# Redmine - Defect #35802

## Redmine profile language form lists non-localized languages as English

2021-08-23 18:10 - Florian Krupicka

| | | | | |
|---|---|---|---|---|
| **Status:** | Needs feedback | | **Start date:** | |
| **Priority:** | Normal | | **Due date:** | |
| **Assignee:** | Florian Krupicka | | **% Done:** | 0% |
| **Category:** | | | **Estimated time:** | 0.00 hour |
| **Target version:** | | | | |
| **Resolution:** | Cant reproduce | | **Affected version:** | |

| Description |
|---|
| In our Redmine installation we found, that the drop down for preferred language was listing "English" twice.<br><br>After some investigation, we found out, that Redmine, using the standard Rails I18N gem, takes all available locales from the I18N backend. A library dependency from one of our custom plugins actually provided I18N for the Khmer language (km). Since the displayed language name is localized in Redmine itself and the fallback mechanism is in place, it was displayed as "English", while still using the language code in several places, resulting in strange error messages in the log file. |

## History

### #1 - 2021-08-24 01:42 - Mischa The Evil

*- Assignee set to Florian Krupicka*

Just to make sure that I understand your report correctly, it actually is the library dependency (and not your custom Redmine plugin) that is providing a *km.yml* file?
Can you provide the name of / link to the library? I did a quick test trying to reproduce the issue and failed to do so.
Can you test if the following change fixes the observed behavior?

```
config/application.rb | 1 +
1 file changed, 1 insertion(+)

diff --git a/config/application.rb b/config/application.rb
index fc6e6a33f..23f86ac6a 100644
--- a/config/application.rb
+++ b/config/application.rb
@@ -41,6 +41,7 @@ module RedmineApp

    # The default locale is :en and all translations from config/locales/*.rb,yml are auto loaded.
    # config.i18n.load_path += Dir[Rails.root.join('my', 'locales', '*.{rb,yml}').to_s]
+    config.i18n.load_path = Dir[Rails.root.join('config', 'locales', '*.{yml}').to_s]
    # config.i18n.default_locale = :de

    config.i18n.enforce_available_locales = true
```

### #2 - 2021-08-24 21:17 - Florian Krupicka

The dependency in question is [validate_url](#) (locale file in question is validate_url-1.0.13/lib/locale/km.yml).

While I cannot verify your proposed configuration right now, it does look like it would break other Redmine plugins, which seldom add their own locales to the i18n load path but instead rely on the default behaviour of the I18N gem (simple grep -r load_path redmine/plugins didn't find anything related to i18n). So it would fix the observed behaviour but would leave us with broken I18N for the plugins themselves.

I guess the way forward would be to ensure, that Redmine only displays those locales as valid options, for which itself has appropriate translations. I would like to contribute a fix to Redmine, but since the project does not monitor the inofficial GitHub mirror and the actual way to contribute actual code patches is not documented in CONTRIBUTING.md, I could use some pointers on how to proceed here. Should I add a diff patch to this issue?

### #3 - 2021-08-24 23:17 - Mischa The Evil

*- Status changed from New to Needs feedback*

*- Resolution set to Cant reproduce*

Florian Krupicka wrote:

> The dependency in question is [validate_url](#) (locale file in question is validate_url-1.0.13/lib/locale/km.yml).

I just did a quick test pulling in that library from a custom Redmine plugin, but still wasn't able to reproduce the given behavior. What Redmine version are you observing this on?

Florian Krupicka wrote:

> While I cannot verify your proposed configuration right now, it does look like it would break other Redmine plugins, which seldom add their own locales to the i18n load path but instead rely on the default behaviour of the I18N gem [...]

FTR: that change was just part of the steps of trying to pin down the cause of the behavior you are reporting. Nevertheless shouldn't it cause any issues with any regular Redmine plugins, since they don't (have to) tamper with the i18n load path themselfs. That is actually handled by the Redmine core plugin API itself (see [source:/trunk/lib/redmine/plugin.rb@21201#L107](source:/trunk/lib/redmine/plugin.rb@21201#L107)).

Florian Krupicka wrote:

> I guess the way forward would be to ensure, that Redmine only displays those locales as valid options, for which itself has appropriate translations.

Please feel free to propose alternative improvements to the Redmine i18n backend (residing in [source:/trunk/lib/redmine/i18n.rb@21201](source:/trunk/lib/redmine/i18n.rb@21201)), but what you're saying is that Redmine should ignore translation files that themselfs do not provide a custom translation string for the general_lang_name key? I doubt whether that is an actual improvement to be honest.

Florian Krupicka wrote:

> I would like to contribute a fix to Redmine, [...]

As previously stated: you are welcome to do so. Thanks in advance.

Florian Krupicka wrote:

> [...] and the actual way to contribute actual code patches is not documented in CONTRIBUTING.md, I could use some pointers on how to proceed here. Should I add a diff patch to this issue?

If you can attach a patch file that would be great. This "process" is documented as such on [Contribute](Contribute), which itself is linked to from [source:/trunk/CONTRIBUTING.md@21201](source:/trunk/CONTRIBUTING.md@21201).
Please make sure to include sufficient test coverage that clearly demonstrates the actual issues you're solving and how you're solving them by showing what you're expecting from the respective code.

If you have any questions please let us know.