

## RPM Support - Issue #5688

### Large memory consumption when syncing RHEL 7 os x86\_64

11/11/2019 03:22 PM - jsherril@redhat.com

|   |                                  |
|---|----------------------------------|
| <b>Status:</b> CLOSED - CURRENTRELEASE  | <b>Start date:</b>               |
| <b>Priority:</b> Normal   | <b>Due date:</b>                 |
| <b>Assignee:</b> fao89  | <b>Estimated time:</b> 0:00 hour |
| <b>Category:</b>  |                                  |
| <b>Sprint/Milestone:</b>  |                                  |
| <b>Severity:</b> 2. Medium  | <b>Groomed:</b> No               |
| <b>Version:</b>   | <b>Sprint Candidate:</b> No      |
| <b>Platform Release:</b>  | <b>Tags:</b> Katello             |
| <b>OS:</b>  | <b>Sprint:</b>                   |
| <b>Triaged:</b> Yes   | <b>Quarter:</b>                  |
| <b>Description</b>  |                                  |
| When syncing the following repo: <a href="https://cdn.redhat.com/content/dist/rhel/server/7/7Server/x86_64/os/">https://cdn.redhat.com/content/dist/rhel/server/7/7Server/x86_64/os/</a> (NOTE, its cert protected),<br>A larger than expected amount of memory is used, seemingly more than pulp2. |                                  |
| <b>Related issues:</b>  |                                  |
| Related to Pulp - Refactor #5701: Performance improvement in remote duplicates  | <b>CLOSED - CURRENTRELEASE</b>   |
| Copied to RPM Support - Issue #5689: Large memory consumption when publishing...  | <b>CLOSED - CURRENTRELEASE</b>   |

#### Associated revisions

##### Revision 80c75dfa - 12/05/2019 09:01 PM - Fabricio Aguiar

Improve speed and memory performance on syncing

<https://pulp.plan.io/issues/5688> ref #5688

##### Revision 3204df51 - 12/09/2019 05:56 PM - Fabricio Aguiar

Improve memory performance on syncing

<https://pulp.plan.io/issues/5688>

closes #5688

##### Revision 805d6bf0 - 12/10/2019 03:26 PM - Fabricio Aguiar

Improve memory performance on syncing

<https://pulp.plan.io/issues/5688>

closes #5688

(cherry picked from commit 3204df5106e815e15b98e5b8c861fcb94b03e79f)

##### Revision 10d85ffa - 12/11/2019 03:47 PM - Fabricio Aguiar

Improve speed and memory performance on syncing

<https://pulp.plan.io/issues/5688> ref #5688

(cherry picked from commit 80c75dfac0114127506f28ae406470e2d83043b3)

## History

---

### #1 - 11/11/2019 03:23 PM - jsherril@redhat.com

- Copied to Issue #5689: Large memory consumption when publishing RHEL 7 os x86\_64 added

### #2 - 11/13/2019 02:06 PM - fao89

- Status changed from NEW to ASSIGNED

- Assignee set to fao89

### #3 - 11/13/2019 06:01 PM - fao89

- Status changed from ASSIGNED to NEW

- Assignee deleted (fao89)

### #4 - 11/18/2019 02:16 PM - fao89

- Status changed from NEW to ASSIGNED

- Assignee set to fao89

### #5 - 11/18/2019 03:38 PM - fao89

from <https://medium.com/zendesk-engineering/hunting-for-memory-leaks-in-python-applications-6824d0518774>

at the end of `synchronize` task, after: `dv.create()`

I put:

```
from pympler import muppy, summary
all_objects = muppy.get_objects()
sum1 = summary.summarize(all_objects)
summary.print_(sum1)
```

and the result was:

| types                           | # objects | total size |
|---------------------------------|-----------|------------|
| multidict._multidict._pair_list | 1         | 594.66 TB  |
| dict                            | 28369     | 10.29 MB   |
| str                             | 73238     | 9.47 MB    |
| type                            | 4891      | 4.69 MB    |
| code                            | 23160     | 3.19 MB    |
| tuple                           | 26499     | 2.15 MB    |
| list                            | 5918      | 656.94 KB  |
| set                             | 968       | 565.75 KB  |
| weakref                         | 6492      | 507.19 KB  |
| abc.ABCMeta                     | 277       | 279.16 KB  |
| function (__init__)             | 1937      | 257.26 KB  |
| int                             | 8147      | 250.07 KB  |
| getset_descriptor               | 3431      | 241.24 KB  |
| wrapper_descriptor              | 2715      | 212.11 KB  |
| cell                            | 4049      | 189.80 KB  |

