

Redmine - Defect #20943

migrate_from_trac.rake dont work

2015-10-08 15:20 - Karel Blumentrit

Status:	Resolved	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Importers	Estimated time:	0.00 hour
Target version:		Affected version:	3.1.1
Resolution:			
Description			
Hello,			
I have encountered several issues with the migrate_from_trac.rake script, trying to import a Trac(V 1.0.2) Project with SQLite (V. 3.8.7.1) Database.			
My Enviroment:			
<pre>cat /etc/debian_version 8.2</pre>			
Environment:			
<pre>Redmine version 3.1.1.stable Ruby version 2.1.5-p273 (2014-11-13) [x86_64-linux-gnu] Rails version 4.2.4 Environment production Database adapter Mysql2</pre>			
SCM:			
<pre>Git 2.1.4 Filesystem</pre>			
Redmine plugins:			
<pre>no plugin installed</pre>			
After applying the Patches from #19173-6 and overcoming that issue, I also applied the patch from #14567 to overcome the timestamp conversion issue.			
However I now encountered a new Issue:			
<pre>rake redmine:migrate_from_trac RAILS_ENV="production" --trace (in /opt/redmine/redmine-3.1.1) ** Invoke redmine:migrate_from_trac (first_time) ** Invoke environment (first_time) ** Execute environment ** Execute redmine:migrate_from_trac</pre>			
WARNING: a new project will be added to Redmine during this process. Are you sure you want to continue ? [y/N] y			
<pre>Trac directory []: /tmp/project Trac database adapter (sqlite3, mysql2, postgresql) [sqlite3]: Trac database encoding [UTF-8]: Target project identifier []: project</pre>			
<pre>Trac database version is: 29 Migrating components..... Migrating milestones..... Migrating custom fields Migrating tickets.rake aborted! ArgumentError: wrong number of arguments (1 for 0) /var/lib/gems/2.1.0/gems/activerecord-4.2.4/lib/active_record/scoping/named.rb:24:in `all' /opt/redmine/redmine-3.1.1/lib/tasks/migrate_from_trac.rake:180:in `attachments' /opt/redmine/redmine-3.1.1/lib/tasks/migrate_from_trac.rake:539:in `block in migrate'</pre>			

```

/var/lib/gems/2.1.0/gems/activerecord-4.2.4/lib/active_record/relation/batches.rb:51:in `block (2
levels) in find_each'
/var/lib/gems/2.1.0/gems/activerecord-4.2.4/lib/active_record/relation/batches.rb:51:in `each'
/var/lib/gems/2.1.0/gems/activerecord-4.2.4/lib/active_record/relation/batches.rb:51:in `block in
find_each'
/var/lib/gems/2.1.0/gems/activerecord-4.2.4/lib/active_record/relation/batches.rb:124:in `find_in_
batches'
/var/lib/gems/2.1.0/gems/activerecord-4.2.4/lib/active_record/relation/batches.rb:50:in `find_each
'
/var/lib/gems/2.1.0/gems/activerecord-4.2.4/lib/active_record/querying.rb:9:in `find_each'
/opt/redmine/redmine-3.1.1/lib/tasks/migrate_from_trac.rake:486:in `migrate'
/opt/redmine/redmine-3.1.1/lib/tasks/migrate_from_trac.rake:797:in `block (2 levels) in <top (requ
ired)>'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:240:in `call'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:240:in `block in execute'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:235:in `each'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:235:in `execute'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:179:in `block in invoke_with_call_chain'
/usr/lib/ruby/2.1.0/monitor.rb:211:in `mon_synchronize'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:172:in `invoke_with_call_chain'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/task.rb:165:in `invoke'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:150:in `invoke_task'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:106:in `block (2 levels) in top_level
'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:106:in `each'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:106:in `block in top_level'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:115:in `run_with_threads'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:100:in `top_level'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:78:in `block in run'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:176:in `standard_exception_handling'
/var/lib/gems/2.1.0/gems/rake-10.4.2/lib/rake/application.rb:75:in `run'
/var/lib/gems/2.1.0/gems/rake-10.4.2/bin/rake:33:in `<top (required)>'
/usr/local/bin/rake:23:in `load'
/usr/local/bin/rake:23:in `<main>'
Tasks: TOP => redmine:migrate_from_trac

