

Impossible to create a new version

2022/05/09 18:24 - Admin Redmine

ステータス:	Needs feedback	開始日:	2022/05/09
優先度:	通常	期日:	
担当者:		進捗率:	0%
カテゴリ:	Projects_11	予定工数:	0.00時間
対象バージョン:		作業時間:	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/27319	status_id:	10
category_id:	11	tracker_id:	1
version_id:	0	plus1:	0
issue_org_id:	27319	affected_version:	
author_id:	324696	closed_on:	
assigned_to_id:	0	affected_version_id:	133
comments:	25		

説明

Hello,

I recently upgraded to Redmine 3.4.3, and I can't create new versions in projects. I have this error :

Status is not included in the list

Nothing in the log...

journals

Are you experiencing the problem in all projects? Or in a particular project?

In all project. The field status for the new version form is not appearing anymore, triggering the error on submit.

I don't understand the cause, but it may be caused by some plugin. Could you try without any plugin?

I've removed all plugins, and changed the theme to the default one to be sure. Same issue.

Yohann Prigent wrote:

In all project. The field status for the new version form is not appearing anymore, triggering the error on submit.

What is "The field status for the new version"?

Here you can see two screenshots.

The first one with the default theme on the page to create a new version, where the field Status is not added (and generating the version).

The second one on another Redmine instance not upgraded which have the field Status, no not generating the error when I try to save this new version.

Yohann Prigent wrote:

The first one with the default theme on the page to create a new version, where the field Status is not added (and generating

the version).

It is expected behaviour.

source:tags/3.4.3/app/views/versions/_form.html.erb#L7

Check your db definition.

source:tags/3.4.3/db/migrate/20091108092559_add_versions_status.rb#L3

```
add_column :versions, :status, :string, :default => 'open'
```

----- I do not have the default open added to my column in my db (MariaDB). I tried to run again `@bundle exec rake db:migrate RAILS_ENV=production@` but same issue. I've added the field manually to the db and now it works... Migration issue ? ----- MariaDB is not supported.

----- @Toshi, how exactly do I check the db definition? I am having this exact issue, however I am using the mysql2 db adapter. Background on my instance of RM, we started out with Redmine v1.2 long time ago, and I've been steadily upgrading with most new versions. We haven't had a problem until we recently moved from v3.3 to v3.4. EDIT: I've run the rake commands several times like so, but the issue remains:

```
bundle exec rake db:migrate RAILS_ENV=production
bundle exec rake redmine:plugins:migrate RAILS_ENV=production
bundle exec rake tmp:cache:clear tmp:sessions:clear RAILS_ENV=production
```

EDIT2: I just tried to run this command thinking it might add the column that's possibly missing (based on comment !#9), but the issue persists:

```
bundle exec add_column :versions, :status, :string, :default => 'open'
```

EDIT3: Tried another rake command, but still no luck:

```
# bundle exec rake redmine:load_default_data
```

Select language: ar, az, bg, bs, ca, cs, da, de, el, en, en-GB, es, es-PA, et, eu, fa, fi, fr, gl, he, hr, hu, id, it, no, pl, pt, pt-BR, ro, ru, sk, sl, sq, sr, sr-YU, sv, th, tr, uk, vi, zh, zh-TW [en]

=====
Some configuration data is already loaded.

Details on my environment:

Environment:

Redmine version	3.4.3.stable
Ruby version	2.4.2-p198 (2017-09-14) [x86_64-linux]
Rails version	4.2.8
Environment	production
Database adapter	Mysql2

SCM:

Subversion	1.8.10
Mercurial	3.1.2
Bazaar	2.7.0
Git	2.1.4
Filesystem	

Redmine plugins:

clipboard_image_paste	1.12
redmine_base_deface	0.0.1
redmine_maintenance_mode	2.1.0
redmine_theme_changer	0.3.1

Adam Gay wrote:

@Toshi, how exactly do I check the db definition?

```
mysql> desc versions;
```

```
+-----+-----+-----+-----+-----+
```

Field	Type	Null	Key	Default	Extra
-------	------	------	-----	---------	-------

```

+-----+-----+-----+-----+
| id          | int(11) | NO | PRI | NULL | auto_increment |
| project_id  | int(11) | NO | MUL | 0     |                 |
| name        | varchar(255) | NO |      |       |                 |
| description  | varchar(255) | YES |      |       |                 |
| effective_date | date    | YES |      | NULL  |                 |
| created_on   | datetime | YES |      | NULL  |                 |
| updated_on   | datetime | YES |      | NULL  |                 |
| wiki_page_title | varchar(255) | YES |      | NULL  |                 |
| status       | varchar(255) | YES |      | open  |                 |
| sharing      | varchar(255) | NO  | MUL | none  |                 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)

