

## Redmine - Feature #22913

### Auto-select fields mapping in Importing

2016-05-27 15:55 - Haihan Ji

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Go MAEDA	<b>% Done:</b>	0%
<b>Category:</b>	Importers	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	4.2.0		
<b>Resolution:</b>	Fixed		
<b>Description</b> I hate to select many times in import-mapping. This patch can auto select by label before you select it.			
<b>Related issues:</b> Related to Redmine - Feature #28234: Add CSV Import for Time Entries <b>Closed</b> Related to Redmine - Defect #34326: CSV import raises an exception if CSV hea... <b>Closed</b> Related to Redmine - Defect #35131: Issue import - allow auto mapping for Uni... <b>Closed</b>			

#### Associated revisions

##### Revision 19523 - 2020-02-19 15:31 - Go MAEDA

Use 'user' as internal field instead of user\_id because the column accepts also user login as value, not only the id (#22913).

Patch by Marius BALTEANU.

##### Revision 19524 - 2020-02-19 15:52 - Go MAEDA

Auto select fields mapping in import based on the internal field name (ex: estimated\_hours, fixed\_version, spent\_on) or field label (Estimated hours, Version, Date) (#22913).

- mappings are case insensitive
- a field is auto mapped only if there is no mapping setting present for that field
- "Current user" default value for User field when the user has permission to log time for other users is override by the auto mapped column

Patch by Haihan Ji, Yuichi HARADA, and Marius BALTEANU.

##### Revision 20576 - 2020-12-05 01:21 - Go MAEDA

CSV import raises an exception if CSV header has empty columns (#22913, #34326).

#### History

##### #1 - 2016-06-01 09:22 - Sebastian Paluch

+1

##### #2 - 2016-06-05 03:13 - Go MAEDA

Very interesting feature.  
Could you add tests? Tests are required to be merged into the core.

##### #3 - 2016-06-06 12:28 - Haihan Ji

- File 20160606\_auto\_select\_fields\_test.patch added

Append Unit Test Code.

##### #4 - 2016-06-13 03:10 - Go MAEDA

- Target version set to Candidate for next major release

##### #5 - 2016-12-26 13:54 - Go MAEDA

- Status changed from New to Needs feedback

I tried this patch but got the following error while processing ImportsController#mapping. Could you test the patch on the current trunk?

```
NameError (undefined local variable or method `issue' for #<ImportsController:0x007fc8be2479b0>
Did you mean?  issue_url):
  app/controllers/imports_controller.rb:72:in `mapping'
  lib/redmine/sudo_mode.rb:63:in `sudo_mode'
```

#### #6 - 2019-11-26 07:00 - Yuichi HARADA

- File 22913\_auto\_mapping.patch added

I fixed [auto\\_select\\_fields.patch](#) and [20160606\\_auto\\_select\\_fields\\_test.patch](#) to work with the current trunk([r19312](#)).  
I attach a fixed patch.

#### #7 - 2019-11-26 16:37 - Joshua Jobin

Yuichi HARADA wrote:

I fixed [auto\\_select\\_fields.patch](#) and [20160606\\_auto\\_select\\_fields\\_test.patch](#) to work with the current trunk([r19312](#)).  
I attach a fixed patch.

This is great. Thank you!

Do you find that if custom fields are absent from the default tracker, that they will not match when you load a tracker with more custom fields? I implemented this exactly and found that to be the case.

#### #8 - 2019-12-20 09:51 - Go MAEDA

Joshua Jobin wrote:

Do you find that if custom fields are absent from the default tracker, that they will not match when you load a tracker with more custom fields? I implemented this exactly and found that to be the case.

Could you show the steps to reproduce the problem?

I didn't see the problem you pointed out with Yuichi HARADA's patch applied. What I did is disable a custom field only in the first tracker.

#### #9 - 2020-01-04 02:43 - Go MAEDA

- Status changed from Needs feedback to New

- Target version changed from Candidate for next major release to 4.2.0

Setting the target version to 4.2.0.

