Redmine - Feature #2448

Graphviz of ticket dependencies (with example)

2009-01-06 15:56 - Matthew W

Status:	New	**Start date:**	2009-01-06
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Resolution:			

Description

I found the output of this (admittedly very hacky, but conceptually quite simple) script a lot more useful than the gannt chart.

Would it be possible to build a feature like this into redmine, generating a graphviz of dependencies between open tickets on any given project or version?

```
#!/bin/sh

(
echo "digraph redmine {";

mysql -N -s -uredmine --password= -Dredmine -e "select id,subject from issues where status_id != 5
 and fixed_version_id = 1" | ruby -e 'puts STDIN.map {|x| x.split(/\t/)}.map {|id,title| "#{id} [l
abel=\"#{id}: #{title.chomp.gsub(/((?:[^ ]+ ){4})/, "\\1\\n")}\"]"}';

mysql -N -s -uredmine --password= -Dredmine -e "select issue_from_id, issue_to_id from issue_relat
ions ir join issues i1 on ir.issue_from_id = i1.id join issues i2 on ir.issue_to_id = i2.id where
relation_type = 'precedes' and i1.status_id != 5 and i2.status_id != 5 and i1.fixed_version_id and
 i2.fixed_version_id = 1" | ruby -e 'puts STDIN.map {|x| x.split(/\t/)}.map {|id1,id2| "#{id1} ->
#{id2}"}';

echo "}"
) | dot -Tpng > /var/www/space/redmine_graph.png
```

Related issues:

Related to Redmine - Feature #279: issue dependencies	**Closed**
Related to Redmine - Feature #559: Workflow Enhancements	**New**
Related to Redmine - Feature #12647: issue relations network view	**New**

History

#1 - 2009-04-06 06:20 - Frederic Morin

can you provide (attach) a sample graph produced by your query ?

#2 - 2009-04-06 12:09 - Matthew W

Frederic Morin wrote:

> can you provide (attach) a sample graph produced by your query ?

I can email you one if you want? It's essentially just a directed graph with nodes for open tickets (with ID and title) and edges for 'precedes' dependencies. Kind of a critical path diagram, although adding estimates to it would help with that.

#3 - 2009-04-06 12:23 - Jean-Baptiste Barth

Matthew W wrote:

> I can email you one if you want?

I think you should attach a sample here, more than one person would be interested in seeing that..

#4 - 2009-04-06 12:43 - Matthew W

> I think you should attach a sample here, more than one person would be interested in seeing that..

Unfortunately it's a private project, not like super top secret or anything but at the same time not sure we want our dev stuff up here in all eternity. So probably best if you email me for it, sorry to be a pain.

#5 - 2009-04-06 13:46 - Jean-Baptiste Barth

- File redmine_graph.png added

OK, no problem. Thanks for having emailed it !

I tried to ruby-ise your code so it uses ActiveRecord and Redmine objects, it will be easier to adapt for other people (maybe it's not totally equivalent, I didn't keep the "fixed_version_id = 1") :

```
#!/bin/sh

(
echo "digraph redmine {";

ruby script/runner 'Issue.find(:all).select{|i| !i.closed? }.map{|i| puts "#{i.id} [label=\"#{i.id}: #{i.subje
ct.chomp.gsub(/((?:[^ ]+ ){4})/, "\\1\\n")}\"]"}';

ruby script/runner 'IssueRelation.find(:all, :include => [:issue_from, :issue_to]).select{|ir| ir.relation_typ
e == "precedes" && !ir.issue_from.closed? && !ir.issue_to.closed? }.map{|ir| puts "#{ir.issue_from_id} -> #{ir
.issue_to_id}"}';

echo "}"
) | dot -Tpng > /var/www/redmine_graph.png
```

It works well, I attach a sample result. I obtained it by replacing the first ruby line by :

```
ruby script/runner 'Issue.find(:all).select{|i| !i.closed? }.map{|i| puts "#{i.id} [label=\"#{i.id}: My issue
#{i.id}\"]"}';
```

Maybe your sample with this line would be better ? If so, delete mine and put your own..

Anyway, it may be difficult to integrate such a feature in core, since it's based upon an external program (graphviz)... but I may be wrong!

#6 - 2009-04-06 14:20 - Jean-Baptiste Barth

Here is a complete ruby/rake solution :

```
namespace :redmine do
  task :graphviz => :environment do
    f = Tempfile.new('graphviz')
    f.puts "digraph redmine {"
    Issue.find(:all).select do |i|
      !i.closed?
    end.map do |i|
      f.puts "#{i.id} [label=\"#{i.id}: #{i.subject.chomp.gsub(/((?:[^ ]+ ){4})/, "\\1\\n")}\"]"
    end
    IssueRelation.find(:all, :include => [:issue_from, :issue_to]).select do |ir|
      ir.relation_type == "precedes" && !ir.issue_from.closed? && !ir.issue_to.closed?
    end.map do |ir|
      f.puts "#{ir.issue_from_id} -> #{ir.issue_to_id}"
    end
    f.puts "}"
    f.flush
    IO.popen("dot -Tpng -o /var/www/redmine_graph.png #{f.path}")
    f.unlink
  end
end
```

You can register it in "lib/tasks/graphviz.rake" and call it with "rake redmine:graphviz".
Any opinion about that ?

#7 - 2009-04-06 14:28 - Matthew W

Jean-Baptiste Barth wrote:

> Here is a complete ruby/rake solution :
> [...]You can register it in "lib/tasks/graphviz.rake" and call it with "rake redmine:graphviz".
> Any opinion about that ?

