

## Scrum - Defect #1146

### :include in IssueQuery should AFAIK be an array not a hash

2016-03-22 00:11 - Christian Luginbühl

<b>Status:</b>	New		
<b>Priority:</b>	Normal		
<b>Category:</b>			
<b>Target version:</b>			
<b>Source:</b>	Customer	<b>Blocked:</b>	No
<b>Detected on version:</b>	Scrum v0.13.2	<b>Alone:</b>	

#### Description

This is a bit of a strange bug report, because I couldn't really detect the root cause of this error.

When running Redmine 3.2.1 on CentOS 7 (Ruby 2.0.0p598) with both plugins "scrum" (0.13.2) and "extended\_fields" (trunk), I get a strange HTTP 500 error when clicking on the "Calendar" of a project and the error log message:

```
I, [2016-03-21T23:55:42.635280 #15869] INFO -- : Started GET "/projects/kaas/issues/calendar" for 10.242.2.6 at 2016-03-21 23:55:42 +0100
I, [2016-03-21T23:55:42.742558 #15869] INFO -- : Processing by CalendarsController#show as HTML
I, [2016-03-21T23:55:42.742728 #15869] INFO -- : Parameters: {"project_id"=>"test"}
I, [2016-03-21T23:55:42.748361 #15869] INFO -- : Current user: admin (id=1)
I, [2016-03-21T23:55:42.866978 #15869] INFO -- : Completed 500 Internal Server Error in 124ms (ActiveRecord: 9.7ms)
F, [2016-03-21T23:55:42.868471 #15869] FATAL -- :
NoMethodError (undefined method `<<' for {}:Hash):
plugins/scrum/lib/scrum/issue_query_patch.rb:47:in `issue_ids_with_scrum'
plugins/extended_fields/lib/extended_calendars_controller_patch.rb:37:in `extended_show'
plugins/scrum/lib/scrum/calendars_controller_patch.rb:18:in `add_sprints'
lib/redmine/sudo_mode.rb:63:in `sudo_mode'
```

Funny point is, that this does not happen, when the "extended\_fields" plugin is not there.

Anyway, I looked into issue\_query\_patch.rb and found these lines

```
def issues_with_scrum(options = {})
  options[:include] ||= {}
  options[:include] << :sprint
  issues_without_scrum(options)
end
alias_method_chain :issues, :scrum

def issue_ids_with_scrum(options = {})
  options[:include] ||= {}
  options[:include] << :sprint
  issue_ids_without_scrum(options)
end
alias_method_chain :issue_ids, :scrum
```

By looking at redmine/app/models/issue\_query.rb (what the issue\_query\_patch.rb is actually patching), I see, that the functions 'issue' and 'issue\_ids' is to my understanding expecting 'options[:include]' to be an array and not a hash. Here (issue\_ids is analogue):

```
def issues(options={})
  order_option = [group_by_sort_order, options[:order]].flatten.reject(&:blank?)

  scope = Issue.visible.
    joins(:status, :project).
    where(statement).
    includes(([:status, :project] + (options[:include] || [])).uniq).
    where(options[:conditions]).
    order(order_option).
    joins(joins_for_order_statement(order_option.join(', '))).
```

```
limit(options[:limit]).  
offset(options[:offset])
```

```
# SNIP
```

So if I change `issue_query_patch.rb` to

```
def issues_with_scrum(options = {})  
  options[:include] ||= []  
  options[:include] << :sprint  
  issues_without_scrum(options)  
end  
alias_method_chain :issues, :scrum  
  
def issue_ids_with_scrum(options = {})  
  options[:include] ||= []  
  options[:include] << :sprint  
  issue_ids_without_scrum(options)  
end  
alias_method_chain :issue_ids, :scrum
```

everything seems to work as expected for me and the error disappears.

It seems strange that the hash way of defining an unknown `'options[:include]'` has been there since pretty much the beginning of this plugin and I didn't find any bug reports here directly addressing this possible problem. Also it doesn't seem either, that Redmine had changed anything drastically, so I feel a bit uncertain what to think about my experiences, but I wanted to share, what I did go get rid of this error in my installation.

Don't hesitate asking if you need further information about my system or make me do tests.