# Redmine - Feature #11159

# **REST API for getting CustomField definitions**

2012-06-14 11:01 - Ruediger Schobbert

Priority:         Normal         Due date:           Assignce:         Jean-Philippe Lang         % Done::         0%           Estimated time:         0.00 hour           Target version:         2.4.0           Resolution:         Fixed           Description         Fixed           It would be helpful to have a possibility to query CustomFields for a certain type, since the customfield id and length/regex limitations might be necessary to create a valid REST API POST request.           One workaround to get the id of the customfield is to do a query for the resource first. E.g. for time_entries:           CET /Line_entries.aml           will also return the customfields.but only if there is at least one time_entry in the database           If there is no time_ontry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids.           But this will not work if any of the customField of.           It suggest something like:           GET /custom_fields.xml?type=TimeEntryCustomField           which will result in the following response: <custom_field id='!!"' name="*StartTime*"> <custom_field id='!!"' name="*StartTime*"></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field>	Status:	Closed	Start date:		
category:     Jean-Philippe Lang     % Done:     0%       Category:     REST API     Estimated time:     0.00 hour       Target version:     2.4.0     Estimated time:     0.00 hour       Description:     Fixed     Estimated time:     0.00 hour       Description:     Fixed     Estimated time:     0.00 hour       Target version:     2.4.0     Estimated time:     0.00 hour       Description:     Fixed     Estimated time:     0.00 hour       Target version:     2.4.0     Estimated time:     0.00 hour       Description:     Fixed     Estimated time:     0.00 hour       Torget version:     2.4.0     Estimated time:     0.00 hour       Description:     One workaround to get the id of the customfields to do a query for the resource first. E.g. for time_entries:       CET / time_entries.xnl     will also return the customfields, but only if there is at least one time_entry in the database       If there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids.       But this will not work if any of the customfields of peaces     Estimated time:       suggest something like:     CET / function_field.dis.nel?type=TimeEstryCustomField       Certion_fields.type=?timeEntryCustomField(types <custom_field dis="" type="?array*"> <custom_field< td=""><td></td><td></td><th></th><td></td></custom_field<></custom_field>					
Category:       REST API       Estimated time:       0.00 hour         Target version:       2.4.0       Resolution:       Fixed       0.00 hour         Description:       Fixed       0.00 hour       0.00 hour         Description:       Fixed       0.00 hour         Description:       Fixed       0.00 hour         Used to helpful to have a possibility to query CutsomFields for a certain type, since the customfield id and length/regex limitations might be necessary to create a valid REST API POST request.         One workaround to get the ld of the customfield is to do a query for the resource first. E.g. for time .entries:         CET /Line_entries.xml         will also return the customfields, but only if there is at least one time_entry in the database         If there is no time .entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids.         But this will not work if any of the customfields are mandatory.         This feature is probably related to Feature #9564         L suggest something like:         GET /custom_fields.xml?type=TimeEntryOustomField         which will result in the following response: <countom_field id="****&lt;/td">       name=**someListField***         <countom_field id="****&lt;/td">       name=**someListField***         <countom_field id="****&lt;/td">       name=**someListField***         <countom_field id="&lt;/th"><th>-</th><th></th><th>Due date:</th><th></th></countom_field></countom_field></countom_field></countom_field>	-		Due date:		
Target version:       2.4.0         Resolution:       Fixed         Description       It would be helpful to have a possibility to query CustomFields for a certain type, since the customfield id and length/regex limitations might be necessary to create a valid REST API POST request.         One workaround to get the id of the customfield is to do a query for the resource first. E.g. for time_entries:         CET /time_entries.xml         will also return the customfields, but only if there is at least one time_entry in the database         If there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids.         But this will not work if any of the customfields are monatory.         This feature is probably related to Feature #B664         I suggest something like:         CET /custom_fields.type="array"> <custom_field ic='type="array"'> <custom_field ic="&lt;/th"><th>Assignee:</th><th>Jean-Philippe Lang</th><th>% Done:</th><th>0%</th></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field>	Assignee:	Jean-Philippe Lang	% Done:	0%	
Resolution:       Fixed         Description         It would be helpful to have a possibility to query CustomFields for a certain type, since the customfield id and length/regex limitations might be necessary to create a valid REST API POST request.         One workaround to get the id of the customfield is to do a query for the resource first. E.g. for time_entries:         GET /time_entries.xml         will also return the customfields, but only if there is at least one time_entry in the database         If there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids.         But this will not work if any of the customfields are mandatory.         This feature is probably related to Feature #B664         Isuggest something like:         CET /custom_fields.type="array"> <custom_field id='11"' name="StartTime"> <custom_field id='11"' name="StartTime"> <custom_field id='11"' name="StartTime"> <custom_field id='21"' name="StartTime"></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field>	Category:	REST API	Estimated time:	0.00 hour	