Nice, yeah that looks a lot cleaner than my shell script.

A couple of gotchas in that script are: it's doing some really crude word-wrapping on issue titles:

{{{
.gsub(/((?:[^]+){4})/, "\\1\\n")
}}}

because I couldn't figure out how to make graphviz do this - but maybe there's a better way. Also the label needs escaping for graphviz incase it contains a closing quote.

Imagine you might wanna customize where it puts the .png output too.

Cheers for tidying this up though, glad it's of use!

#8 - 2009-04-06 14:40 - Matthew W

Re dependencies, I don't think it'd hurt to have a rake task in there with a runtime dependency on graphviz - it could warn people when run if graphviz isn't installed, but wouldn't affect the smooth running of anything else.

#9 - 2009-04-06 16:21 - Jean-Philippe Lang

Here is a quick conversion to a controller that can be used inside a plugin:

```
class GraphvizController < ApplicationController
  unloadable

  def graph
    f = ""
    f << "digraph redmine {\n"
    Issue.find(:all).select do |i|
      !i.closed?
    end.map do |i|
      f << "#{i.id} [label=\"#{i.id}: #{i.subject.chomp.gsub(/((?:[^ ]+ ){4})/, "\\1\\n")}\"]\n"
    end
    IssueRelation.find(:all, :include => [:issue_from, :issue_to]).select do |ir|
      ir.relation_type == "precedes" && !ir.issue_from.closed? && !ir.issue_to.closed?
    end.map do |ir|
      f << "#{ir.issue_from_id} -> #{ir.issue_to_id}\n"
    end
    f << "}\n"

    png = nil
    IO.popen("dot -Tpng", "r+") do |io|
      io.binmode
      io.write f
      io.close_write
      png = io.read
    end
    send_data png, :type => 'image/png', :filename => 'graph.png', :disposition => 'inline'
  end
end
```

No file is written on disk.
I just converted the script but some optimizations can be done when retrieving issues...

#10 - 2009-04-06 18:09 - Jean-Baptiste Barth

Jean-Philippe Lang wrote:

> Here is a quick conversion to a controller that can be used inside a plugin:
>
> [...]
>
> No file is written on disk.
> I just converted the script but some optimizations can be done when retrieving issues...

Thanks ! Actually I was wondering how to do that without a Tempfile, it's a perfect example...
And indeed, I didn't try to optimize it, it was just to show how it could be integrated to a rake task or something else in the app.

#11 - 2010-08-08 18:29 - Norbert Melzer

Is there already a plugin the code from Jean-Philippe Lang?
I use redmine to keep track over non ruby (on rails) projects, so I really dont know how to really do anything ruby related.

Really cool would be if I could set different filters, and if closed issues that are blocking ore preceeding other issues would be shown also, perfectly in another color. Different arrow-types for blocking, duplicate, preceeding would be awesome too :D

#12 - 2010-11-13 14:09 - Hauke Heibel

Since we cannot vote on features (at least not that I am aware of), I am writing this post to state that I would love to see the Graphiviz feature too!

#13 - 2011-01-21 14:58 - Sascha Herrmann

+1

#14 - 2012-05-12 10:06 - Terence Mill

You could use redmine-wiki-external-filter-plugin to integrate this external program call when writing a <macro>show_issue_graph</macro> on a wiki seite where this png file will be embedded instead.

#15 - 2012-10-31 02:51 - Matt Palmer

I've just created a plugin based on the controller code Jean-Philippe posted in #9. It's available on Github at https://github.com/mpalmer/redmine_issue_dependency_graph. At the moment it only does a graph of issues in a particular version, but I might go nuts one day and write a full "show me all issues related to this one" tree graph.

#16 - 2013-02-12 23:45 - Terence Mill

related:

issue relations network view
Workflow Enhancements -Part2

#17 - 2014-08-27 17:14 - Santiago Pérez

Matt Palmer wrote:

> I've just created a plugin based on the controller code Jean-Philippe posted in #9. It's available on Github at https://github.com/mpalmer/redmine_issue_dependency_graph. At the moment it only does a graph of issues in a particular version, but I might go nuts one day and write a full "show me all issues related to this one" tree graph.

Hi Matt,

I tried to install your plugin with the actual redmine (2.5.2) with no success (I have no error on terminal when installing, but nothing appears on the plugin administrator tab on redmine). Have you tried to install it in new redmine versions?

Your plugin is the only one I have found that graph the dependencies between the issues, really valuable. Is there another one now around?.

Thanks in advance,

Santiago

#18 - 2014-11-08 09:31 - Laurent Dairaine

+1

#19 - 2015-02-17 13:51 - Dawid Grzegorczyk

Hi,
I successfully migrated this plugin to Redmine 2.3.1. My pull request is available here:
https://github.com/mpalmer/redmine_issue_dependency_graph/pull/2
Best of whishes,
Zły Kapitan

#20 - 2015-09-07 06:25 - Toshi MARUYAMA

- Related to Feature #12647: issue relations network view added

#21 - 2016-11-30 14:00 - Kamran Soomro

Hi Matt,

I was wondering if your plugin has been updated for the latest version of Redmine. If so, could you please send it to me?

Matt Palmer wrote:

> I've just created a plugin based on the controller code Jean-Philippe posted in #9. It's available on Github at https://github.com/mpalmer/redmine_issue_dependency_graph. At the moment it only does a graph of issues in a particular version, but I might go nuts one day and write a full "show me all issues related to this one" tree graph.

Files

redmine_graph.png	68 KB	2009-04-06	Jean-Baptiste Barth