```

Please correct me if I am wrong, as I have absolutely no idea about ruby on rails programming, but (after reading throught this issue, which really ha nothing to do with the migration script: https://github.com/arkhitech/redmine_timesheet_plugin/issues/49) I assume that the Problem are those two lines:

```

180      TracMigrate::TracAttachment.all(:conditions => ["type = 'ticket' AND id = ?", self.i
d.to_s])

235      TracMigrate::TracAttachment.all(:conditions => ["type = 'wiki' AND id = ?", self.id.
to_s])

```

which passes arguments to the all() function, which doesnt accept any arguments anymore. This seems to have changed in some Ruby on Rails Version. Maybe someone knows how to fix this and could provide a patch I could test? I tried myself, but with litteraly no knowledge about Ruby on Rails programming I failed =P

I attached the migrate_from_trac.rake with all the patches included.

Related issues:

Related to Redmine - Patch #14567: migrate_from_trac.rake does not convert ti...	New
Related to Redmine - Defect #19173: Trac Migrate Script with Redmine 3.0	New
Related to Redmine - Defect #23268: migrate_from_trac.rake - NOT WORKING WITH...	New

History

#1 - 2015-10-08 16:57 - Karel Blumentrit

Ok, I found a solution: https://github.com/rails/activerecord-deprecated_finders

edit redmine-3.1.1/Gemfile

Add:

```
gem "activerecord-deprecated_finders", require: "active_record/deprecated_finders"
```

after editin, run *bundle install*

then you should be able to run the patched *migrate_from_trac.rake* script. I think this is rather a temporary solution, as the gem package add support for the old *all()* syntax in ActiveRecord. But at least it works :)

#2 - 2015-11-05 05:31 - Toshi MARUYAMA

- Related to Patch #14567: *migrate_from_trac.rake* does not convert timestamps in Trac database version 23 added

#3 - 2015-11-05 05:32 - Toshi MARUYAMA

- Related to Defect #19173: *Trac Migrate Script with Redmine 3.0* added

#4 - 2015-12-16 12:36 - Karel Blumentrit

- File *migrate_from_trac.rake* added

- Status changed from *New* to *Resolved*

There still was an issue with the script provided earlier, as importing of TRAC attachments still did not work. It seems that the attachment path has changed in the TRAC enviroment with the 1.0 Version. I have now fixed this. (credits also for https://github.com/eLvErDe/migrate_from_trac.rake/blob/master/migrate_from_trac.rake)

Now import from TRAC 1.0.2 to Redmine 3.1 works fine with the script provided:

```
rake redmine:migrate_from_trac RAILS_ENV="production"
```

```
WARNING: a new project will be added to Redmine during this process.
Are you sure you want to continue ? [y/N] y
```

```
Trac directory []: /tmp/project
Trac database adapter (sqlite3, mysql2, postgresql) [sqlite3]:
Trac database encoding [UTF-8]:
Target project identifier []: testprojekt
```

```
Trac database version is: 29
Migrating components.....
Migrating milestones.....
Migrating custom fields
Migrating tickets.....
.....
.....
.....
.....
.....
Migrating wiki.....
.....
.....
.....
.....
```

```
Components:      22/22
Milestones:      16/16
Tickets:         638/638
Ticket files:    144/144
Custom values:   0/0
Wiki edits:      611/611
Wiki files:      16/16
```

#5 - 2016-07-18 15:17 - Andreas Kohlbecker

Thank you Karel for that link to https://github.com/akohlbecker/migrate_from_trac.rake and for your improvements. I did further improve the migration script.