#### #10 - 2020-01-04 10:45 - Marius BĂLTEANU

- Assignee set to Marius BĂLTEANU

Go MAEDA wrote:

Setting the target version to 4.2.0.

The patch cannot be committed as it is, please let me review it.

#### #11 - 2020-01-27 00:15 - Marius BĂLTEANU

- Related to Feature #28234: Add CSV Import for Time Entries added

#### #12 - 2020-01-27 00:46 - Marius BĂLTEANU

- File demo\_auto\_map\_fields.patch added

- Assignee deleted (Marius BĂLTEANU)

Sorry for my late review on this.

My main concern with the proposed patch are the hardcoded fields from mapping method (ImportsController). Because of them, the "auto mapping" feature it won't be easily extended by plugins or other models (it is against the generalisation added in [r18145](#)).

I'm attaching a patch with another solution, where each subclass of the import have a class constant (AUTO\_MAPPABLE\_FIELDS in my patch) or a method that returns an array with the fields that can be auto-mapped. In ImportsController, the method auto\_map\_fields uses this constant and try to

auto map the fields based on some conditions. This auto mapping uses only the "internal" name of the fields, not labels or translations.

To support labels and translations as well, I see two options:

1. We modify the `AUTO_MAPPABLE_FIELDS` to return a hash with each internal field and his label. For example:

```
AUTO_MAPPABLE_FIELDS = [
  'tracker' => 'label_tracker',
  'status' => 'label_status',
  ...
]
```

and then inside the `auto_map_fields` method we use it.

2. We use Javascript where we have access to all the required information, we just need to iterate throw each select element. This solution can handle both cases.

#### #13 - 2020-02-13 02:35 - Yuichi HARADA

- File `22913_auto_mapping_v2.patch` added

Marius BALTEANU wrote:

My main concern with the proposed patch are the hardcoded fields from mapping method (ImportsController). Because of them, the "auto mapping" feature it won't be easily extended by plugins or other models (it is against the generalisation added in [r18145](#)).

I'm attaching a patch with another solution, where each subclass of the import have a class constant (`AUTO_MAPPABLE_FIELDS` in my patch) or a method that returns an array with the fields that can be auto-mapped. In ImportsController, the method `auto_map_fields` uses this constant and try to auto map the fields based on some conditions. This auto mapping uses only the "internal" name of the fields, not labels or translations.

To support labels and translations as well, I see two options:

1. We modify the `AUTO_MAPPABLE_FIELDS` to return a hash with each internal field and his label. For example:

[...]

and then inside the `auto_map_fields` method we use it.

2. We use Javascript where we have access to all the required information, we just need to iterate throw each select element. This solution can handle both cases.

Marius, thank you for pointing it out. It was very helpful because there were some specifications that I knew for the first time.

I fixed [22913\\_auto\\_mapping.patch](#) based on the provided sample code.

#### #14 - 2020-02-16 11:41 - Marius BĂLTEANU

- File `0002-Auto-select-fields-mapping-in-import-based-on-the-in.patch` added

- File `0001-Use-user-as-internal-field-instead-of-user_id-becaus.patch` added

Thanks Yuichi for updating the patch.

I tried it and I've found some issues:

- the auto mapping is case sensitive. For example: `tracker` is not mapped to `Tracker`.
- auto mapping by internal fields name (`tracker`, `estimated_hours`, `cf_6`, `spent_on`) is not working. I find it useful to allow this as well for advanced users.
- the auto mapping works only when `import.mapping` is empty. I don't think that this condition is necessary because you already check inside the `auto_map_fields` method if the mapping for that field exists or not.

I decided to rework a little bit your patch in order to fix those issues, other edge cases and to reuse the existing import time entries CSV file, please let me know what do you thing. The first patch changes the internal field name for User from `user_id` to `user` because the import accepts also logins.

Test results here: <https://gitlab.com/redmine-org/redmine/pipelines/118278329>

#### #15 - 2020-02-18 02:49 - Yuichi HARADA

Marius BALTEANU wrote:

Thanks Yuichi for updating the patch.

