Redmine - Defect #13172

Changing assignee then tracker or status reverts unsaved assignee change (during issue edit/update)

2013-02-14 12:25 - Dietmar H

Status:	Confirmed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Issues	**Estimated time:**	0.00 hour
Target version:			
Resolution:		**Affected version:**	2.2.2

Description

In update issue view, when first changing assignee, then status, the assignee field is reverted to the previous state.

```
Environment:
  Redmine version                    2.2.2.stable
  Ruby version                       1.8.7 (i686-linux)
  Rails version                      3.2.11
  Environment                        production
  Database adapter                   SQLite
```

Related issues:

Related to Redmine - Defect #13250: Status change refresh problem - cannot wr...	**New**

History

#1 - 2013-02-14 18:38 - Jean-Philippe Lang

- Subject changed from Update issue: Changinge status reverts assignee field to Update issue: Changinge assignee then status reverts assignee

#2 - 2013-02-16 00:47 - Mischa The Evil

- Subject changed from Update issue: Changinge assignee then status reverts assignee to Update issue: Changing assignee then status reverts assignee

#3 - 2013-02-22 18:16 - Dietmar H

Changing Tracker has the same effect. Looks like all operations triggering Ajax requests reset the assignee field.

#4 - 2013-05-22 18:28 - Dietmar H

Does nobody have this issue or does nobody care? We find it quite annoying.
Config now is:

```
Environment:
  Redmine version                    2.2.3.stable
  Ruby version                       1.9.3 (i686-linux)
  Rails version                      3.2.12
  Environment                        production
  Database adapter                   Mysql2
```

#5 - 2013-05-25 00:33 - Mischa The Evil

- Subject changed from Update issue: Changing assignee then status reverts assignee to Changing assignee then tracker or status reverts unsaved assignee change (during issue edit/update)

- Status changed from New to Confirmed

I have done some testing for this issue. I am able to reproduce the described behavior:

- Environment:
    - Redmine 2.3.1.devel.11845 (trunk)
    - Ruby 1.9.3 & 2.0.0 [x86_64-linux]
    - Rails 3.2.13
    - Database adapter: Mysql2
    - Redmine plugins: no plugins installed

- Steps to reproduce:

1. Load Redmine default config
2. Create new issue
3. Assign the new issue an assignee (e.g. admin)
4. Update/edit the issue (but don't submit the form)
    1. Change the value of the assignee field (<< me >> is sufficient for this use-case)
    2. Change the tracker or status field value (which - as expected - triggers an Ajax request)

- Current behavior:
    - After 4.2: the assignee field value is reset from the new value (<< me >>, see 4.1) to the old value (admin, see 3)
- Expected behavior:
    - After 4.2: the assignee field value is kept (still << me >>, see 4.1)

- [Random trace (without errors)Random trace (without errors)](#)

```
Started GET "/issues/16" for xxx.xxx.xxx.xxx at 2013-05-25 00:10:50 +0200
Processing by IssuesController#show as HTML
  Parameters: {"id"=>"16"}
  Current user: admin (id=1)
  Rendered issues/_action_menu.html.erb (5.8ms)
  Rendered issue_relations/_form.html.erb (2.2ms)
  Rendered issues/_relations.html.erb (3.9ms)
  Rendered issues/_history.html.erb (36.4ms)
  Rendered issues/_action_menu.html.erb (4.9ms)
  Rendered issues/_form_custom_fields.html.erb (0.1ms)
  Rendered issues/_attributes.html.erb (24.7ms)
  Rendered issues/_form.html.erb (45.7ms)
  Rendered attachments/_form.html.erb (2.5ms)
  Rendered issues/_edit.html.erb (63.1ms)
  Rendered issues/_sidebar.html.erb (5.5ms)
  Rendered watchers/_watchers.html.erb (5.6ms)
  Rendered issues/show.html.erb within layouts/base (162.6ms)
Completed 200 OK in 276ms (Views: 169.0ms | ActiveRecord: 37.0ms)
Started PUT "/projects/test-2/issues/update_form.js?id=16" for xxx.xxx.xxx.xxx at 2013-05-25 00:11:09 +020
0
Processing by IssuesController#update_form as JS
  Parameters: {"utf8"=>"âœ"", "authenticity_token"=>"<<token>>", "issue"=>{"is_private"=>"0", "project_id"
=>"5", "tracker_id"=>"1", "subject"=>"8", "description"=>"", "status_id"=>"2", "priority_id"=>"3", "assign
ed_to_id"=>"1", "fixed_version_id"=>"10", "parent_issue_id"=>"", "start_date"=>"2013-05-05", "due_date"=>"
", "estimated_hours"=>"", "done_ratio"=>"0", "notes"=>"", "private_notes"=>"0", "lock_version"=>"7"}, "tim
e_entry"=>{"hours"=>"", "activity_id"=>"", "comments"=>""}, "last_journal_id"=>"78", "id"=>"16", "project_
id"=>"test-2"}
  Current user: admin (id=1)
  Rendered issues/_form_custom_fields.html.erb (5.9ms)
  Rendered issues/_attributes.html.erb (35.3ms)
  Rendered issues/_form.html.erb (47.4ms)
  Rendered issues/update_form.js.erb (49.0ms)
Completed 200 OK in 127ms (Views: 44.2ms | ActiveRecord: 17.4ms)
```

#6 - 2016-04-04 18:09 - @ go2null

The issue is that changing the status applies the new status, which it should not do as the new status is just a change, like any other other change done.

That is the new status permissions should NOT be enforced/triggered when in the input form.

Duplicate issue: Defect [#13250](#)

#7 - 2016-05-04 06:06 - Toshi MARUYAMA

- Related to Defect #13250: Status change refresh problem - cannot write fileds writeable in a given statatus added

#8 - 2017-03-15 18:43 - Dietmar H

Seems to be solved in 3.3.1 (don't know about older version).