Arvados - Bug #14885

Review and merge chapmanb's ciso and conda packaging pull request

02/25/2019 08:36 PM - Eric Biagiotti

<table>
<thead>
<tr>
<th>Status:</th>
<th>Resolved</th>
<th>Start date:</th>
<th>03/01/2019</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assigned To:</td>
<td>Eric Biagiotti</td>
<td>% Done:</td>
<td>100%</td>
</tr>
<tr>
<td>Category:</td>
<td></td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Target version:</td>
<td>2019-03-13 Sprint</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Description
https://github.com/curoverse/arvados/pull/86

Subtasks:
Task # 14905: Review 14885-ciso-and-conda-packaging-pr

Associated revisions

Revision d0584563 - 02/27/2019 07:24 PM - Eric Biagiotti
Merge branch 'master' into 14885-ciso-and-conda-packaging-pr
refs #14885
Arvados-DCO-1.1-Signed-off-by: Eric Biagiotti <ebiagiotti@veritasgenetics.com>

Revision 3af38eca - 03/01/2019 03:02 PM - Eric Biagiotti
Merge branch '14885-ciso-and-conda-packaging-pr'
refs #14885
Arvados-DCO-1.1-Signed-off-by: Eric Biagiotti <ebiagiotti@veritasgenetics.com>

Revision 72c6ba53 - 03/06/2019 07:43 PM - Eric Biagiotti
Merge branch '14885-ciso-and-conda-packaging-pr'
refs #14885
Arvados-DCO-1.1-Signed-off-by: Eric Biagiotti <ebiagiotti@veritasgenetics.com>

History

#1 - 02/27/2019 01:57 PM - Eric Biagiotti
- Target version changed from 2019-02-27 Sprint to 2019-03-13 Sprint

#2 - 02/27/2019 06:52 PM - Eric Biagiotti
The original PR updates the ciso8601 library to >=2.0.0 for sdk/python. Note that we also have this dependency for services/fuse and sdk/cwl which need to be updated.

#3 - 02/27/2019 07:05 PM - Eric Biagiotti
Note the ciso8601 migration guide regarding the change to parse_datetime_unaware now called parse_datetime_naive:
https://github.com/closeio/ciso8601/blob/master/CHANGELOG.md#parse_datetime_unaware-has-been-renamed
Confirm that when we are using parse_datetime_naive we are parsing datetimes with time zone info into naive datetimes, instead of just normal parsing.

#4 - 02/27/2019 07:48 PM - Eric Biagiotti
Another change which subsequently breaks a sdk/python test:
https://github.com/closeio/ciso8601/blob/master/CHANGELOG.md#other-changes

----------------------------------------
ERROR: test_get_trash_at (tests.test_collections.CollectionMethods)
----------------------------------------
Traceback (most recent call last):
File "usr/src/arvados/sdk/python/tests/test_collections.py", line 817, in test_get_trash_at

02/18/2020 1/4
5 - 02/27/2019 08:42 PM - Eric Biagiotti

Apparently our datetime strings have a 'Z' at the end, which is considered time zone info by ciso8601 according to https://pypi.org/project/ciso8601/#time-zone-information. Consequently, we would need a pytz dependency when using ciso8601 2.0 with python 2.7 according to https://github.com/closeio/ciso8601/blob/master/CHANGELOG.md#other-changes.

6 - 02/27/2019 10:12 PM - Eric Biagiotti

A few notes about different ciso8601 parsing options using the following script with Python 2.7:

```python
import ciso8601
tmp = ciso8601.parse_datetime('2111-01-01T11:11:11.111111000Z')
print(str(tmp))
```

Using parse_datetime:

- ImportError: Cannot parse a timestamp with time zone information without the pytz dependency.

Using parse_datetime_as_naive:

- Works, but you lose all time zone info i.e. 2111-01-01 11:11:11

Using parse_rfc3339:

- ImportError: Cannot parse a timestamp with time zone information without the pytz dependency.

An additional caveat is that parse_rfc3339 requires a UTC offset in the string and throws an exception if its not there. In Python 3, no pytz is required because python datetime is used under the hood instead, so both parse_datetime and parse_rfc3339 produce 2111-01-01 11:11:11.111111+00:00 when the results are converted back to strings.