I tried it and I've found some issues:

- the auto mapping is case sensitive. For example: `tracker` is not mapped to `Tracker`.
- auto mapping by internal fields name (`tracker`, `estimated_hours`, `cf_6`, `spent_on`) is not working. I find it useful to allow this as well for advanced users.
- the auto mapping works only when `import.mapping` is empty. I don't think that this condition is necessary because you already check inside the `auto_map_fields` method if the mapping for that field exists or not.

I decided to rework a little bit your patch in order to fix those issues, other edge cases and to reuse the existing import time entries CSV file, please let me know what do you thing. The first patch changes the internal field name for User from user\_id to user because the import accepts also logins.

Test results here: <https://gitlab.com/redmine-org/redmine/pipelines/118278329>

Thanks, Marius for improving the patches. I confirmed that it works properly.

In particular, it was very good that the test/functional/imports\_controller\_test.rb was written in detail and was easy to understand. However, I think it would be better if the following were modified.

```
diff --git a/app/controllers/imports_controller.rb b/app/controllers/imports_controller.rb
index 664d40019..96c493a31 100644
--- a/app/controllers/imports_controller.rb
+++ b/app/controllers/imports_controller.rb
@@ -186,7 +186,7 @@ class ImportsController < ApplicationController
   next if mappings.include?(field_nm)
   index = headers.index(field_nm) || headers.index(field.name.downcase)
   if index
-     mappings[field_nm] = headers.index(field_nm) || headers.index(field.name.downcase)
+     mappings[field_nm] = index
   end
 end
 mappings
```

#### #16 - 2020-02-18 18:02 - Marius BĂLTEANU

- File 0002-Auto-select-fields-mapping-in-import-based-on-the-in.patch added

Yuichi HARADA wrote:

Marius BALTEANU wrote:  
[...]

Thanks, Marius for improving the patches. I confirmed that it works properly.

You're welcome.

In particular, it was very good that the test/functional/imports\_controller\_test.rb was written in detail and was easy to understand. However, I think it would be better if the following were modified.

Agree, I've updated the second patch. Thanks for pointing this out.

#### #17 - 2020-02-18 18:02 - Marius BĂLTEANU

- File deleted (0002-Auto-select-fields-mapping-in-import-based-on-the-in.patch)

#### #18 - 2020-02-19 06:10 - Go MAEDA

The patch series works fine. Thank you.  
I will commit them in a few days.

#### #19 - 2020-02-19 06:18 - Marius BĂLTEANU

- Assignee set to Go MAEDA

#### #20 - 2020-02-19 15:54 - Go MAEDA

- Subject changed from Auto-select fields mapping in Importing to Auto-select fields mapping in Importing a CSV file  
- Status changed from New to Closed

Committed the patches.

Thank you all for working on this nice improvement.

#### #21 - 2020-02-20 00:27 - Go MAEDA

- Subject changed from Auto-select fields mapping in Importing a CSV file to Auto-select fields mapping in Importing

#### #22 - 2020-11-27 06:58 - Go MAEDA

- Related to Defect #34326: CSV import raises an exception if CSV header has empty columns added

**#23 - 2021-03-28 01:10 - Go MAEDA**

- Tracker changed from Patch to Feature
- Resolution set to Fixed

**#24 - 2021-04-20 21:08 - Marius BĂLTEANU**

- Related to Defect #35131: Issue import - allow auto mapping for Unique ID and relation type fields added

**Files**

auto_select_fields.patch	1.14 KB	2016-05-27	Haihan Ji
20160606_auto_select_fields_test.patch	1.37 KB	2016-06-06	Haihan Ji
22913_auto_mapping.patch	4.47 KB	2019-11-26	Yuichi HARADA
demo_auto_map_fields.patch	2.35 KB	2020-01-26	Marius BĂLTEANU
22913_auto_mapping_v2.patch	6.36 KB	2020-02-13	Yuichi HARADA
0001-Use-user-as-internal-field-instead-of-user_id-becaus.patch	4.4 KB	2020-02-16	Marius BĂLTEANU
0002-Auto-select-fields-mapping-in-import-based-on-the-in.patch	10 KB	2020-02-18	Marius BĂLTEANU