The improvements include:

- Better and more complete textile conversion
- Cleaning up of messed up user names in ticket fields
- Migrating trac severities
- Migrating priorities
- Migrating parent/child issue relations
- Closing completed milestones

- Support for multiple storage policies of various trac versions
- Option to preserve Trac ticket ids
- ...

Please see my commit logs at github for more details: https://github.com/akohlbecker/migrate_from_trac.rake

With this script I successfully migrated a trac instance which had over 5000 tickets and ~500 Wikipages with ~1000 attachments in total. The trac instance was initially setup with version 0.7 and was upgraded to version 0.12.

Feel free to use this script or parts of it for redmine.

#6 - 2016-07-18 15:20 - Andreas Kohlbecker

Andreas Kohlbecker wrote:

Thank you Karel for that link to https://github.com/akohlbecker/migrate_from_trac.rake and for your improvements.

Sorry, this was the wrong link, the right one is of course : https://github.com/eLvErDe/migrate_from_trac.rake/blob/master/migrate_from_trac.rake

I can't edit my posting, therefore I am fixing it with this comment.

#7 - 2016-08-09 14:00 - Toshi MARUYAMA

- Related to Defect #23268: migrate_from_trac.rake - NOT WORKING WITH trac 1.0.1 added

#8 - 2017-08-01 13:32 - Matt Schneider

I'm trying to migrate from Trac to Redmine but running into problems with migrate_from_trac.rake script provided by Andreas (see link above).

This is the error I get:

```
root@2d2a590c9cf5:/usr/src/redmine# rake redmine:migrate_from_trac RAILS_ENV="production"

WARNING: a new project will be added to Redmine during this process.
Are you sure you want to continue ? [y/N] y

Trac directory []: /uss/src/redmine/files/home/matthias/tracbackup
This directory doesn't exist!
Trac directory []: /usr/src/redmine/files/home/matthias/tracbackup
Trac database adapter (sqlite3, mysql2, postgresql) [sqlite3]:
Trac database encoding [UTF-8]:
Target project identifier []: asv
Preserve Trac ticket ids (y|n) [y]:
Migrate or map Trac priorities and ticket types? (migrate|map) [migrate]:

Trac database version is: 41
Migrating ticket types          [*****] 100%
Migrating priorities            [*****] 100%
rake aborted!erities           [ ] 0%
ActiveRecord::RecordInvalid: Validation failed: Possible values cannot be blank
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/validations.rb:79:in `raise_record_invalid'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/validations.rb:43:in `save!'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/attribute_methods/dirty.rb:29:in `save!'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/transactions.rb:291:in `block in save!'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/transactions.rb:351:in `block in with_transaction_returning_status'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/connection_adapters/abstract/database_statements.r
b:213:in `block in transaction'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/connection_adapters/abstract/transaction.rb:184:in
`within_new_transaction'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/connection_adapters/abstract/database_statements.r
b:213:in `transaction'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/transactions.rb:220:in `transaction'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/transactions.rb:348:in `with_transaction_returning
_status'
/usr/local/bundle/gems/activerecord-4.2.8/lib/active_record/transactions.rb:291:in `save!'
/usr/src/redmine/lib/tasks/migrate_from_trac.rake:547:in `migrate'
/usr/src/redmine/lib/tasks/migrate_from_trac.rake:1360:in `block (2 levels) in <top (required)>'
/usr/local/bundle/gems/rake-12.0.0/exe/rake:27:in `<top (required)>'
Tasks: TOP => redmine:migrate_from_trac
(See full trace by running task with --trace)
```

Anyone around who could help?

thanks,
Matt

Files

migrate_from_trac.rake	30 KB	2015-10-08	Karel Blumentrit
migrate_from_trac.rake	31 KB	2015-12-16	Karel Blumentrit