST to MODIFIED

2.0-dev is now being included in Platform 2.13 builds again, returning this to a testable state. Note that any smash tests related to this issue should not be run prior to 2.13.0; this is tracked in pulp-smash issue <https://github.com/PulpQE/pulp-smash/issues/588>.

#18 - 04/12/2017 04:43 PM - pcreech

- Platform Release set to 2.13.0

#19 - 04/17/2017 03:17 PM - pcreech

- Status changed from MODIFIED to 5

#20 - 04/27/2017 03:52 PM - pcreech

- Status changed from 5 to CLOSED - CURRENTRELEASE

#21 - 04/15/2019 10:31 PM - bmbouter

- Tags Pulp 2 added

Files

keys_tester.py	3.37 KB	04/29/2016	amacdona@redhat.com
----------------	---------	------------	---------------------