

Redmine - Defect #32985

Garbage collection issue when exporting huge CSV

2020-02-11 21:16 - Ko Nagase

| | | | |
|------------------------|-------------|--------------------------|-----------|
| Status: | New | Start date: | |
| Priority: | Normal | Due date: | |
| Assignee: | | % Done: | 0% |
| Category: | Issues list | Estimated time: | 0.00 hour |
| Target version: | | Affected version: | 4.0.5 |
| Resolution: | | | |

Description

I set "10000" as "Issues export limit" (instead of default "500") for some reasons, but I noticed that exporting around 10,000 issues CSV causes out of memory issue.

I thought that the cause is inside some plugins, but it seems to be happen without any plugins, and after changing the following Issues Controller part, the memory usage seems to become stable (no memory leak).

```
diff --git a/app/controllers/issues_controller.rb b/app/controllers/issues_controller.rb
index 69a947b03..6596df30c 100644
--- a/app/controllers/issues_controller.rb
+++ b/app/controllers/issues_controller.rb
@@ -63,8 +63,8 @@ class IssuesController < ApplicationController
   render_feed(@issues, :title => "#{@project || Setting.app_title}: #{l(:label_issue_plural)}")
 }
 format.csv {
-   @issues = @query.issues(:limit => Setting.issues_export_limit.to_i)
-   send_data(query_to_csv(@issues, @query, params[:csv]), :type => 'text/csv; header=present', :filename => 'issues.csv'
+   tmp_issues = @query.issues(:limit => Setting.issues_export_limit.to_i)
+   send_data(query_to_csv(tmp_issues, @query, params[:csv]), :type => 'text/csv; header=present', :filename => '
issues.csv')
 }
 format.pdf {
   @issues = @query.issues(:limit => Setting.issues_export_limit.to_i)
```

Ruby process memory usage

| Pattern | Initial | 1st | 2nd | 3rd |
|---------------------|----------|---------|----------|----------|
| Before modification | 126.4 MB | 1.20 GB | 1.92 GB | 973.4 MB |
| After modification | 131.4 MB | 1.04GB | 776.3 MB | 803.5 MB |

I am not sure whether above modification is correct, so someone's review is quite helpful...

Here is my local reproducible environment:

- Redmine: 4.0.5
- OS: macOS Mojave
- Ruby: 2.6.5
- DB: PostgreSQL 12
- Server: Puma

Thanks,