

## Arvados - Bug #18339

### SweepTrashedObjects scaling issues

11/05/2021 06:19 PM - Peter Amstutz

<b>Status:</b>	Resolved	<b>Start date:</b>	11/16/2021
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assigned To:</b>	Tom Clegg	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	2021-11-24 sprint		
<b>Description</b>			
Customer reported problem in which a large number of projects, which had been trashed two weeks previously, became eligible for deletion all at once, and the resulting sweep processes crushed the database.			
<ol style="list-style-type: none"><li>1. If the last sweep hasn't finished, currently it starts a new, overlapping sweep operation. Instead, it should probably do something like try to take a lock (perhaps by on a empty, special-purpose table), and then if it can't acquire the lock, don't do anything (it checks the sweep task periodically, so it'll eventually run again when lock is no longer held).</li><li>2. The default value for TrashSweepInterval is 60s. Deleting trashed projects is a very low priority process, so the default interval could be much lower (eg 5m or 15m or 60m)</li><li>3. The delete operations should run in a transaction, to reduce overhead from auto-commit. Although, if it takes any locks they will be held for the duration of the transaction.</li><li>4. Destroying a group also destroys any permission links pointing to it. Ruby on Rails is running the before_destroy hook for the permission link, which does an update_permissions operation, which takes the table lock on materialized_permissions. We can set Thread.current[:suppress_update_permissions] = true to prevent permissions from being recomputed, we just need to be confident that direct updates to the permission table are correct.</li></ol>			
<b>Subtasks:</b>			
Task # 18357: Review 18339-sweep-trash-lock			<b>Resolved</b>

#### Associated revisions

##### Revision e6769d20 - 11/18/2021 08:01 PM - Tom Clegg

Merge branch '18339-sweep-trash-lock'

fixes #18339

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <[tom@curii.com](mailto:tom@curii.com)>

##### Revision 771804a8 - 11/22/2021 06:58 PM - Tom Clegg

Merge branch '18339-sweep-trash-lock'

fixes #18339

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <[tom@curii.com](mailto:tom@curii.com)>

#### History

##### #1 - 11/05/2021 06:45 PM - Peter Amstutz

- Description updated

##### #2 - 11/05/2021 06:58 PM - Peter Amstutz

- Description updated

##### #3 - 11/05/2021 07:01 PM - Peter Amstutz

- Description updated

##### #4 - 11/05/2021 07:02 PM - Peter Amstutz

- Description updated

##### #5 - 11/05/2021 07:06 PM - Peter Amstutz

- Description updated

**#6 - 11/05/2021 07:29 PM - Ward Vandewege**

- Release set to 45

**#7 - 11/10/2021 04:20 PM - Peter Amstutz**

- Assigned To set to Tom Clegg

**#8 - 11/11/2021 09:00 PM - Tom Clegg**

- Status changed from New to In Progress

- Description updated

**#9 - 11/12/2021 03:16 PM - Tom Clegg**

Maybe this is a good time to establish a better pattern for background tasks.

Propose:

- Every controller process runs a "periodic trash sweep" goroutine
- Use `pg_advisory_lock()` to ensure only one such goroutine is active even when there are multiple controller processes
- RailsAPI has an admin-only synchronous `/sys/sweep_trashed_objects` endpoint instead of a background task

One side effect is that `sweep_trashed_objects` will run even when the cluster is idle (vs. current implementation which only runs when collection lookups happen).

**#10 - 11/16/2021 04:30 PM - Tom Clegg**

18339-sweep-trash-lock @ [aa3d4030686f5db784dcaf2d7f28225eb98c4267](https://ci.arvados.org/view/Developer/job/developer-run-tests/2800/) -- <https://ci.arvados.org/view/Developer/job/developer-run-tests/2800/>

**#11 - 11/17/2021 02:34 PM - Lucas Di Pentima**

Some questions & comments:

- File `sys_controller.rb`
  - Related comment: Looking at the old code that was kept on this branch, I'm seeing at line 22 that only project groups are selected for deletion, so other non project groups that have been being trashed will live forever on the trash unless they're untrashed and deleted again after [#18340](#) is merged and released? Am I thinking this right? If so, maybe we'll need to add a cleanup migration?
  - Lines 26-29 take care of moving trash-at-the-future objects to the trash, should that still need to be tied with the trash sweeping behavior? If the site config is set to garbage collect every N hours, the auto-trashing will be affected and may not be what the users expect? (I know this is the way it worked before, but still think is worth asking, maybe for another ticket)
- Would it be interesting to expose the `/sys/sweep_trashed_objects` to the site admin so that they can make manual sweeps?

Other than that, LGTM.

**#12 - 11/17/2021 03:01 PM - Tom Clegg**

Lucas Di Pentima wrote:

[...] non project groups that have been being trashed will live forever on the trash unless they're untrashed and deleted again after [#18340](#) is merged and released? Am I thinking this right? If so, maybe we'll need to add a cleanup migration?

It seems easiest, and possibly more efficient, to remove the `where({group_class: 'project'}).conditions`. Then, pre-[#18340](#)-trashed zombie projects will be deleted without a migration.

...and add an upgrade note in case anyone is currently relying on zombie roles.

- Lines 26-29 take care of moving trash-at-the-future objects to the trash, should that still need to be tied with the trash sweeping behavior? If the site config is set to garbage collect every N hours, the auto-trashing will be affected and may not be what the users expect? (I know this is the way it worked before, but still think is worth asking, maybe for another ticket)

I think this is why the default sweep interval is 60s. The config comment does mention that auto-trash is affected, although I suppose determining the user-visible consequences of a longer interval is left as an exercise to the reader™... do we need to clarify this?

- Would it be interesting to expose the `/sys/sweep_trashed_objects` to the site admin so that they can make manual sweeps?

I suppose it's good form to add it to the discovery doc, at least.

**#13 - 11/17/2021 08:53 PM - Tom Clegg**

18339-sweep-trash-lock @ [eafbd28d0a866807471951e133a8132dbdfa9cfc](https://ci.arvados.org/view/Developer/job/developer-run-tests/2805/) -- <https://ci.arvados.org/view/Developer/job/developer-run-tests/2805/>

- delete zombie role groups
- add upgrade note
- add test cases (missed adding the file in previous commit)

**#14 - 11/17/2021 09:31 PM - Lucas Di Pentima**

One last observation:

- I think there's one missing place that's enforcing only projects to be deleted: `services/api/app/controllers/sys_controller.rb:L41`

Other than that, LGTM!

**#15 - 11/17/2021 09:38 PM - Peter Amstutz**

The upgrade notes should probably note the implications of the old behavior (role groups you thought were deleted, were not) and what the cleanup will do (delete them for real).

**#16 - 11/18/2021 06:32 PM - Tom Clegg**

18339-sweep-trash-lock @ [8a33f7899da36343e687febb63678d90e83b7d63](https://ci.arvados.org/view/Developer/job/developer-run-tests/2807/) -- <https://ci.arvados.org/view/Developer/job/developer-run-tests/2807/>

**#17 - 11/18/2021 08:16 PM - Tom Clegg**

- *Status changed from In Progress to Resolved*

Applied in changeset `arvados-private:commit:arvados|e6769d20505e2c8c74b2d7e3f9c2f33f2a2db092`.