

Redmine - Defect #6082

acts_as_event email key

2010-08-09 09:24 - Felix Schäfer

Status:	Confirmed	Start date:	2010-08-09
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Activity view	Estimated time:	0.00 hour
Target version:		Affected version:	
Resolution:			
Description			
<p>When avatars are enabled, the activity view tries to show the gravatar by passing a user object to the avatar method. While this works for all events that have local users, the changeset model/event can't always create a local user object for the changeset while still having the email of the user that generated this changeset.</p> <p>Concrete use-case: I have a git repository in which I have cherry-picked 2 revisions from a user not member of my redmine installation, in this case the changeset retains the author not as a user object, but as a string "name <email>" (very much like git presents it), the changeset object only presents the "name" as the author (see source:trunk/app/models/changeset.rb#L68). While this is correct behavior for places just polling event_author for a name, the avatar won't get the email address from it, though that information is present. I've changed the activity view to try to use committer (the whole "name <email>", which avatar parses correctly) if the "normal" way doesn't yield an avatar, the following line source:trunk/app/views/projects/activity.rhtml#L10</p> <pre><%= avatar(e.event_author, :size => "24") if e.respond_to?(:event_author) %></pre> <p>to read:</p> <pre><%= (avatar(e.event_author, :size => "24") if e.respond_to?(:event_author)) or (avatar(e.committer, :size => "24") if e.respond_to?(:committer)) %></pre> <p>This solves the problem for changesets, but will not solve it for the generic case.</p> <p>I see 2 options here to deal with that: either extend acts_as_event to also provide a method/an interface to query the author's email address (but that feels somewhat kludgy), or mock a (temporary) user object with the information we have (basically the name for the to_s method and the email address for email) to provide the same interface as for "local users". This might break further along down the way though if anything polling the events expects to receive a full user object but only gets one with a very limited interface.</p>			

History

#1 - 2017-01-14 01:52 - Go MAEDA

- Category set to Activity view

#2 - 2020-12-07 07:23 - Go MAEDA

- File bugs_ruby-lang_org-activity.png added

- Status changed from New to Confirmed

I have confirmed the issue on <https://bugs.ruby-lang.org/>.

Gravatar icons for committers who don't have an account of the Redmine are not displayed on the Activity page.

bugs_ruby-lang_org-activity.png

#3 - 2020-12-22 05:34 - Yuichi HARADA

- File 6082.patch added

I created a helper method for the activity because it complicates the Redmine account and the Repository committer decisions.

```
diff --git a/app/helpers/activities_helper.rb b/app/helpers/activities_helper.rb
index 49074b5617..d48a9939e1 100644
--- a/app/helpers/activities_helper.rb
+++ b/app/helpers/activities_helper.rb
@@ -30,4 +30,12 @@ module ActivitiesHelper
  end
  sorted_events
end
+
+ def activity_avatar(event)
+ return nil unless event.respond_to?(:event_author)
+
+ author = event.event_author
+ committer = event.respond_to?(:committer) ? event.committer : nil
+ avatar(author.is_a?(User) ? author : (committer || author))
+ end
end
diff --git a/app/views/activities/_activities.html.erb b/app/views/activities/_activities.html.erb
index aaeea76bc2..5de9a7fd37 100644
--- a/app/views/activities/_activities.html.erb
+++ b/app/views/activities/_activities.html.erb
@@ -4,7 +4,7 @@
<dl>
<% sort_activity_events(events_by_day[day]).each do |e, in_group| -%>
  <dt class="<%= e.event_type %> icon icon-<%= e.event_type %> <%= "grouped" if in_group %> <%= User.current.logged? &&
e.respond_to?(:event_author) && User.current == e.event_author ? 'me' : nil %>">
- <%= avatar(e.event_author) if e.respond_to?(:event_author) %>
+ <%= activity_avatar(e) %>
  <span class="time"><%= format_time(e.event_datetime, false) %></span>
  <%= content_tag('span', e.project, :class => 'project') if @project.nil? || @project != e.project %>
  <%= link_to format_activity_title(e.event_title), e.event_url %>
```

#4 - 2021-01-03 04:46 - Go MAEDA

Thank you for posting the patch but the test fails.

```
$ ruby test/functional/activities_controller_test.rb
```

```
Run options: --seed 15717
```

```
# Running:
```

```
.....E
```

```
Error:
```

```
ActivitiesControllerTest#test_index_with_mercurial_changesets_and_gravatar_enabled_should_display_gravatar:
```

```
NoMethodError: undefined method `committer' for nil:NilClass
```

```
test/functional/activities_controller_test.rb:312:in `test_index_with_mercurial_changesets_and_gravatar_enabled_should_display_gravatar'
```

bin/rails test test/functional/activities_controller_test.rb:302

#5 - 2021-03-05 06:44 - Yuichi HARADA

Go MAEDA wrote:

Thank you for posting the patch but the test fails.

[...]

I was able to reproduce the test error.

```
$ rm -rf tmp/test/mercurial_repository
```

```
$ ruby test/functional/activities_controller_test.rb
```

```
Run options: --seed 63630
```

```
# Running:
```

```
.....E
```

```
Error:
```

```
ActivitiesControllerTest#test_index_with_mercurial_changesets_and_gravatar_enabled_should_display_gravatar:
```

```
NoMethodError: undefined method `committer' for nil:NilClass
```

```
test/functional/activities_controller_test.rb:313:in `test_index_with_mercurial_changesets_and_gravatar_enabled_should_display_gravatar'
```

```
bin/rails test test/functional/activities_controller_test.rb:303
```

```
.....
```

```
Finished in 3.751118s, 4.7986 runs/s, 16.5284 assertions/s.
```

```
18 runs, 62 assertions, 0 failures, 1 errors, 0 skips
```

Please execute `bundle exec rake test:scm:setup:mercurial` to create the Mercurial repository.

Files

bugs_ruby-lang_org-activity.png	251 KB	2020-12-07	Go MAEDA
6082.patch	4.52 KB	2020-12-22	Yuichi HARADA