## #6 - 11/19/2019 08:43 AM - ttereshc

- Triaged changed from No to Yes

## #7 - 11/21/2019 10:32 PM - fao89

Using the following PR:

[https://github.com/pulp/pulp\\_rpm/pull/1518](https://github.com/pulp/pulp_rpm/pull/1518)

And adding RHEL 7 url to the sync performance tests:

[http://cdn.stage.redhat.com/content/dist/rhel/server/7/7Server/x86\\_64/os/](http://cdn.stage.redhat.com/content/dist/rhel/server/7/7Server/x86_64/os/)

I got these results:

```
Process: pytest - LEAKED: 45.85MB rss | 48.87MB vms in pulp_rpm/tests/performance/test_sync.py::SyncTestCase::test_centos7_on_demand
Process: pytest - LEAKED: 7.16MB rss | 4.27MB vms in pulp_rpm/tests/performance/test_sync.py::SyncTestCase::test_centos8_appstream_on_demand
Process: systemd-journald - LEAKED: 2.54MB rss | 16.13MB vms in pulp_rpm/tests/performance/test_sync.py::SyncTestCase::test_centos8_baseos_on_demand
Process: journalctl - LEAKED: 0.00MB rss | 15.54MB vms in pulp_rpm/tests/performance/test_sync.py::SyncTestCase::test_centos8_baseos_on_demand
Process: pytest - LEAKED: 12.63MB rss | 12.50MB vms in pulp_rpm/tests/performance/test_sync.py::SyncTestCase::test_centos8_baseos_on_demand
Process: rg - LEAKED: 2890.87MB rss | 3311.25MB vms in pulp_rpm/tests/performance/test_sync.py::SyncTestCase::test_rhel7_on_demand
```

### Logs:

```
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: Traceback (most recent call last):
[5/1130]
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/usr/local/lib/pulp/lib64/python3.7/site-packages/rq/worker.py", line 822, in perform_job
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: rv = job.perform()
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/usr/local/lib/pulp/lib64/python3.7/site-packages/rq/job.py", line 605, in perform
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: self._result = self._execute()
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/usr/local/lib/pulp/lib64/python3.7/site-packages/rq/job.py", line 611, in _execute
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: return self.func(*self.args, **self.kwargs)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/home/vagrant/devel/pulp_rpm/pulp_rpm/app/tasks/synchronizing.py", line 150, in synchronize
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: dv.create()
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/home/vagrant/devel/pulpcore/pulpcore/plugin/stages/declarative_version.py", line 149, in create
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: loop.run_until_complete(pipeline)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/usr/lib64/python3.7/asyncio/base_events.py", line 584, in run_until_complete
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: return future.result()
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/home/vagrant/devel/pulpcore/pulpcore/plugin/stages/api.py", line 209, in create_pipeline
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: await asyncio.gather(*futures)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/home/vagrant/devel/pulpcore/pulpcore/plugin/stages/api.py", line 43, in __call__
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: await self.run()
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/home/vagrant/devel/pulpcore/pulpcore/plugin/stages/content_stages.py", line 101, in run
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: d_content.content.save()
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/home/vagrant/devel/pulpcore/pulpcore/app/models/base.py", line 108, in save
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: return super().save(*args, **kwargs)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/usr/local/lib/pulp/lib64/python3.7/site-packages/django/db/models/base.py", line 741, in save
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: force_update=force_update, update_fields=update_fields)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/usr/local/lib/pulp/lib64/python3.7/site-packages/django/db/models/base.py", line 779, in save_base
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: force_update, using, update_fields,
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]: File "/usr/local/lib/pulp/lib64/python3.7/site-packages/django/db/models/base.py", line 870, in _save_table
```

```
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      result = self._do_insert(cls._base_
manager, using, fields, update_pk, raw)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/models/base.py", line 908, in _do_insert
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          using=using, raw=raw)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/models/manager.py", line 82, in manager_method
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          return getattr(self.get_queryset(),
name)(*args, **kwargs)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/models/query.py", line 1186, in _insert
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          return query.get_compiler(using=usi
ng).execute_sql(return_id)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/models/sql/compiler.py", line 1335, in execute_sql
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          cursor.execute(sql, params)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/backends/utils.py", line 67, in execute
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          return self._execute_with_wrappers(
sql, params, many=False, executor=self._execute)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/backends/utils.py", line 76, in _execute_with_wrapp
ers
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          return executor(sql, params, many,
context)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/backends/utils.py", line 84, in _execute
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          return self.cursor.execute(sql, par
ams)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/utils.py", line 89, in __exit__
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          raise dj_exc_value.with_traceback(t
raceback) from exc_value
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      File "/usr/local/lib/pulp/lib64/pytho
n3.7/site-packages/django/db/backends/utils.py", line 84, in _execute
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:          return self.cursor.execute(sql, par
ams)
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      django.db.utils.OperationalError: out o
f memory
Nov 21 21:20:47 pulp3-source-fedora30.localhost.example.com rq[26493]:      DETAIL: Failed on request of size 5242
88 in memory context "ErrorContext".
```

