TextCritical.net - Bug #435

Perseus importer memory exhaustion

11/12/2012 05:50 PM - Luke Murphey

Status:	Closed	**Start date:**	
Priority:	Urgent	**Due date:**	
Assignee:	Luke Murphey	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	0.1		

Description

A number of issues have been observed related to excessive memory use on the importer:

- The importer uses large amounts of memory which slows down importation because the disk begins swapping; the CPUs cannot get sufficient IO to keep them fully allocated
- The importer provides conflicting numbers regarding how many works were imported
- The importer does not consistently import the same number of works

Associated revisions

Revision 138 - 11/14/2012 03:30 AM - Luke Murphey

Added a database router that allows the use of prebuilt database for the library so that the works do not have to be built on the local machine. Reference #435.

Revision 138 - 11/14/2012 03:30 AM - Luke Murphey

Added a database router that allows the use of prebuilt database for the library so that the works do not have to be built on the local machine. Reference #435.

Revision 132 - 11/14/2012 03:30 AM - Luke Murphey

Added a database router that allows the use of prebuilt database for the library so that the works do not have to be built on the local machine. Reference #435.

Revision 139 - 11/14/2012 07:11 AM - Luke Murphey

Modified the database router such that it allows syncdb on the database but only for the reader models. Closes #435.

Revision 139 - 11/14/2012 07:11 AM - Luke Murphey

Modified the database router such that it allows syncdb on the database but only for the reader models. Closes #435.

Revision 133 - 11/14/2012 07:11 AM - Luke Murphey

Modified the database router such that it allows syncdb on the database but only for the reader models. Closes #435.

History

#1 - 11/13/2012 03:24 AM - Luke Murphey

Running the perseus_index command involves many of the same operations as the import operation except without the database operations. Running this ought to help tell what part of the code if failing.

The index operation will be executed in order to determine:

- If the memory increase is observed
- If the results are deterministic

#2 - 11/13/2012 04:52 AM - Luke Murphey

It is looking like minidom is the problem. It uses lots of memory and causes an exhausted memory scenario. See http://blog.behnel.de/index.php?p=197 for info. Here are the options:

1. Use another XML parsing library like ElementTree
2. Use multi-processing to perform the imports in separate processes (circumvents memory leaks)
3. Build the database on a high memory system and convert to another format (like JSON) for import

#3 - 11/13/2012 04:52 AM - Luke Murphey

- Status changed from New to In Progress

#4 - 11/13/2012 06:47 AM - Luke Murphey

I added the ability to do filtering based on filename using a regex. This is useful because we can do this check before actually importing the work. However, in this case, we still need to parse the work unless all of the works use a filename filter. I may need just to pull out the works and put them in a separate directory for now so that we can avoid the work of sifting through works that we don't care about.

#5 - 11/13/2012 07:35 AM - Luke Murphey

I ran the import on box with an i7 and 8 GB of memory and the import went surprisingly fast. 235 works were imported and 12 could not be due to import errors of some sort.

However, only 212 works are listed on the list of works. This is the same number listed on the Hybrid box. This is using the default policy which does not import all Greek works.

#6 - 11/13/2012 07:43 AM - Luke Murphey

The following Splunk searches are useful for getting stats about the importation process:

List of imported works

```
sourcetype="django_app" ("Successfully imported work") | table title
```

List works imported or being analyzed for import

```
sourcetype="django_app" ("Analyzing" OR "Successfully imported work")
```

#7 - 11/13/2012 07:53 AM - Luke Murphey

I imported twice on the i7 host and it imported 235 works both times. Now trying against all Greek works.

#8 - 11/13/2012 05:44 PM - Luke Murphey

Now up to 308 works imported. Extra memory seems to be the key to doing the imports. However, the logs indicate that 339 works were imported (with 70 skipped due to errors):

```
[INFO] reader.importer.PerseusBatchImporter: Import complete, files_imported=339, import_errors=70, duration=1
799
```

The import took 29 minutes.

#9 - 11/13/2012 05:44 PM - Luke Murphey

- % Done changed from 0 to 50

#10 - 11/13/2012 09:20 PM - Luke Murphey

We could use database routing to allow shipping a database with the works already included. See:

- http://stackoverflow.com/questions/8054195/django-multi-database-routing
- https://docs.djangoproject.com/en/dev/topics/db/multi-db/

#11 - 11/14/2012 03:52 AM - Luke Murphey

- % Done changed from 50 to 80

#12 - 11/14/2012 05:57 AM - Luke Murphey

The latest import indicates that it imported 339 works with 70 failures. However, it ended up with 372 works (!).

53 works could not be imported because they had a division type that was none:

```
reader.importer.PerseusBatchImporter: Exception generated when attempting to process file="52_gk.xml"
Traceback (most recent call last):
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py",
line 326, in process_directory
    if self.__process_file__( os.path.join( root, f) ):
... 7 lines omitted ...
    return func(*args, **kwargs)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 582, in
import_xml_document
    divisions = self.import_body_sub_node(body_node, current_state_set)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 968, in
import_body_sub_node
    if state.name.lower() == division_type.lower() and state.level is not None:
AttributeError: 'NoneType' object has no attribute 'lower'
```

The above errors can be found with:

```
sourcetype="django_app" ERROR "object has no attribute 'lower'"
```