Description         It would be helpful to have a possibility to query CustomFields for a cortain type, since the customfield id and length/regex limitations might be necessary to create a valid REST API POST request.         One workaround to get the id of the customfield is to do a query for the resource first. E.g. for time_entries:         CRT /time_entries.xml         will also return the customfields, but only if there is at least one time_entry in the database         If there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids. But this will not work if any of the customfields are mandatory.         This feature is probably related to Feature #9664         I suggest something like:         CET /custom_fields.type=TimeEntryCustomField         which will result in the following response: <custom_field id="1" name="StartTime"> <custom_field id="1" name="StartTime"> <custom_field id="2" name="SomeListPield"> <custom_field "="" id="2" name="fortartTime"> <custom_fields type="*array*"> <custom_field< cust<="" customfield<="" th=""><th>Resolution:</th><th>Fixed</th><th></th><th></th></custom_field<></custom_fields></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field>	Resolution:	Fixed			
<pre>might be necessary to create a valid REST API POST request. One workaround to get the id of the customfield is to do a query for the resource first. E.g. for time_entries: CET /time_entries.xml will also return the customfields, <b>but only if there is at least one time_entry in the database</b> If there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids. But this will not work if any of the customfields are mandatory. This feature is probably related to Feature #9564 I suggest something like: CET /custom_fields.xml?type=TimeEntryCustomField which will result in the following response: <custom_field id="1" name="StartTime"> <custom_field id="1" name="StartTime"> <custom_field< td=""><td>Description</td><td></td><th>I</th><td></td></custom_field<></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></custom_field></pre>	Description		I		
<pre>CET /time_entries.xml will also return the customfields, but only if there is at least one time_entry in the database if there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids. But this will not work if any of the customfields are mandatory. This feature is probably related to Feature #2064 I suggest something like: CRET /custom_fields.xml?type=TimeEntryCustomField which will result in the following response: <custom_field stype="array">         (custom_field id="1" name="StartTime"&gt;         (custom_field id="2" name="StartTime"&gt;         (custom_fi</custom_field></pre>					
<pre>will also return the customfields, but only if there is at least one time_entry in the database if there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids. But this will not work if any of the customfields are mandatory. This feature is probably related to Feature #9664 I suggest something like: GET /custom_fields.xml?type=TimeEntryCustomField which will result in the following response: <custom_fields type="array"></custom_fields></pre>	g				
<pre>W there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield dis. But this will not work if any of the customfields are mandatory. This feature is probably related to Feature #9564 I suggest something like: GBT /custom_fields.xml?type=TimeEntryCustomField which will result in the following response: <custom_field stype="array"></custom_field></pre>	GET /time_entries.xml				
But this will not work if any of the customfields are mandatory. This feature is probably related to Feature #9664 I suggest something like: GET /custom_fields.xml?type=TimeEntryCustomField which will result in the following response: <custom_field id="1" name="StartTime"> <custom_field id="1" name="StartTime"> <formative: <formative: <custom_field id="1" name="StartTime"> <formative: <formative: <custom_field id="1" name="StartTime"> <formative: <custom_fields: <custom_field id="1" name="StartTime"> <formative: <custom_field id="1" name="StartTime"> <formative: <custom_field id="1" name="StartTime"> <formative: <custom_field id="2" name="SomeListField"> <formative: <custom_field id="2" name="SomeListField"> <formative: <custom_fields <custom_fields< custom<br="" custom_field<=""><custom_fields> </custom_fields> </custom_fields<></custom_fields </formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></formative: </custom_field></custom_fields: </formative: </custom_field></formative: </formative: </custom_field></formative: </formative: </custom_field></custom_field>	will also return the cust	comfields, <b>but only if there is at least one t</b>	ime_entry in the data	base	
<pre>I suggest something like: GET /custom_fields.xml?type=TimeEntryCustomField which will result in the following response: <custom_field id="1" name="StartTime"></custom_field></pre>	If there is no time_entry yet, you can create a dummy one first (also via REST API), and delete that after you got the customfield ids. But this will not work if any of the customfields are mandatory.				
<pre>GET /custom_fields.xml?type=TimeEntryCustomField which will result in the following response:  <custom_field id="1" name="StartTime"></custom_field></pre>	This feature is probably related to Feature <u>#9664</u>				
<pre>which will result in the following response: <custom_fields type="array"></custom_fields></pre>	I suggest something like:				
<pre><custom_field id="1" name="StartTime"></custom_field></pre>					
Related to Redmine - Defect #7819: REST API Populating issue field enumeratio New 2011-03-09	<pre><custom_field <type="" i="">TimeEnt     <format>Text&lt;     <min_length>5     <max_length>5     <regex>[0-2][     <default_valu <required="">1<!--/     </custom_field-->     <custom_field <type="" i="">TimeEnt     <format>List&lt;     <multiple>0<!--/     <possible_val         <value-->Fi         <value>Fi         <value>Se     0<!--/     </custom_field--></value></value></multiple></format></custom_field></default_valu></regex></max_length></min_length></format></custom_field></pre>	<pre>d="1" name="StartTime"&gt; ryCustomField /format&gt;  0-9]:[0-5][0-9] e&gt; required&gt; d="2" name="SomeListField"&gt; ryCustomField /format&gt; multiple&gt; ues&gt; rst cond lues&gt; e&gt; required&gt;</pre>			

