

Redmine - Feature #7849

custom issue relation types

2011-03-13 10:37 - Markus Valle-Klann

Status:	New	Start date:	2011-03-13
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Issues	Estimated time:	0.00 hour
Target version:			
Resolution:			
Description <p>In different projects I am using redmine with a number of custom trackers. To define relations between the tickets I would very much like to use other relations than the ones currently available.</p> <p>For instance, we have trackers for requirements and features. And we would like to be able to define an "implements" relation: feature implements requirement.</p> <p>The most sustainable solution would be to be able to define custom relation types much like custom fields. I searched for this on the redmine redmine but didn't find anything.</p> <p>From looking at source:trunk/app/models/issue_relation.rb it seems fairly straightforward to factor out the relation types and define them like custom fields, including symmetry relationships between the types etc.</p> <p>As we don't have any redmine development capabilities at the moment I hope more people would be interested to have that feature and somebody interested in implementing it :) With some guidance and advise from experienced redmine developers my team might also be able to contribute to the development.</p>			
Related issues:			
Related to Redmine - Feature #13690: Allow plugins to be able to add relation...		New	
Related to Redmine - Feature #15340: Custom field referencing other tickets		New	
Related to Redmine - Feature #16117: Add a new type of related issue: "exclus...		New	
Related to Redmine - Feature #18034: related issue improvements		Closed	
Related to Redmine - Feature #25384: Add new issue relation type: Obsoletes a...		New	
Related to Redmine - Feature #26786: Complex issue relation		New	
Related to Redmine - Feature #7629: Add "causes" and "caused by" issue relati...		New	2011-02-15
Has duplicate Redmine - Feature #11276: Missing related issues "regressed" an...		Closed	
Has duplicate Redmine - Defect #11311: New issue relationship		Closed	

History

#1 - 2012-03-07 15:12 - Johan Larsson

+1
Would also like to add custom issue relation types.

#2 - 2012-03-07 21:28 - Laurent Dairaine

+1

#3 - 2012-08-16 10:01 - Kelvin Chen

+1
Would like to have this function as well :)

#4 - 2012-08-31 17:15 - Matt Andrews

+1

#5 - 2012-09-16 16:14 - Adrián A.

+1

#6 - 2012-09-19 20:18 - Mauro Chojrin

+1. In my case, we use support tickets as QA tasks. It would be really helpfull to define relations such as "Tested in" (with simmetry in "Is tested by"). This feature combined with a custom workflow woudl definitely be a great improvement of my daily workflow.

#7 - 2012-10-18 16:57 - Fred Giusto

+1
That's a very good idea

#8 - 2012-12-01 13:03 - Bo Hansen

+1

#9 - 2013-01-25 14:05 - Tomas K

+1

#10 - 2013-04-10 07:15 - Dipan Mehta

+1. This one would be a great addition to many workflows.

#11 - 2013-05-31 13:10 - Lauren Copeland

+1 This feature would be useful.

#12 - 2013-07-01 20:43 - Brandon Liles

+1 Definitely agree. In our organization we assign review tickets for another developer to review our work. We currently use relations to track the review ticket in relation to the work ticket, but it would be nice to have a relationship type for this.

#13 - 2013-07-01 20:59 - Jeremy Thomerson

+1 I just found this as well and it would be great to be able to add symmetrical relation types either by plugins, or especially through the admin UI.

#14 - 2014-01-30 08:49 - Toshi MARUYAMA

- *Related to Feature #15340: Custom field referencing other tickets added*

#15 - 2014-02-27 18:15 - Mikhail Grinfeld

+1

#16 - 2014-10-06 12:45 - Toshi MARUYAMA

- *Description updated*

#17 - 2014-10-06 12:46 - Toshi MARUYAMA

- *Related to Feature #16117: Add a new type of related issue: "exclusive" added*

#18 - 2014-10-06 12:47 - Toshi MARUYAMA

- *Related to Feature #18034: related issue improvements added*

#19 - 2017-04-19 13:26 - Toshi MARUYAMA

- *Related to Feature #25384: Add new issue relation type: Obsoletes and Obsoleted_by added*

#20 - 2017-09-28 06:08 - Toshi MARUYAMA

- *Related to Feature #26786: Complex issue relation added*

#21 - 2018-03-31 07:10 - Go MAEDA

- *Related to Feature #7629: Add "causes" and "caused by" issue relationships added*

#22 - 2019-02-14 18:35 - Vito Marolda

+1 For this feature: our workflow would be "feature x implement request y". Anyway, a new type of custom field which accepts issues ([#15340](#)), with single or multiple choices, would also suit this need, maybe with better querying capabilities.

#23 - 2019-12-21 12:59 - Yasu Saku

+1

#24 - 2020-02-14 12:44 - Gunasekar R

+1

#25 - 2020-03-19 19:06 - shawn freeman

+1

It looks like many of the "relationship types" have specific meanings and probably corresponding code level behaviors. I can guess that this is why this isn't already something we can configure for ourselves, like states.

I would like to suggest that the scope of this RFE be focused only on the "Related To" relationship. Specifically:

1. Allow the admin to define any number of aliases for "Related To".
2. Present the list of "Related To" aliases in the drop-down as if they were each unique.
3. Leave the underlying code behavior as-is.

#26 - 2020-10-02 10:36 - Stefan Lindner

+1

#27 - 2020-10-23 08:22 - Fabien Durand

+1

#28 - 2021-07-23 16:40 - shawn freeman

Related to [#4799](#)?

#29 - 2021-08-06 08:03 - Denis Lebedev

+1

#30 - 2021-10-28 19:57 - Roberto Tavares

+1

#31 - 2022-07-09 14:04 - Daniel N

+1

#32 - 2022-11-03 16:48 - Giulio Quaresima

+1