```

Thanks Toshi. I have made my status field match yours. After I set the status column to default 'open', and then ran the same rake commands, I still have the issue.

```

mysql> show columns from versions;
+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+
| id             | int(11)       | NO   | PRI | NULL    | auto_increment |
| project_id     | int(11)       | NO   | MUL | 0       |                 |
| name           | varchar(255)  | NO   |      |         |                 |
| description     | varchar(255)  | YES  |      |         |                 |
| effective_date | date          | YES  |      | NULL    |                 |
| created_on      | datetime      | YES  |      | NULL    |                 |
| updated_on      | datetime      | YES  |      | NULL    |                 |
| wiki_page_title | varchar(255)  | YES  |      | NULL    |                 |
| status          | varchar(255)  | YES  |      | open    |                 |
| sharing         | varchar(255)  | NO   | MUL | none    |                 |
| ir_start_date   | date          | YES  |      | NULL    |                 |
| ir_end_date     | date          | YES  |      | NULL    |                 |
+-----+-----+-----+-----+
12 rows in set (0.00 sec)

```

EDIT: Made my versions table match yours exactly, except for the two additional columns I have. The issue still persists.

Here's a screenshot of the page after I made the db changes. The "status" drop down does not appear. However if I go and try to edit an already existing version, the "status" dropdown is available and I can submit the changes without issue.

!status_error.png!

I can manually add the new version via MySQL console, and then go into the Redmine interface and edit it. This at least allows my software team to continue working, however it is not ideal if I have to add new versions via console every time.

EDIT: for reference to anyone out there that may be experiencing this same issue with MySQL, here's what I ran in MySQL to add the info:

```

INSERT INTO versions (project_id,name,status,sharing)
VALUES (1,'name of version','open','descendants');

```

I updated a few components on my server's backend last night, and after a soft reboot of the server itself my issue has now magically disappeared.

It seems like the issue may have been resolved after the update to v3.4.3 by:

Manually setting the MySQL db details as above (status to default 'open')

Running rake commands

Reboot the server

It's unclear what step was actually the resolution. Perhaps an Apache restart? I don't use Passenger in my instance, and it is virtualized behind a proxy, so it doesn't fully make sense to me that the Apache restart on the bare server would have done anything. I had rebooted the VM a couple of times yesterday with no luck, but a full server reboot seemed to do the trick. Odd, but at least we're back to normal now.

We experienced the same; adding a default value of "open" to the "status" field of the "version" table fixed the issue.

We also ran into this problem after upgrading to a version > 3.4.0. It was only with one of the two redmine instances we maintain (the older one, which has been running since 2008; both on MySQL).

Checking with @DESCRIBE versions;@, both the @status@ and the @sharing@ columns didn't have the defaults defined. (Additionally, NULL was allowed for @sharing@.) I'm not sure how to reproduce those DB differences, or whether it's feasible to do so. The change in #23377 caused it to surface, but I bet the DB differences have been present for a long time on that system.

The fix was to alter the columns to how they're supposed to be (this is for MySQL and redmine 4.0.4):

```
ALTER TABLE versions CHANGE status status varchar(255) DEFAULT 'open';
ALTER TABLE versions CHANGE sharing sharing varchar(255) DEFAULT 'none' NOT NULL;
```

Whatever serves your redmine application needs to be restarted afterwards (with passenger, just restart the redmine application).

Max Herold wrote:

We also ran into this problem after upgrading to a version > 3.4.0. It was only with one of the two redmine instances we maintain (the older one, which has been running since 2008; both on MySQL).

Do you or did you use any plugin? Maybe one of them removed those defaults. Otherwise, I don't understand why those defaults weren't set by the migrations.

Thank you, the alter cmd worked. Is there more db migrations steps missing? I'm worried about other tables missing fields/data...

Posted duplicate issue here #36384

I am seeing the same issue after migrating from 3.3.1 to 4.2.3.

I retested the migration with no plugins installed and the problem persisted.

