

## Redmine - Feature #2084

### Customizable "protocols" to generate external links

2008-10-24 11:56 - Erwan Ducroquet

|  |                                  |
|--|----------------------------------|
| <b>Status:</b> New   | <b>Start date:</b> 2008-10-24    |
| <b>Priority:</b> Normal  | <b>Due date:</b>                 |
| <b>Assignee:</b>   | <b>% Done:</b> 0%                |
| <b>Category:</b> Wiki  | <b>Estimated time:</b> 0.00 hour |
| <b>Target version:</b>   |                                  |
| <b>Resolution:</b>   |                                  |
| <b>Description</b><br>Allows to define "prototypes" along with an URL prefix, so it will be easy in markup to reference external resource without needing a copy/paste of a full URL<br><br>Example with proto='mywiki' ; URL prefix="http://wiki.company.com/display/space/" :<br><br>markup : "Check This":mywiki:somewikipage<br>formatted into : <a href='http://wiki.company.com/display/space/somewikipage'>Check This</a><br><br>Other examples :<br><br>"Great language":wp:Ruby_(Programing_Language)<br><a href='http://en.wikipedia.org/wiki/Ruby_(Programing_Language) ' >Great language</a><br><br>:wp:Redmine<br><a href='http://en.wikipedia.org/wiki/Redmine'>Redmine</a><br><br><i>note:</i> also same feature asked for CustomFields |                                  |
| <b>Related issues:</b>   |                                  |
| Related to Redmine - Feature #2083: CustomField of type "external-link-to" wi...   | <b>Closed</b> <b>2008-10-24</b>  |
| Related to Redmine - Feature #2953: application launch URLs not rendered as ...  | <b>New</b> <b>2009-03-11</b>     |

### History

#### #1 - 2008-10-24 11:57 - Erwan Ducroquet

see also [#2083](#)

#### #2 - 2008-10-24 14:48 - Erwan Ducroquet

For info : I've currently hardcoded such feature by adding a method in wiki\_formatting.rb, but it's not very pretty ...

```
##-- Erwan Ducroquet 2008-10-23 generate links depending on proto
def inline_specific_link_do(all, start, proto, url, title)
  case proto
  when 'stp'
    "#{start}<a class='external' href='http://wiki.fr.world.socgen/display/STP/#{url}'>#{title}</a>"
  when 'ccl'
    "#{start}<a class='external' href='http://srvparstp20.fr.world.socgen/cc/view,label,#{url}'>#{title}</a>"
  when 'wp'
    "#{start}<a class='external' href='http://en.wikipedia.org/wiki/#{url}'>#{title}</a>"
  when 'google'
    "#{start}<a class='external' href='http://www.google.com/search?q=#{url}'>#{title}</a>"
  when 'file'
    "#{start}<a class='external' href='file://#{url}'>#{title}</a>"
  else
    all
  end
end
```

```
##-- Erwan Ducroquet 2008-10-23 add own way to handle specific URL prototypes : STP wiki links, files, ClearCas
```

```
e source, Rules, ext Reviews
#-- format is :repository:resource or "title":repository:resource
def inline_specific_link(text)
  text.gsub!(/\"([^\"]\n+)\"\":([a-z]+)\:([\w_\//\\\:\-\\+\(\)]+)/) do
    inline_specific_link_do( $&, '', $2, $3, $1)
  end
  text.gsub!(/\:([a-z]+)\:([\w_\//\\\:\-\\+\(\)]+)/) do
    inline_specific_link_do( $&, '', $1, $2, $2)
  end
end
end
```