# Associated revisions

# Revision 12165 - 2013-09-29 12:08 - Jean-Philippe Lang

REST API: custom fields definition (#11159).

#### History

### #1 - 2012-07-11 00:10 - Ruediger Schobbert

Since this probably takes a while, until it is integrated in redmine, I wrote a little webapp (in groovy, because I only have minimal ruby/redmine knowhow) to serve the customfields.

If anyone is interested they may have a look on github: https://github.com/rschobbert/redmine-customfield-info

## #2 - 2012-07-24 00:12 - Ruediger Schobbert

The project redmine-customfield-info on github has been deprecated in favor of the new project redmine-miss-api, which also serves the enumerations. The project can be found on github under: <a href="https://github.com/rschobbert/redmine-miss-api">https://github.com/rschobbert/redmine-miss-api</a>

#### #3 - 2013-03-29 17:51 - Stéphane Klein

+1

#### #4 - 2013-08-30 10:28 - Jaap de Haan

+1 too, as filtering is possible on the issues with (nnn being the field id),

#### GET /issues.xml?cf\_nnn=abcdef

there should be at least a way to programmatically figure out what are the fields and which ids they have. (At the moment there is only a dirty workaround by examinating the contents of a ticket, by hoping the wanted fields are present)

My proposal would be to extend the project query by adding one include option (beside issue\_categories and trackers also **fields** for example) For a first shot this does not need to be too complicated, having the field names, ids, tpyes and if they are required or not would be already great.

#### #5 - 2013-09-09 01:45 - Toshi MARUYAMA

- Related to Defect #7819: REST API Populating issue field enumerations + Issue list filters added

## #6 - 2013-09-24 14:32 - Vincent Caron

A patch for 'GET /custom\_fields.xml' (without the filtering part) is available in #9664.

#### #7 - 2013-09-29 11:59 - Jean-Philippe Lang

- Related to Feature #9664: CRUD operations for "custom field definitions" (not setting custom fields on issues!) added

#### #8 - 2013-09-29 12:19 - Jean-Philippe Lang

- Subject changed from REST API query CustomField Definitions to REST API for getting CustomField definitions
- Status changed from New to Closed
- Assignee set to Jean-Philippe Lang
- Target version set to 2.4.0
- Resolution set to Fixed

Feature added in r12165. The response is slighty different from the proposal, see Rest CustomFields.

## #9 - 2013-11-14 23:04 - Alex Last

what about CRUD for custom field definitions (http://www.redmine.org/issues/9664) ?