My versions table looks like this after the migration.

```
mysql> show columns from versions;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
project_id	int(11)	NO	MUL	0	
name	varchar(255)	NO		NULL	
description	varchar(255)	YES		NULL	
effective_date	date	YES		NULL	
created_on	datetime	YES		NULL	
updated_on	datetime	YES		NULL	
wiki_page_title	varchar(255)	YES		NULL	
status	varchar(255)	YES		NULL	
sharing	varchar(255)	NO	MUL	NULL	

```
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

Note: If I bring up a fresh redmine 3.3.1, the status field does have a correct default set. So presumably the migration issue occurred before 3.3.1 but wasn't an issue until the Status field was removed from the new version form.

```
mysql> show columns from versions;
```

```
+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+
| id | int(11) | NO | PRI | NULL | auto_increment |
| project_id | int(11) | NO | MUL | 0 | |
| name | varchar(255) | NO | | | |
| description | varchar(255) | YES | | | |
| effective_date | date | YES | | NULL | |
| created_on | datetime | YES | | NULL | |
| updated_on | datetime | YES | | NULL | |
| wiki_page_title | varchar(255) | YES | NULL | | |
| status | varchar(255) | YES | | open | |
| sharing | varchar(255) | NO | MUL | none | |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

So for completeness, I dumped the tables from a fresh 4.2.3 database and from my migrated 4.2.3 database. Cleaned up the output some and diffed them. There are quite a lot of fields from different tables which have the wrong default and I'm not sure what the correct way of fixing them all is.

Some other notable differences:

- At some point datetime must have been changed to timestamp but wasn't migrated
- At some point mediumtext must have been changed to text but wasn't migrated

```
mysqldump (redacted) --no-data > dump-defs-4.2.3-fresh.sql
```

```
mysqldump (redacted) --no-data > dump-defs-4.2.3-migrated.sql
```

```
cp dump-defs-4.2.3-migrated.sql dump-defs-4.2.3-migrated-noplugins.sql
```

```
# Manually removed plugin tables
```

```
# Trim CHARSET and Collate settings, also AUTO_INCREMENT to reduce diff size
```

```
sed -e 's/DEFAULT CHARSET=[A-Za-z0-9_]*[ ;]//g; s/COLLATE.[A-Za-z0-9_]*[ ;]//g; s/AUTO_INCREMENT=[0-9]* //g' dump-defs-4.2.3-fresh.sql > dump-defs-4.2.3-fresh-trimmed.sql
```

```
sed -e 's/DEFAULT CHARSET=[A-Za-z0-9_]*[ ;]//g; s/COLLATE.[A-Za-z0-9_]*[ ;]//g; s/AUTO_INCREMENT=[0-9]* //g' dump-defs-4.2.3-migrated-noplugins.sql > dump-defs-4.2.3-migrated-noplugins-trimmed.sql
```

```
diff -u -d --strip-trailing-cr dump-defs-4.2.3-fresh-trimmed.sql dump-defs-4.2.3-migrated-noplugins-trimmed.sql > dump-defs-4.2.3.diff
```

Following @Max Herolds suggestion, i fixed the versions default values. Note I also fixed the name and description field to match the fresh 4.2.3 database. I needed to restart redmine for this to take effect.

```
mysql> ALTER TABLE versions CHANGE status status varchar(255) DEFAULT 'open';
```

```
mysql> ALTER TABLE versions CHANGE sharing sharing varchar(255) DEFAULT 'none' NOT NULL;
```

```
mysql> ALTER TABLE versions CHANGE name name varchar(255) DEFAULT " NOT NULL;
```

```
mysql> ALTER TABLE versions CHANGE description description varchar(255) DEFAULT ";
```

```
mysql> show columns from versions;
```

```
+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+
| id | int(11) | NO | PRI | NULL | auto_increment |
| project_id | int(11) | NO | MUL | 0 | |
| name | varchar(255) | NO | | | |
| description | varchar(255) | YES | | | |
| effective_date | date | YES | | NULL | |
| created_on | datetime | YES | | NULL | |
+-----+-----+-----+-----+
```

updated_on	datetime	YES		NULL	
wiki_page_title	varchar(255)	YES		NULL	
status	varchar(255)	YES		open	
sharing	varchar(255)	NO	MUL	none	
+-----+-----+-----+-----+-----+					

I would like to know if there is a easy way to fix all the other broken defaults incase they cause issues at a later time. Manually fixing them would take forever.

related_issues
relates,Closed,32556,Cannot create project version
duplicates,Closed,36384,Error on roadmap creation after migrating to 4.2.3

履歴

#1 - 2022/05/10 17:05 - Admin Redmine

- カテゴリ を Projects_11 にセット