redmineorg-copy202205 - Vote #81832

Background job and dedicated status for project deletion

2022/05/09 19:21 - Admin Redmine

ステータス:	New	**開始日:**	2022/05/09
優先度:	通常	**期日:**	
担当者:		**進捗率:**	0%
カテゴリ:	Administration_8	**予定工数:**	0.00時間
対象バージョン:	5.1.0_176	**作業時間:**	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/36691	**status_id:**	1
category_id:	8	**tracker_id:**	3
version_id:	176	**plus1:**	0
issue_org_id:	36691	**affected_version:**	
author_id:	40856	**closed_on:**	
assigned_to_id:	40856	**affected_version_id:**	
comments:	12		

説明

Due to the deletion of dependent objects (issues etc) and child projects,
project deletion may take a long time.

This patch, which was extracted from Planio,
moves the actual project deletion into an ActiveJob job. It also introduces a
new project status (SCHEDULED_FOR_DELETION) that is used to effectively hide
the project that is about to be deleted (and any potential descendant projects)
from the system immediately.

A security notification is sent out to the user who deleted the project,
informing about success / failure.

The projects list is extended to be able to filter for the new status,
so in case of a failure, the project can still be accessed for
examination.

journals

Nice feature!

Jens, should we take into consideration also #34987?

maybe my group members issue #36696 can benefit from this as well

Marius BALTEANU wrote:

> Jens, should we take into consideration also #34987?

Ah, that's interesting. If I understand correctly, that would mean the project record (plus potential child projects' records) would be deleted synchronously, but removal of all dependent objects would be pushed to the job queue.

I see few problems with that:

1. Data integrity. Before the job is finished, we have an inconsistent database (i.e. issues referencing a nonexisting project). I am not sure we handle such a situation gracefully (i.e. resulting in an HTTP 404 vs a 500) in all cases.
2. Two transactions instead of one. Currently, if the project deletion fails due to a problem somewhere along the way (due to a failing callback for example), the transaction is rolled back and everything is as if nothing happened. Splitting the process in two separate transactions (one for the project, one later for the dependent objects), means the project record will be gone even if some dependent object insists on not being destroyed. I do not think core has any such callbacks, but plugins might.
3. Does not work at all when there are foreign keys referencing projects.id. We do not have such in core, but again, there may be

plugins that do.

Thanks Jens for your quick response.

Marius, I just have re-submitted this patch as part of a larger work on the admin projects list in #33422. It would be great if you could have a look and consider integrating that one instead.

There is a random failing test, you can reproduce by running the tests using the following command: @TESTOPTS="--seed=42328" rake test@

```
Failure:
DestroyProjectJobTest#test_should_destroy_project_and_send_email [/work/test/unit/jobs/destroy_project_job_test.rb:60]:
No enqueued job found with {:job=>ActionMailer::MailDeliveryJob, :args=>#}
```

Jens, can you take a look, please?

this is really strange, looks like under some circumstances emails are processed inline and not result in a queued mailer job. I did not find the cause of this, but the attached patch would handle both cases in the tests (the same issue exists in the test case for the job that handles deletion of multiple projects).

related_issues

relates,New,31076,Issues CSV / PDF export via ActiveJob
relates,Closed,33422,Re-implement admin project list using ProjectQuery system

履歴

#1 - 2022/05/10 16:57 - Admin Redmine

- カテゴリ を Administration_8 にセット

- 対象バージョン を 5.1.0_176 にセット