

Scrum - To do #1508

issue_patch : Optimize update_parent_pbi_on_closed_tasks

2017-11-14 13:39 - Redmine Smile

Status:	New	Estimated time:	0.00 hour
Priority:	Normal		
Assignee:			
Blocked:	No		
Description			
break if task opened found :			
<pre>def update_parent_pbi_on_closed_tasks statuses = IssueStatus.where(:id => Scrum::Setting.closed_pbi_status_id).order("position ASC") pbi = self.parent if statuses.length == 1 and pbi and pbi.is_pbi? pbi_status_to_set = statuses.first all_tasks_closed = self.closed? pbi.children.each do task if task.is_task? task = self if task.id == self.id unless task.closed? all_tasks_closed = false break # at least a task opened, no need to go further end end end if all_tasks_closed and pbi.status != pbi_status_to_set pbi.init_journal(User.current, l(:label_pbi_status_auto_updated_all_tasks_closed, :pbi_status => pbi_status_to_set.name)) pbi.status = pbi_status_to_set pbi.save! end end end</pre>			

Associated revisions

Revision bffe26cc - 2017-12-22 14:59 - Jérôme BATAILLE

break in update_parent_pbi_on_closed_tasks, update_parent_pbi, scrum_closed?, closed_on_for_burndown
+ cache added to total_time

[#1508] issue_patch : Optimize update_parent_pbi_on_closed_tasks
modifié : lib/scrums/issue_patch.rb

Revision 8ad5166b - 2017-12-26 12:04 - Emilio González Montaña

[#1508] Refactorize cache optimizations

History

#1 - 2017-11-14 14:01 - Redmine Smile

The same with update_parent_pbi, break at first task not new

```
def update_parent_pbi
  new_status = IssueStatus.task_statuses.first
  in_progress_status = IssueStatus.task_statuses.second
  if new_status && in_progress_status
    pbi = self.parent
    if pbi and pbi.is_pbi?
      all_tasks_new = (self.status == new_status)
      pbi.children.each do |task|
        if task.is_task?
          task = self if task.id == self.id
        end
      end
    end
  end
end
```

```

        if task.status != new_status
          all_tasks_new = false
          break # at least a task not new, no need to go further
        end
      end
    end
  end
  if pbi.status == new_status and !all_tasks_new
    pbi.init_journal(User.current,
                    l(:label_pbi_status_auto_updated_one_task_no_new,
                     :pbi_status => in_progress_status.name,
                     :task_status => new_status.name))
    pbi.status = in_progress_status
    pbi.save!
  elsif pbi.status != new_status and all_tasks_new
    pbi.init_journal(User.current,
                    l(:label_pbi_status_auto_updated_all_tasks_new,
                     :pbi_status => new_status.name,
                     :task_status => new_status.name))
    pbi.status = new_status
    pbi.save!
  end
end
end
end
end
end

```

#2 - 2017-11-14 14:07 - Redmine Smile

The same for `scrum_closed?` :

```

def scrum_closed?
  closed = self.closed?
  if !closed and is_pbi? and self.children.any? and
    Scrum::Setting.pbi_is_closed_if_tasks_are_closed?
    closed = true
  self.children.each do |task|
    if !task.closed?
      closed = false
      break # at least a task opened, no need to go further
    end
  end
end
return closed
end

```

#3 - 2017-11-14 14:20 - Redmine Smile

Again for `closed_on_for_burndown` :

```

def closed_on_for_burndown
  completed_date = nil
  closed_statuses = IssueStatus::closed_status_ids
  if self.closed?
    self.journals.order('created_on DESC').each do |journal|
      if completed_date.nil?
        journal.details.where(:prop_key => 'status_id',
                              :value => closed_statuses).each do |detail|
          completed_date = journal.created_on
          break # a date found, no need to go further
        end
      end
    end
  end
  if self.is_pbi? and self.children.any? and
    Scrum::Setting.pbi_is_closed_if_tasks_are_closed
    all_tasks_closed = true
    last_closed_task_date = completed_date
    self.children.each do |task|
      if all_tasks_closed and task.closed?
        task_closed_on = task.closed_on_for_burndown
        if task_closed_on
          if last_closed_task_date.nil? or
            last_closed_task_date > task_closed_on
            last_closed_task_date = task_closed_on
          end
        end
      end
    end
  end
end

```

```
    else
      all_tasks_closed = false
    end
  end
  if all_tasks_closed and last_closed_task_date
    completed_date = last_closed_task_date
  end
end
return completed_date
end
```

#4 - 2017-12-22 15:13 - Redmine Smile

Merge Request : https://git.ociotec.com/redmine/scrum/merge_requests/2

#5 - 2018-01-03 17:58 - Redmine Smile

Emilio, I have checked your modifications on the MR, it's OK for me.