#8 - 11/22/2019 03:40 PM - fao89

- File profile.svg added

#9 - 11/22/2019 03:42 PM - fao89

- File deleted (profile.svg)

#10 - 11/22/2019 03:42 PM - fao89

- File profile.svg added

Profile by <https://github.com/benfred/py-spy>

#11 - 11/22/2019 08:01 PM - fao89

From tracemalloc <https://docs.python.org/3/library/tracemalloc.html> :

```
#1: sql/compiler.py:1512: 6468677.4 KiB
    for rows in iter((lambda: cursor.fetchmany(itersize)), sentinel):
#2: tasks/synchronizing.py:154: 3375857.5 KiB
    dv.create()
#3: stages/declarative_version.py:149: 3375786.0 KiB
    loop.run_until_complete(pipeline)
#4: models/repository.py:631: 3375301.1 KiB
    self.repository.finalize_new_version(self)
#5: models/repository.py:96: 3375297.6 KiB
    remove_duplicates(new_version)
#6: rq/job.py:611: 3373787.9 KiB
    return self.func(*self.args, **self.kwargs)
#7: rq/job.py:605: 3371726.6 KiB
    self._result = self._execute()
#8: rq/worker.py:822: 3367619.0 KiB
    rv = job.perform()
#9: tasking/worker.py:100: 3367611.6 KiB
    return super().perform_job(job, queue)
#10: rq/worker.py:684: 3367586.3 KiB
    self.perform_job(job, queue)
#11: rq/worker.py:610: 3365505.0 KiB
    self.main_work_horse(job, queue)
#12: plugin/repo_version_utils.py:31: 3342127.9 KiB
    detail_item = item.cast()
#13: models/base.py:124: 3342126.1 KiB
    return getattr(self, rel.name).cast()
#14: fields/related_descriptors.py:401: 3342099.1 KiB
    rel_obj = self.get_queryset(instance=instance).get(**filter_args)
#15: models/query.py:1242: 3337243.6 KiB
    self._result_cache = list(self._iterable_class(self))
#16: models/query.py:402: 3311326.4 KiB
    num = len(clone)
#17: models/query.py:256: 3311326.4 KiB
    self._fetch_all()
#18: models/query.py:55: 3272178.9 KiB
    results = compiler.execute_sql(chunked_fetch=self.chunked_fetch, chunk_size=self.chunk_size)
#19: sql/compiler.py:1133: 3240648.8 KiB
    return list(result)
#20: db/utils.py:96: 3234343.1 KiB
    return func(*args, **kwargs)
#21: psycopg2/_json.py:166: 3162792.2 KiB
    return loads(s)
#22: json/decoder.py:337: 3162791.1 KiB
    obj, end = self.raw_decode(s, idx=_w(s, 0).end())
#23: json/__init__.py:348: 3162791.1 KiB
    return _default_decoder.decode(s)
#24: json/decoder.py:353: 3162790.6 KiB
    obj, end = self.scan_once(s, idx)
#25: rq/worker.py:670: 200616.5 KiB
    self.fork_work_horse(job, queue)
#26: tasking/worker.py:69: 187682.8 KiB
    super().execute_job(*args, **kwargs)
#27: rq/worker.py:477: 181088.2 KiB
    self.execute_job(job, queue)
#28: click/core.py:555: 176709.1 KiB
```

```

    return callback(*args, **kwargs)
#29: cli/cli.py:252: 176243.3 KiB
    worker.work(burst=burst, logging_level=logging_level, date_format=date_format, log_format=log_format, max_
jobs=max_jobs)
#30: cli/cli.py:78: 99395.9 KiB
    return ctx.invoke(func, cli_config, *args[1:], **kwargs)
#31: models/query.py:73: 46230.7 KiB
