Performance improvement when rendering news or calendar block on My page

The following patch improves the performance when retrieving objects to render especially when there are many visible projects.

Index: app/helpers/my_helper.rb

---

app/helpers/my_helper.rb (revision 17763)
+++
app/helpers/my_helper.rb (working copy)
@@ -78,7 +78,7 @@
def render_calendar_block(block, settings)
    calendar = Redmine::Helpers::Calendar.new(User.current.today, current_language, :week)
    calendar.events = Issue.visible.
    - where(:project_id => User.current.projects.pluck(:id)).
    + where(:project_id => User.current.projects).  
      where("(start_date>=? and start_date<=?) or (due_date>=? and due_date<=?)", calendar.startdt, calendar.enddt, calendar.startdt, calendar.enddt).
      includes(:project, :tracker, :priority, :assigned_to).
      references(:project, :tracker, :priority, :assigned_to).
@@ -142,7 +142,7 @@

def render_news_block(block, settings)
    news = News.visible.
    - where(:project_id => User.current.projects.pluck(:id)).
    + where(:project_id => User.current.projects).
      limit(10).
      includes(:project, :author).
      references(:project, :author).

Here is a result of the benchmark test. About 9 times faster with 1000 projects.

```ruby
require 'benchmark/ips'
include Redmine::Utils::DateCalculation

calendar = Redmine::Helpers::Calendar.new(User.current.today, :en, :week)

Benchmark.ips do |x|
  x.report("before") do
    Issue.visible.
      where(:project_id => Project.all.pluck(:id)).
      where("(start_date>=? and start_date<=?) or (due_date>=? and due_date<=?)", calendar.startdt, calendar.enddt, calendar.startdt, calendar.enddt).
      includes(:project, :tracker, :priority, :assigned_to).
```
references(:project, :tracker, :priority, :assigned_to)
end

x.report("after") do
  Issue.visible.
  where(:project => Project.all).
  where("(start_date>=? and start_date<=?) or (due_date>=? and due_date<=?)", calendar.startdt, calendar.enddt, calendar.startdt, calendar.enddt).
  includes(:project, :tracker, :priority, :assigned_to).
  references(:project, :tracker, :priority, :assigned_to)
end

x.compare!
end

})

$ bin/rails r bench.rb
Warming up -------------------------------
before  5.000  i/100ms
after   64.000  i/100ms
Calculating -------------------------------
before  68.797  (± 7.3%) i/s - 345.000 in 5.040461s
after  652.797  (± 6.0%) i/s - 3.264k in 5.018461s

Comparison:
  after: 652.8 i/s
  before: 68.8 i/s - 9.49x slower

Associated revisions
Revision 17771 - 2018-12-21 04:41 - Go MAEDA

Performance improvement when rendering news or calendar block on My page (#30249).

Patch by Go MAEDA.

History
#1 - 2018-12-19 12:22 - Pavel Rosický

lgtm

#2 - 2018-12-20 02:02 - Go MAEDA
- Target version set to 4.1.0

Pavel Rosický wrote:

lgtm
Thank you for reviewing the patch. Setting the target version to 4.1.0.

#3 - 2018-12-21 04:41 - Go MAEDA
- Status changed from New to Closed
- Assignee set to Go MAEDA

Committed.