

Redmine - Defect #8539

Fix NoMethodError in Issue#blocked? due to invalid issue_from_id in Issue#relations_from

2011-06-06 13:13 - Terence Mill

Status:	Closed	Start date:	2011-06-06
Priority:	Normal	Due date:	
Assignee:	Go MAEDA	% Done:	0%
Category:	Issues	Estimated time:	0.00 hour
Target version:	5.0.10	Affected version:	1.1.2
Resolution:	Fixed		

Description

We have a taks that raises following error if we try to access it in any way (edit, view, history, context menu in issues list view). The issue although is correctly listed in issue list view, and all values get displayed. We are not sure, since when that raises, but the issue was moved between projects in history, that is what we can remember, although that don't have to be the cause for that error.

How can we acces that issue again, or at least remive it completly

```
NoMethodError (undefined method `closed?' for nil:NilClass):
  app/models/issue.rb:421:in `blocked?'
  app/models/issue.rb:421:in `blocked?'
  app/models/issue.rb:430:in `new_statuses_allowed_to'
  app/controllers/issues_controller.rb:112:in `show'
  passenger (3.0.2) lib/phusion_passenger/rack/request_handler.rb:96:in `process_request'
  passenger (3.0.2) lib/phusion_passenger/abstract_request_handler.rb:513:in `accept_and_process_n
ext_request'
  passenger (3.0.2) lib/phusion_passenger/abstract_request_handler.rb:274:in `main_loop'
  passenger (3.0.2) lib/phusion_passenger/classic_rails/application_spawner.rb:321:in `start_reque
st_handler'
  passenger (3.0.2) lib/phusion_passenger/classic_rails/application_spawner.rb:275:in `send'
  passenger (3.0.2) lib/phusion_passenger/classic_rails/application_spawner.rb:275:in `handle_spaw
n_application'
  passenger (3.0.2) lib/phusion_passenger/utils.rb:479:in `safe_fork'
  passenger (3.0.2) lib/phusion_passenger/classic_rails/application_spawner.rb:270:in `handle_spaw
n_application'
  passenger (3.0.2) lib/phusion_passenger/abstract_server.rb:357:in `__send__'
  passenger (3.0.2) lib/phusion_passenger/abstract_server.rb:357:in `server_main_loop'
  passenger (3.0.2) lib/phusion_passenger/abstract_server.rb:206:in `start_synchronously'
  passenger (3.0.2) lib/phusion_passenger/abstract_server.rb:180:in `start'
  passenger (3.0.2) lib/phusion_passenger/classic_rails/application_spawner.rb:149:in `start'
  passenger (3.0.2) lib/phusion_passenger/spawn_manager.rb:219:in `spawn_rails_application'
  passenger (3.0.2) lib/phusion_passenger/abstract_server_collection.rb:132:in `lookup_or_add'
  passenger (3.0.2) lib/phusion_passenger/spawn_manager.rb:214:in `spawn_rails_application'
  passenger (3.0.2) lib/phusion_passenger/abstract_server_collection.rb:82:in `synchronize'
  passenger (3.0.2) lib/phusion_passenger/abstract_server_collection.rb:79:in `synchronize'
  passenger (3.0.2) lib/phusion_passenger/spawn_manager.rb:213:in `spawn_rails_application'
  passenger (3.0.2) lib/phusion_passenger/spawn_manager.rb:132:in `spawn_application'
  passenger (3.0.2) lib/phusion_passenger/spawn_manager.rb:275:in `handle_spawn_application'
  passenger (3.0.2) lib/phusion_passenger/abstract_server.rb:357:in `__send__'
  passenger (3.0.2) lib/phusion_passenger/abstract_server.rb:357:in `server_main_loop'
  passenger (3.0.2) lib/phusion_passenger/abstract_server.rb:206:in `start_synchronously'
  passenger (3.0.2) helper-scripts/passenger-spawn-server:99
```

Associated revisions

Revision 23048 - 2024-09-10 02:55 - Go MAEDA

Fix NoMethodError in Issue#blocked? due to invalid issue_from_id in Issue#relations_from (#8539).

Patch by Go MAEDA (user:maeda).

Revision 23049 - 2024-09-11 09:32 - Go MAEDA

Merged r23048 from trunk to 5.1-stable (#8539).

Revision 23050 - 2024-09-11 09:36 - Go MAEDA

Merged r23048 from trunk to 5.0-stable (#8539).

