Redmine - Feature #4298

An "internal" option for custom fields

2009-11-27 10:29 - Jan from Planio www.plan.io

Status:	Closed	**Start date:**	2009-11-27
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Custom fields	**Estimated time:**	0.00 hour
Target version:			
Resolution:	Invalid		

Description

When developing plugins, I often need persistent data attached to a user, a project, an issue, etc. Except for users, where I could technically exploit user_preferences from our plugins, I can't do anything but create my own migrations with custom tables or columns.

This way of doing it kind of does not feel right. Plugins should not create their own tables and columns in my opinion, unless they're really creating new entities that deserve their own model classes, too.

I've started using CustomFields for this which comes with a drawback: The values of these fields are publicly visible by everyone. In some cases it might only cause confusion or "not look so good" to have stuff displayed on a user or project page (done this in the PluginHoptoadServer plugin). In other cases it could be a security issue, when these fields simply cannot be displayed throughout Redmine.

I have 2 solution alternatives to this in mind:

1. create two new options for custom fields: being invisible (not displayed in views) and being "not editable" (not included in forms). This way enables three (four minus one existing) new usage scenarios:
   1. **invisible** & **not editable**: the field can be used internally by a plugin without interference by users
   2. **invisible** & **editable**: the field can be used by a plugin, but the plugin does not need to provide its own forms for updating the values
   3. **visible** & **not editable**: not sure about the added value of this. A plugin could provide its own way of manipulating custom field values, but data still gets displayed in views in the regular way.
   4. **visible** & **editable**: this is the default case which we have right now
2. simply create project_preferences, issue_preferences, etc. similar to user_preferences

I would be happy to take on this. I could easily make it a plugin, but it would monkey-patch the core and could break in future versions. I would rather see this go into the core, but I'd like to discuss it first rather than just submit a patch.

From an implementation perspective, I think this could be easily achievable using named (or even default) scopes on custom fields.

Let me know what you think.

Related issues:

Related to Redmine - Feature #1738: Add a "Visible" flag to project/user cust...	**Closed**	**2008-08-03**
Related to Redmine - Patch #1746: Patch for #1738; Add a "Hide on overview ta...	**Closed**	**2008-08-05**
Related to Redmine - Feature #6631: Provide security (e.g. display only) for ...	**Closed**	**2010-10-12**

History

#1 - 2010-01-24 11:57 - Jan from Planio www.plan.io

just bumping this up to get some feedback :) if nobody thinks this is useful, just let me know...

#2 - 2010-02-19 20:05 - Oren Laadan

Yes, such functionality will be very useful !

I'm using the Tab plugin (http://www.redmine.org/wiki/1/PluginTab) and it is annoying that the iframe information appears on the project overview. It looks so out of place that at first I thought it was a bug :(

Are you aware of these two: feature #1738, and related patch #1746 ?

#3 - 2011-08-31 10:26 - Oleg Aksenov

I, too, would benefit from this feature.
I hope this will work for all custom fields including "IssueCustomField".

#4 - 2013-10-01 15:57 - Toshi MARUYAMA

- Related to Feature #6631: Provide security (e.g. display only) for custom fields added

#5 - 2014-09-08 13:17 - Jan from Planio www.plan.io

- Status changed from New to Closed

- Resolution set to Invalid

Part of this was implemented in #1746. Furthermore, after almost 5 years of experience working with Redmine professionally, present time Jan disagrees with past time Jan. Migrations with own tables and columns are always better than trying to stuff things in magic custom fields... :-)

I am therefore closing this issue as invalid.