    obj = model_cls.from_db(db, init_list, row[model_fields_start:model_fields_end])
#32: models/base.py:513: 46230.7 KiB
    new = cls(*values)
#33: sql/compiler.py:405: 35924.2 KiB
    sql, params = node.as_sql(self, self.connection)
#34: sql/compiler.py:1087: 31590.7 KiB
    sql, params = self.as_sql()
#35: click/core.py:956: 28400.9 KiB
    return ctx.invoke(self.callback, **ctx.params)
#36: click/core.py:1137: 28308.0 KiB
    return _process_result(sub_ctx.command.invoke(sub_ctx))
#37: models/base.py:430: 27427.5 KiB
    _setattr(self, field.attname, val)
#38: models/query.py:274: 25935.6 KiB
    self._fetch_all()
#39: plugin/repo_version_utils.py:30: 25832.8 KiB
    for item in repository_version.added():
#40: models/query.py:892: 22539.7 KiB
    return self._filter_or_exclude(False, *args, **kwargs)
#41: models/query.py:910: 22523.3 KiB
    clone.query.add_q(Q(*args, **kwargs))
#42: models/query.py:399: 22480.0 KiB
    clone = self.filter(*args, **kwargs)
#43: sql/query.py:1290: 16216.5 KiB
    clause, _ = self._add_q(q_object, self.used_aliases)
#44: sql/where.py:81: 14659.8 KiB
    sql, params = compiler.compile(child)
#45: sql/compiler.py:489: 14610.8 KiB
    where, w_params = self.compile(self.where) if self.where is not None else ("", [])
#46: models/lookups.py:162: 14556.8 KiB
    lhs_sql, params = self.process_lhs(compiler, connection)
#47: psycopg2/extras.py:678: 14543.6 KiB
    lambda data, cursor: data and uuid.UUID(data) or None)
#48: sql/query.py:796: 12610.6 KiB
    aliases = list(aliases)
#49: sql/query.py:1323: 12608.7 KiB
    needed_inner = joinpromoter.update_join_types(self)
#50: sql/compiler.py:474: 10788.3 KiB
    extra_select, order_by, group_by = self.pre_sql_setup()
#51: sql/compiler.py:54: 10780.2 KiB
    self.setup_query()
#52: sql/compiler.py:45: 10780.2 KiB
    self.select, self.klass_info, self.annotation_col_map = self.get_select()
#53: click/core.py:717: 9453.3 KiB
    rv = self.invoke(ctx)
#54: models/query.py:72: 9309.9 KiB
    for row in compiler.results_iter(results):
#55: models/query.py:1219: 8312.5 KiB
    obj = self._clone()
#56: models/query.py:1231: 8312.5 KiB
    c = self.__class__(model=self.model, query=self.query.chain(), using=self._db, hints=self._hints)
#57: models/query.py:1072: 8295.1 KiB
    obj = self._chain()
#58: models/query.py:401: 8295.1 KiB
    clone = clone.order_by()
#59: models/lookups.py:153: 8245.9 KiB
    lhs_sql, params = super().process_lhs(compiler, connection, lhs)
#60: models/lookups.py:79: 8240.3 KiB
    lhs = lhs.resolve_expression(compiler.query)
#61: models/expressions.py:332: 8237.9 KiB
    return copy.copy(self)
#62: models/expressions.py:238: 8237.9 KiB
    c = self.copy()
#63: models/base.py:408: 7366.0 KiB
    pre_init.send(sender=cls, args=args, kwargs=kwargs)
#64: plugin/repo_version_utils.py:37: 7106.5 KiB
    item_query = Q(**unit_q_dict) & ~Q(pk=detail_item.pk)
#65: click/core.py:764: 6697.3 KiB