History

#1 - 2011-06-12 23:57 - Jean-Baptiste Barth

Can you give the output of the following commands in a ruby script/console production (after replacing 1234 by the real ID of your issue) :

```
Issue.find(1234).relations_to.each{|ir| puts "#{ir.relation_type} => #{ir.issue_from_id}"}
```

EDIT: fixed the code... please try this new one-liner instead

#2 - 2011-06-14 12:52 - Terence Mill

Result is..

```
>> Issue.find(84).relations_to.each{|ir| puts "#{ir.relation_type} => #{ir.issue_from_id}" }
blocks => 971
blocks => 972
=> [#<IssueRelation id: 50, issue_from_id: 971, issue_to_id: 84, relation_type: "blocks", delay: nil>, #<Issue
Relation id: 55, issue_from_id: 972, issue_to_id: 84, relation_type: "blocks", delay: nil>]
```

Tx for help!

#3 - 2024-09-07 10:40 - Go MAEDA

- File 8539.patch added

- Priority changed from High to Normal

- Target version set to Candidate for next major release

The issue is likely caused by the issue_from_id column in the issue_relations table pointing to a non-existent issue, possibly one that was manually deleted using an SQL DELETE command.

You can reproduce the error by setting an invalid issue_from_id value.

```
redmine-app(dev)> ir = IssueRelation.find(1)
  IssueRelation Load (0.3ms)  SELECT "issue_relations".* FROM "issue_relations" WHERE "issue_relations"."id" =
 ? LIMIT ? [{"id", 1}, ["LIMIT", 1]]
=> #<IssueRelation:0x00000001367f0ff8 id: 1, issue_from_id: 10, issue_to_id: 9, relation_type: "blocks", delay
: nil>
redmine-app(dev)> Issue.find(9).blocked?
  Issue Load (0.1ms)  SELECT "issues".* FROM "issues" WHERE "issues"."id" = ? LIMIT ? [{"id", 9}, ["LIMIT", 1
]]
  IssueRelation Load (0.5ms)  SELECT "issue_relations".* FROM "issue_relations" WHERE "issue_relations"."issue
_to_id" = ? [{"issue_to_id", 9}]
  Issue Load (0.1ms)  SELECT "issues".* FROM "issues" WHERE "issues"."id" = ? LIMIT ? [{"id", 10}, ["LIMIT",
1]]
  IssueStatus Load (0.1ms)  SELECT "issue_statuses".* FROM "issue_statuses" WHERE "issue_statuses"."id" = ? LI
MIT ? [{"id", 1}, ["LIMIT", 1]]
=> true
redmine-app(dev)>
redmine-app(dev)> ir.update_column(:issue_from_id, 9999)
  IssueRelation Update (1.8ms)  UPDATE "issue_relations" SET "issue_from_id" = ? WHERE "issue_relations"."id"
 = ? [{"issue_from_id", 9999}, ["id", 1]]
=> true
redmine-app(dev)> Issue.find(9).blocked?
  Issue Load (0.1ms)  SELECT "issues".* FROM "issues" WHERE "issues"."id" = ? LIMIT ? [{"id", 9}, ["LIMIT", 1
]]
  IssueRelation Load (0.0ms)  SELECT "issue_relations".* FROM "issue_relations" WHERE "issue_relations"."issue
_to_id" = ? [{"issue_to_id", 9}]
  Issue Load (0.0ms)  SELECT "issues".* FROM "issues" WHERE "issues"."id" = ? LIMIT ? [{"id", 9999}, ["LIMIT"
, 1]]
app/models/issue.rb:1021:in `block in blocked?': undefined method `closed?' for nil (NoMethodError)

!relations_to.detect {|ir| ir.relation_type == 'blocks' && !ir.issue_from.closed?}.nil?
                                     ^^^^^^^^^
  from app/models/issue.rb:1021:in `detect'
  from app/models/issue.rb:1021:in `blocked?'
```

The attached patch addresses the handling of nil values when issue_from is missing.

#4 - 2024-09-09 07:40 - Go MAEDA

- Target version changed from Candidate for next major release to 6.0.0

Setting the target version to 6.0.0.

#5 - 2024-09-09 17:54 - Marius BĂLTEANU

- Target version changed from 6.0.0 to 5.0.10

I think we can merge this small fix also to stable branches.

#6 - 2024-09-10 02:57 - Go MAEDA

- Subject changed from accessing issue raises error "undefined method `closed?' for nil:NilClass" to Fix NoMethodError in Issue#blocked? due to invalid issue_from_id in Issue#relations_from

- Status changed from New to Resolved

- Assignee set to Go MAEDA

- Resolution set to Fixed

Committed the fix in [r23048](#).

#7 - 2024-09-11 09:36 - Go MAEDA

- Status changed from Resolved to Closed

Merged the change to stable branches.

Files

8539.patch	1.26 KB	2024-09-07	Go MAEDA
------------	---------	------------	----------