This includes the following works:

```
1    52_gk.xml
2    51_gk.xml
3    50_gk.xml
4    49_gk.xml
5    48_gk.xml
6    47_gk.xml
7    46_gk.xml
8    45_gk.xml
9    44_gk.xml
10    43_gk.xml
11    42_gk.xml
12    41_gk.xml
13    40_gk.xml
14    39_gk.xml
15    38_gk.xml
16    37_gk.xml
17    36_gk.xml
18    35_gk.xml
19    34_gk.xml
20    33_gk.xml
21    32_gk.xml
22    31_gk.xml
23    30_gk.xml
24    29_gk.xml
25    28_gk.xml
26    27_gk.xml
27    26_gk.xml
28    25_gk.xml
29    24_gk.xml
30    23_gk.xml
31    22_gk.xml
32    21_gk.xml
33    20_gk.xml
34    19_gk.xml
35    18_gk.xml
36    17_gk.xml
37    16_gk.xml
38    15_gk.xml
39    14_gk.xml
40    13_gk.xml
41    11_gk.xml
42    10_gk.xml
43    09_gk.xml
44    08_gk.xml
45    06_gk.xml
46    05_gk.xml
47    04_gk.xml
48    03_gk.xml
49    02_gk.xml
50    2_gk.xml
51    nt_gk.xml
52    ath15_gk.xml
53    01_gk.xml
```

5 could not be imported because they had no biblStruct:

```
reader.importer.PerseusBatchImporter: Exception generated when attempting to process file="hh_gk.xml" 
Traceback (most recent call last):
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", 
line 326, in process_directory
    if self.__process_file__( os.path.join( root, f) ):
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", 
line 277, in __process_file__
... 5 lines omitted ...
  File "/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/site-packages/django/db/transaction.py
", line 209, in inner
    return func(*args, **kwargs)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 536, in 
import_xml_document
    bibl_struct_node = tei_header.getElementsByTagName("biblStruct")[0]
IndexError: list index out of range
```

This includes the following files:

```
1    hh_gk.xml
2    aristot.vir_gk.xml
3    aristot.nic.eth_gk.xml
4    aristot.ath.pol_gk.xml
5    apollod_gk.xml
```

11 could not be imported because they had no state set:

```
reader.importer.PerseusBatchImporter: Exception generated when attempting to process file="aristoph.wasps_gk.xml"
Traceback (most recent call last):
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", line 326, in process_directory
    if self.__process_file__( os.path.join( root, f) ):
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", line 277, in __process_file__
    return self.process_file(file_path, document_xml, title, author, language)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", line 453, in process_file
    perseus_importer.import_file(file_path)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 139, in import_file
    return self.import_xml_document(doc)
  File "/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/site-packages/django/db/transaction.py", line 209, in inner
    return func(*args, **kwargs)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 571, in import_xml_document
    current_state_set = state_sets[self.state_set]
IndexError: list index out of range
```

Which includes:

```
1    aristoph.wasps_gk.xml
2    aristoph.thes_gk.xml
3    aristoph.pl_gk.xml
4    aristoph.peace_gk.xml
5    aristoph.lys_gk.xml
6    aristoph.kn_gk.xml
7    aristoph.frogs_gk.xml
8    aristoph.eccl_gk.xml
9    aristoph.cl_gk.xml
10    aristoph.birds_gk.xml
11    aristoph.ach_gk.xml
```

One failed (dh.hist04_gk.xml) because a verse didn't have an indicator:

```
reader.importer.PerseusBatchImporter: Exception generated when attempting to process file="dh.hist04_gk.xml"
Traceback (most recent call last):
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", line 326, in process_directory
    if self.__process_file__( os.path.join( root, f) ):
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", line 277, in __process_file__
    return self.process_file(file_path, document_xml, title, author, language)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/PerseusBatchImporter.py", line 449, in process_file
    perseus_importer.import_file(file_path)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 139, in import_file
    return self.import_xml_document(doc)
  File "/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/site-packages/django/db/transaction.py", line 209, in inner
    return func(*args, **kwargs)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 586, in import_xml_document
    verses_created = self.make_verses(divisions, current_state_set)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 604, in make_verses
    verses_created = verses_created + self.make_verses_for_division(division, state_set)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 627, in make_verses_for_division
    return self.import_verse_content( division, root_node, state_set)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 822, in import_verse_content
    verses_created_temp, created_verse_node = self.import_verse_content(division, node, state_set, import_context, parent_node=next_level_node, recurse=True)
  File "/Users/lmurphey/Documents/SP/Workspace/TextCritical.com/src/reader/importer/Perseus.py", line 778, in import_verse_content
    import_context.verse.indicator = node.attributes["n"].value
  File "/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/xml/dom/minidom.py", line 524, in __getitem__
    return self._attrs[attname_or_tuple]
```

#13 - 11/14/2012 06:01 AM - Luke Murphey

The error messages account for all of the missing works (53 + 17 = 70). What I cannot account for is why I got 33 more works than the final log reported. Additionally, I cannot account for why it got more works this time. The only thing I changed was the database routing.

#14 - 11/14/2012 06:14 AM - Luke Murphey

It looks like some works are getting imported that have no divisions (see "Abdicatus", file 52_gk.xml). These are getting imported despite that import_xml_document uses @transaction.commit_on_success.

#15 - 11/14/2012 06:20 AM - Luke Murphey

Generated another index, 409 Greek works were found (as expected). From what I can tell. The transactions are not being reliably rolled back.

#16 - 11/14/2012 07:10 AM - Luke Murphey

I found 27 works more when I did a database dump to CSV from the SQL file than what the interface displays and queries return.

This is looking like some sort of problem with the transactions that can be avoided by not replying on the transaction to do a roll back. I am going to focus on fixing the causes of the exceptions and hope the transaction problems disappear.

#17 - 11/14/2012 07:11 AM - Luke Murphey

- Status changed from In Progress to Closed

- % Done changed from 80 to 100

Applied in changeset r139.