```

```

    return self.main(*args, **kwargs)
#66: sql/compiler.py:254: 6598.9 KiB
    sql, params = self.compile(col, select_format=True)
#67: models/expressions.py:737: 6597.4 KiB
    return "%s.%s" % (qn(self.alias), qn(self.target.column)), []
#68: models/lookups.py:159: 6308.8 KiB
    return lhs_sql, list(params)
#69: sql/query.py:2259: 6305.8 KiB
    query.demote_joins(to_demote)
#70: sql/query.py:1293: 6305.3 KiB
    self.demote_joins(existing_inner)
#71: sql/query.py:2258: 6302.9 KiB
    query.promote_joins(to_promote)
#72: sql/query.py:763: 6302.9 KiB
    aliases = list(aliases)
#73: sql/compiler.py:1054: 6257.6 KiB
    converters = self.get_converters(fields)
#74: models/base.py:411: 5893.3 KiB
    self._state = ModelState()
#75: utils/deconstruct.py:17: 5889.3 KiB
    obj._constructor_args = (args, kwargs)
#76: fields/related.py:986: 5874.0 KiB
    return super().get_col(alias, output_field)
#77: fields/__init__.py:381: 5873.3 KiB
    return Col(alias, self, output_field)
#78: python3.7/uuid.py:204: 5738.3 KiB
    self.__dict__['int'] = int
#79: sql/compiler.py:472: 5494.5 KiB
    refcounts_before = self.query.alias_refcount.copy()
#80: sql/compiler.py:1019: 4694.4 KiB
    field_converters = expression.get_db_converters(self.connection)
#81: sql/compiler.py:1053: 4588.6 KiB
    fields = [s[0] for s in self.select[0:self.col_count]]
#82: sql/query.py:350: 4196.5 KiB
    obj = self.clone()
#83: sql/compiler.py:219: 4158.0 KiB
    cols = self.get_default_columns()
#84: sql/compiler.py:666: 4153.8 KiB
    column = field.get_col(alias)
#85: sql/query.py:309: 4150.3 KiB
    obj.alias_refcount = self.alias_refcount.copy()
#86: python3.7/copy.py:274: 4124.2 KiB
    y = func(*args)
#87: python3.7/copyreg.py:88: 4123.7 KiB
    return cls.__new__(cls, *args)
#88: python3.7/copy.py:106: 4123.5 KiB
    return _reconstruct(x, None, *rv)
#89: python3.7/copy.py:96: 4117.8 KiB
    rv = reductor(4)
#90: models/expressions.py:159: 4116.6 KiB
    state = self.__dict__.copy()
#91: models/query_utils.py:82: 3947.9 KiB
    return self._combine(other, self.AND)
#92: python3.7/uuid.py:161: 3874.2 KiB
    int = int_(hex, 16)
#93: bin/rq:10: 3873.0 KiB
    sys.exit(main())
#94: sql/query.py:1318: 3597.0 KiB
    split_subq=split_subq, simple_col=simple_col,
#95: models/expressions.py:749: 3143.1 KiB
    self.target.get_db_converters(connection))
#96: fields/__init__.py:709: 3120.8 KiB
    return []
#97: utils/tree.py:108: 2590.7 KiB
    self.children.extend(data.children)
#98: models/query_utils.py:72: 2369.2 KiB
    obj = type(self)()
#99: models/query_utils.py:85: 2369.0 KiB
    obj = type(self)()
#100: models/base.py:503: 2214.0 KiB
    post_init.send(sender=cls, instance=self)
#101: models/query.py:199: 2067.6 KiB
    self._known_related_objects = {} # {rel_field: {pk: rel_obj}}
#102: models/query.py:190: 2055.7 KiB
    self.model = model

```

```

#103: utils/tree.py:23: 1978.0 KiB
    self.children = children[:] if children else []
#104: models/query_utils.py:59: 1975.8 KiB
    super().__init__(children=[*args, *sorted(kwargs.items())], connector=_connector, negated=_negated)
#105: models/query_utils.py:74: 1801.5 KiB
    obj.add(self, conn)
#106: sql/query.py:1251: 1772.9 KiB
    condition = self.build_lookup(lookups, col, value)
#107: sql/query.py:1116: 1772.3 KiB
    lookup = lookup_class(lhs, rhs)
#108: models/lookups.py:19: 1769.0 KiB
    self.lhs, self.rhs = lhs, rhs
#109: sql/query.py:68: 1758.4 KiB
    return target.get_col(alias, field)
#110: sql/query.py:1249: 1757.4 KiB
    col = _get_col(targets[0], join_info.final_field, alias, simple_col)
#111: fields/related.py:974: 1569.4 KiB
    converters = super().get_db_converters(connection)
#112: sql/compiler.py:1018: 1563.2 KiB
    backend_converters = self.connection.ops.get_db_converters(expression)
#113: base/operations.py:564: 1563.2 KiB
    return []
#114: models/expressions.py:747: 1551.4 KiB
    return self.output_field.get_db_converters(connection)
#115: models/query.py:1893: 1481.2 KiB
    related_klass_infos = klass_info.get('related_klass_infos', [])
#116: models/query.py:63: 1481.2 KiB
    related_populators = get_related_populators(klass_info, select, db)
#117: models/query.py:62: 1445.7 KiB
    for f in select[model_fields_start:model_fields_end]]
#118: models/query.py:61: 1445.7 KiB
    init_list = [f[0].target.attname
968 other: 8.9 MiB
Total allocated size: 82418.8 MiB

