

Redmine - Feature #1518

Log time via commit messages

2008-06-23 23:48 - Jonas von Andrian

Status: Closed	Start date: 2008-06-23
Priority: Normal	Due date:
Assignee: Mischa The Evil	% Done: 50%
Category: SCM	Estimated time: 0.00 hour
Target version:	
Resolution:	
Description	
Hello	
I use the log time feature quite often and am getting used to insert the time in my commit comments.	
So i got the idea if it would be possible to automatically log the time if a ticket is referenced.	
The syntax may be:	
<pre>refs #213 time 1.5 the rest of my message</pre>	
So ticket 213 gets a log with "the rest of my message"	
If there are multiple references the time is equally distributed	
Perhaps it would make sense to create an extra keyword for the log message too	
<pre>refs #123 time 1,5 log short log message the rest of my commit message</pre>	
If someone would be kind enough to point me in the right direction i am more than willing to do this myself, because it would save me time in the long run.	
Jonas	
Related issues:	
Related to Redmine - Feature #2639: Easier time reporting	Closed 2009-02-01
Related to Redmine - Feature #1088: Time logging via commit messages	Closed 2008-04-22
Related to Redmine - Feature #1232: Referencing and fixing issues in commit m...	New 2008-05-13
Related to Redmine - Feature #4155: Automatic spent time logging from commit ...	Closed 2009-11-02

History

#1 - 2008-06-24 00:29 - Eric Davis

I wouldn't use this feature myself but as long as it has an option to disable it I'm happy.

A commit is called a Changeset in Redmine and the code that looks for issue ids in the commit message in `scan_comment_for_issue_ids` in <http://www.redmine.org/repositories/entry/redmine/trunk/app/models/changeset.rb>

#2 - 2008-06-24 09:08 - Jonas von Andrian

I think we could do without a disable option, if the syntax is strict enough.

Anyways, is there a settings api or "the" way to add settings?

Thanks for your quick reply. I will look into it next week.

#3 - 2008-06-24 17:59 - Eric Davis

Jonas von Andrian wrote:

I think we could do without a disable option, if the syntax is strict enough.

Maybe. I like how the repository settings ones allow someone to enter the syntax.

Anyways, is there a settings api or "the" way to add settings?

No, you will have to just add the fields to the settings page. That parts easy to do.

#4 - 2008-07-03 19:41 - Jonas von Andrian

- File *patch_commit_messages.diff* added

- Status changed from *New* to *Resolved*

- Target version set to *0.8*

- % Done changed from *0* to *80*

Here is the patch. I added an option to disable this feature completely. And you can specify which keywords to use.

You can now log time and change the ratio of a ticket via commit messages. The comment for the log is everything but the keywords+value

```
refs #1
time 3,4
ratio 40
```

```
comment for this ticket
```

```
#2 #if you specified * as the ref keyword
time 30 m
ratio 30%
```

```
comment for second ticket
```

This results in time logs with "comment for this ticket" as comments

All of the time formats are supported except 12 hours and 30 min(utes)

I added translations for en and de. And I added tests for the new features and for some old ones to avoid bugs.

I hope this gets added.

Jonas

#5 - 2008-07-03 20:07 - Jonas von Andrian

- File *patch_commit_messages2.diff* added

Small fix

```
before_save > before_create
```

#6 - 2008-07-03 21:44 - Jonas von Andrian

- File *patch_commit_messages3.diff* added

another update:

- Time entries are given the correct time
- Prevents duplicate time entries

Rereading the repo will probably be an issue with this feature. So perhaps the best would be to disable advanced keywords upon repo initialization.

I have no idea how to do this. Pointers are welcome.

#7 - 2008-07-05 17:52 - Jonas von Andrian

Sorry, but this isn't working on my server.
I will have to look into it further

#8 - 2008-07-06 18:38 - Jonas von Andrian

- File `patch_commit_messages4.diff` added

It's done. This time successfully tested in a live environment.

The error was hidden in an AR bug.

#9 - 2008-09-24 00:00 - Mischa The Evil

I finally had some time to apply this to recent trunk ([r1900](#)) and test it to see if it's usable for me. Here a few comments/notes:

At first I must say that IMHO this patch really rocks. After using Redmine together with SVN for awhile now I found it un-handly to log time on an issue which had been auto-closed (in my workflow resolved) by an SCM-commit. This patch provides me an option to include the spent-time immediately on the commit-message.

I will later test this on my private vendor-branch with the [start/end-time tracking](#) patch applied to see if it can exist (not work!) together without conflicting eachother.