The easiest approach is to create a conditional Python 2.7 pytz dependency for sdk/python, sdk/cwl, and services/fuse. I think this is reasonable. Here is a little background on why pytz isn't an explicit dependency for ciso8601 https://github.com/closeio/ciso8601#dependency-on-pytz-python-2.

If we want to tighten our adherence to rfc3339, we could also switch to using parse_rfc3339; but note that this also requires a conditional Python 2.7 dependency.

7 - 03/01/2019 01:58 PM - Eric Biagiotti

Latest at d9e5d620d8e68b349bb79b9749a74110603a89d4

- Changed usage of parse_datetime_as_naive (formerly parse_datetime_unaware) to parse_datetime
- Added conditional pytz dependency for Python 2.7 for sdk/python, sdk/cwl, and services/fuse.

Unit tests: https://ci.curoverse.com/view/Developer/job/developer-run-tests/1094/

8 - 03/01/2019 02:53 PM - Peter Amstutz

Eric Biagiotti wrote:

Latest at d9e5d620d8e68b349bb79b9749a74110603a89d4

- Changed usage of parse_datetime_as_naive (formerly parse_datetime_unaware) to parse_datetime
- Added conditional pytz dependency for Python 2.7 for sdk/python, sdk/cwl, and services/fuse.

Unit tests: https://ci.curoverse.com/view/Developer/job/developer-run-tests/1094/

This LGTM

9 - 03/01/2019 03:05 PM - Eric Biagiotti

- Status changed from In Progress to Resolved

10 - 03/01/2019 07:34 PM - Tom Morris
- **Release set to 15**

## #11 - 03/01/2019 08:09 PM - Peter Amstutz

- **Status changed from Resolved to Feedback**

00:02:07.040 + ./run-diagnostics.sh

00:02:09.392 2019-03-01 20:05:51 cwltool INFO: /usr/bin/arvados-cwl-runner 1.3.1.20190301150258, arvados-python-client 1.3.1.20190301150258, cwltool 1.0.20181217162649


00:02:11.034 2019-03-01 20:05:52 cwltool ERROR: Workflow error, try again with --debug for more information:

can't compare offset-naive and offset-aware datetimes

Build step 'Execute shell' marked build as failure

## #12 - 03/05/2019 07:19 PM - Eric Biagiotti

Peter Amstutz wrote:

> [...] 

Reverted back to parse_datetime_as_naive in 2e4ec565fe74e6de19fe4d2a8670fd319544ca73

Most of our python code uses datetime.utcnow(), which is time zone naive. In order to be compatible with the removal of as_naive in ffe38539bd86a224abbb7590480c0f1deb319124 we would need to update all these instances and introduce the pytz dependency for all versions of python. This seems unnecessary considering the original scope of this story and the fact that we don't currently have other problems related to datetimes.


## #13 - 03/06/2019 07:40 PM - Peter Amstutz

Eric Biagiotti wrote:

> [...] 

Reverted back to parse_datetime_as_naive in 2e4ec565fe74e6de19fe4d2a8670fd319544ca73

Most of our python code uses datetime.utcnow(), which is time zone naive. In order to be compatible with the removal of as_naive in ffe38539bd86a224abbb7590480c0f1deb319124 we would need to update all these instances and introduce the pytz dependency for all versions of python. This seems unnecessary considering the original scope of this story and the fact that we don't currently have other problems related to datetimes.


This LGTM

Eric Biagiotti wrote:

> [...] 

Reverted back to parse_datetime_as_naive in 2e4ec565fe74e6de19fe4d2a8670fd319544ca73

Most of our python code uses datetime.utcnow(), which is time zone naive. In order to be compatible with the removal of as_naive in ffe38539bd86a224abbb7590480c0f1deb319124 we would need to update all these instances and introduce the pytz dependency for all versions of python. This seems unnecessary considering the original scope of this story and the fact that we don't currently have other problems related to datetimes.


02/18/2020 3/4

Rerun of fuse tests: https://ci.curoverse.com/job/developer-run-tests-services-fuse/1118/

LGTM

#14 - 03/06/2019 07:45 PM - Eric Biagiotti
- Status changed from Feedback to Resolved