```

**#12 - 11/22/2019 08:52 PM - fao89**

Here is how I got the result above:

[https://github.com/pulp/pulp\\_rpm/commit/f3f079010cfe81b7f5cf3ef94c2402b1ccf7d90c](https://github.com/pulp/pulp_rpm/commit/f3f079010cfe81b7f5cf3ef94c2402b1ccf7d90c)



After running the sync test for RHEL 7,  
I ran this script:

```
import linecache
import os
import tracemalloc

snapshots = []

traces = [t for t in os.listdir("/tmp") if t.startswith("tracemalloc-")]
traces = sorted(traces, key=lambda t: int(t.split(".")[2].split("-")[-1]))

def display_top(snapshot, key_type='lineno'):
    snapshot = snapshot.filter_traces((
        tracemalloc.Filter(False, "<frozen importlib._bootstrap>"),
        tracemalloc.Filter(False, "<unknown>"),
    ))
    top_stats = snapshot.statistics(key_type=key_type, cumulative=True)

    for index, stat in enumerate(top_stats, 1):
        frame = stat.traceback[0]
        # replace "/path/to/module/file.py" with "module/file.py"
        filename = os.sep.join(frame.filename.split(os.sep)[-2:])
        if (stat.size / 1024 / 1024) < 1:
            break
        print("#%s: %s:%s: %.1f KiB"
              % (index, filename, frame.lineno, stat.size / 1024))
        line = linecache.getline(frame.filename, frame.lineno).strip()
        if line:
            print('    %s' % line)

    other = top_stats[index:]
    if other:
        size = sum(stat.size for stat in other)
        print("%s other: %.1f MiB" % (len(other), size / 1024 / 1024))
    total = sum(stat.size for stat in top_stats)
    print("Total allocated size: %.1f MiB" % (total / 1024 / 1024))

print("\nLast Snapshot")
last_trace = "/tmp/" + traces[-1]
snapshot = tracemalloc.Snapshot.load(last_trace)
display_top(snapshot)
```

sometimes the last snapshot is corrupted because it was being saved when process got killed

**#13 - 11/26/2019 07:24 PM - fao89**

Thread on pulp-dev:

<https://www.redhat.com/archives/pulp-dev/2019-November/msg00073.html>

**#14 - 12/03/2019 04:47 PM - fao89**

- Related to Refactor #5701: Performance improvement in remote duplicates added

**#15 - 12/05/2019 08:26 PM - fao89**

- Status changed from ASSIGNED to POST

<https://github.com/pulp/pulpcore/pull/440>

[https://github.com/pulp/pulp\\_rpm/pull/1550](https://github.com/pulp/pulp_rpm/pull/1550)

**#16 - 12/09/2019 07:47 PM - Anonymous**

- Status changed from POST to MODIFIED

Applied in changeset [3204df5106e815e15b98e5b8c861fcb94b03e79f](#).

**#17 - 12/10/2019 07:33 PM - Anonymous**

Applied in changeset [805d6bf0f3ae1fd9a51bd06886d264288aee49a4](#).

**#18 - 12/13/2019 08:51 AM - ttereshc**

- Status changed from MODIFIED to CLOSED - CURRENTRELEASE

**#19 - 05/08/2020 07:44 PM - ggainey**

- Tags Katello added

- Tags deleted (Katello-P2)

## Files

---

|             |        |            |       |
|-------------|--------|------------|-------|
| profile.svg | 133 KB | 11/22/2019 | fao89 |
|-------------|--------|------------|-------|