Two sidenotes though:

1. this patch expects the order of the parts of the commit-message to be:

```
refs #1
time 3,4
ratio 40

comment for this ticket

#2 #if you specified * as the ref keyword
time 30 m
ratio 30%

comment for second ticket
```

to fully work with the patch (especially the timelog-comment).

p.
Because of the fact that I'm trying to keep the revision-log as tight and readable as possible I definately want to start (at least the first line) the commit-message with the same line which should be used as the timelog-comment. Then (optional) a more detailed description of the commit, followed by a block of keywords+values.

It would be great if the commit-message can be made-up like:

```
comment for this ticket

refs #1
time 3,4
ratio 40

comment for second ticket

#2 #if you specified * as the ref keyword
time 30 m
ratio 30%
```

without screwing-up the timelog-comment (i've tested the alternative order already and it came out not-working).

2. when only time is logged using the commit-message without changing the done-ratio **no** journal-message should be added to the issue. Currently the patch writes a blank journal only displaying the resolved `:text_status_changed_by_changeset` (Applied in changeset rxxx). IMHO a journal-message should be added only when the done-ratio is changed.

#10 - 2008-09-24 00:40 - Mischa The Evil

Oh, I forgot a third (more important) sidenote:

When multiple issues are referenced, the keyword:value-changes are only applied to the latest referenced issue. They are completely ignored on the first referenced issue (when referencing two issues in total).

I think it would be good to clearly define how multiple references should be handled. I think it's best to define the handle as follows:

- time should be equally distributed (thus with two referenced issues "time 1" in commit-message should be distributed into 0.5 per issue)
- done-ratio should be applied according to the keyword-value for all referenced issues the same (thus with two referenced issues "ratio 30%" in commit-message should set done-ratio to 30% **on both** issues)

And a little addition on my first sidenote:

This could be solved by adding a new keyword log like you've proposed in the initial issue description. Then the whole block can be appended to the commit-message instead of requiring it to be prepended.

I'll look into this in the upcoming weeks. If I have 'something' I'll post it here...

#11 - 2008-09-24 00:54 - Mischa The Evil

Now... I again, forgot one point:

It would be a good thing to add a setting for the timelog activity-type so it becomes configurable which activity-type is used for timelogs added through a commit-message.

This could be implemented by modifying the activity_id-method in `../app/models/changeset.rb`. I'll look into it and post any results... ;)

#12 - 2008-09-25 02:17 - Chris Platts

I'm really interested in this patch - will try to apply it tomorrow for some testing! However, Mischa The Evil's comments show it's not quite 'there' ... (still great work!)

Perhaps the time, ratio and (later) activity type values in the commit log could be tokenised somehow, in much the same way that 'references' and 'closes' links are currently implemented. This way, the values could be entered anywhere in the text and still be understood by the parser. Perhaps the issue references could be 'piggy-backed' which would ensure time, ratio and activity values are associated with the correct issue when > 1 issues are referenced.

For example,

```
- Fixes nasty crash in background thread - fixes #121 {1.30:50%:Bugfixing}
- Looked at how enhancement #544 {0.30:10%:Planning} should be implemented
```

... or something similar. Of course, the timelog block {...} would not be included in the log's text. The parser's logic would be something like:

- Have I found an issue reference? Yes.
- Is there a {...} block after it? Yes.
- Look at the content inside the {...} block and attempt to split it into parts by the : delimiter
- Look at each part - can I parse it as a time (e.g. 1.30 or 1,30 or 1.5) or can I parse it as an integer between 0 and 100 or is it a text string?
- Apply found attributes to the timelog.

Of course, there's much to consider regarding the choice of attribute delimiter (a colon in my examples) and the block delimiter (curly-braces in my example). Straight away, it's clear that some cultures/users would want to use : as the HH:MM delimiter. Then there's the fact that mainland Europe tends to use , as the decimal separator, whilst the UK and USA use a full-stop (period). Finally, I'm sure some C programmers would take issue with using { and } :)

Perhaps this could be a configuration setting in the same way that the issue reference/closes tokens are set?

Oh, and one last thing - I'm unsure how the times are expressed in these examples. Am I right in thinking the ',' is being used as a decimal separator (I'm in the UK). In that case, what's the convention for how the second part of the figure is used? For example, is 1,5 1 and a half hours, or is it 1 hour, 50 minutes? Might I suggest that whilst 1.5 = 1hr 30mins might be fine for us, less technical users might want to express time in the 'natural' style of 1.30 or 1:30 -- or even 1h30.

