

Redmine - Feature #25043

Custom fields (type List): values inheritance, per project values

2017-02-10 12:38 - Adela Maba

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Custom fields	Estimated time:	0.00 hour
Target version:			
Resolution:			

Description

It would be nice to have possibility to define per project values of custom fields (that are List type), and also have possibility to inherit values from Project-level down to Issue-level and TimeLog-level.

Here is use case:

Consider having a project hierarchy that all deal with issues related to one or more Assets.

Every project has its own assets defined as **Asset** custom field at Project-level (type List).

In order to make connection of specific issue to one or more projects asset, custom field **Asset** on Issue-level must be created. That field is type List and inherits only selected values of custom field defined at owning project.

In order to track time that is spent on specific asset(s), **Asset** custom field of type List must be created at the TimeLog-level. That field inherits only selected values of custom field defined at owning issue.

How can that be done (just an idea)

The following examples are written in general manner, but in order to be applicable for above use case, only `##ReferencedCustomFieldName##` should be changed with `##Asset##`.

TimeLog-level custom field inherits possible values from parent issue

That can be done by defining special syntax that will be used in definition of Custom field values.

Here is example for custom field values defined at TimeLog-Level:

```
Value1  
Value2  
##ParentIssue##ReferencedCustomFieldName##
```

In this example TimeLog-level custom field can have multiple values (Value1, Value2), and also inherits values from custom field named "ReferencedCustomFieldName" at parent issue (indicated by `##ParentIssue##` indicator).

Issue-level custom field inherits possible values from parent project

Here is example of custom field values at Issue-level that are referencing values at parent project:

```
Value3  
Value4  
##ParentProject##ReferencedCustomFieldName##
```

In this example Issue-level custom field can have multiple values (Value3, Value4), and also inherits values from custom field named "ReferencedCustomFieldName" at parent project (indicated by `##ParentProject##` indicator).

Issue-level custom field inherits possible values from parent Issue

Also you might consider the possibility to inherit values in parent-child related issues, so child issue inherits selected values from parent issue.

Example:

Value5
Value6
##ParentIssue##ReferencedCustomFieldName##

In this example Issue-level custom field can have multiple values (Value5, Value6), and also inherits values from custom field named "ReferencedCustomFieldName" at parent issue (indicated by ##ParentIssue## indicator).

Note:

Above is only an example how that can be resolved. I suppose that there is a number of other ways to do that in better way.

Related issues:

Related to Redmine - Feature #21149: Linking "Possible values" for List custo...	New
Related to Redmine - Feature #25513: New custom field type for Project Level ...	New

History

#1 - 2017-02-15 10:47 - VD DV

This feature request is excelent. It is related to [#21149](#).

This means, for example while entering value at custom field (that are of type List), list of valid values are made dinamcaly according to union of static values (in your example "Value5", "Value6" ...) and values gathered through link fields (eg. ##ParentIssue##ReferencedCustomFieldName##).

Looking forward for this feature!

#2 - 2017-03-11 04:42 - Toshi MARUYAMA

- Related to Feature #21149: Linking "Possible values" for List custom fields to another custom fields of same type. added

#3 - 2017-05-04 04:06 - Toshi MARUYAMA

- Related to Feature #25513: New custom field type for Project Level (similar to "List type") - per Project specific set of values added

#4 - 2018-12-27 15:56 - VD DV

This issue is related to [#5127](#)

Do you have any plans to include this functionality to Redmine?