Regards,
Chris

#13 - 2008-09-25 02:25 - Chris Platts

Heh - spot the deliberate mistake :) The first example shows a bug being closed, yet it also specified 50% as the progress. In reality, the 'fixes' marker might, depending on the Redmine configuration automatically set the issue to 100%. So, the example should read something like:

```
- Fixes nasty crash in background thread - fixes #121 {1.30:Bugfixing}
```

The parser would see that no percentage has been specified, but still picks up the time and activity type. :)

#14 - 2008-10-06 02:12 - Mischa The Evil

- File `logtime_changeratio_via_commit-message-r1926.diff` added

- File `logtime_changeratio_via_commit-message_extended-r1926.diff` added

Motives

Jonas von Andrian wrote:

- File [patch_commit_messages4.diff](#) added

It's done. This time successfully tested in a live environment.

Mischa The Evil wrote (abstraction):

...there exists some issues with this patch...

Ok... I now have modified/extended this patch to solve my above quoted issues. Sharing it here for ppl who suffers the issue(s) too...

What's done?

proces-description:

I started with the rebasing of the quoted diff to the trunk (currently [r1926](#)) and then I started to make the desired modifications. I'll attach both versions for clarity.

changelog:

Here comes the changelog, with excerpts of my postings describing the issues prepended where available, against the quoted patch:

commit-message parts-ordering (possible use of log keyword) (<http://www.redmine.org/issues/show/1518#note-9>)

- added configurable timelog-comment keyword (default setting is timelogcomment)
- fixed unit-test "test_log_time_splits_the_time_equally" after timelog-comment keyword implementation

issue-journal message added only when done-ratio is changed (<http://www.redmine.org/issues/show/1518#note-9>)

- changed issue-journal creation behaviour in commit-message processing; now a journal is only created when:

```
- [core] an issue is fixed using fixing-keywords (thus status (AND possibly done-ratio) is changed)
```

OR

```
- [ext] advanced commit parsing is active AND  
      issue is not fixed using core fixing-keywords AND  
      issue-ratio is changed explicitly in commit-message using ratio-keyword (this requires ref-keyword to be configured explicitly)
```

OR

```
- [ext] advanced commit parsing is active AND  
      issue is not fixed using core fixing-keywords AND  
      * is configured as reference-keyword AND  
      issue-ratio is changed explicitly in commit-message using ratio-keyword
```

configurable default activity-type for timelogs created using commit-messages (<http://www.redmine.org/issues/show/1518#note-11>)

- added (configurable) default timelog activity-type setting

Extensions I've made "myself":

- added default timelog-comment setting if time is added in commit-message but timelog-comment is missing (blank [default] or changeset-ID)
- changed used issue-journal text when only an issue's done-ratio is changed using ratio-keyword in commit-message; The text is now as follows:
 - when only an issue's done-ratio is changed: "Partially applied in rxxx"
 - when an issue is fixed using fixing-keywords: "Applied in rxxx"
- changed order of advanced commit parsing settings-view to reflect changes made in this patch

(new) syntax:

```
refs #1  
ratio 40  
time 1,0  
timelogcomment comment for this ticket
```

...multi-paragraph...

...description...

AND

```
#2 #if you specified * as the ref keyword
ratio 30%
time 30 m
timelogcomment comment for second ticket

...commit-description...
```

AND

```
...one-liner which is used as description-line revisionlists in most svn-clients...

...multi-paragraph...
...extensive...
...description...

refs #1
ratio 40
time 3,0
timelogcomment comment for this ticket
```

unchanged issues:

multiple issue-references handling (<http://www.redmine.org/issues/show/1518#note-10>)

This issue was mistakenly noted by me. I wasn't using correct syntax then.

Though, one thing should be documented. After some extensive testing I have found that a commit-message like below is not giving the expected result (depending on setting `ref_keywords[refs]` && setting `fix_keywords[closes]`):

```
oneliner describing the changes

refs #1 and also closes #2
ratio 50
time 2,0
timelogcomment comment for this ticket
```

This results in the following wrong behaviour:

- the keyword:value-changes are only applied to the latest *referenced* issue
- they are completely ignored on the first *referenced* issue (when *referencing* two issues in total).

The conclusion that should be taken is that it is **not possible** to mix-up references **and** closures using **shared time-/ratio-keywords** while using this patch. It is valid and working in the core though, so this is a distinction in syntax-behaviour brought by this patch.

So in this case the following commit-message should be used:

```
oneliner describing the changes

refs #1
ratio 50
time 1,0
timelogcomment comment for this timelog

closes #2
ratio 50 <-- this is omitted when fixing an issue -->
time 1,0
timelogcomment comment for this other timelog
```

Note: It is **possible** though to use multiple references like below:

```
oneliner describing the changes

refs #1, #3 & #4
ratio 50
time 3,0
timelogcomment Work for (unfixed) referenced issues #1, #3 and #4; one hour each.
```

```
closes #2 & #5
ratio 50      <-- this is omitted when fixing an issue -->
time 2,0
timelogcomment Work for fixed issues #2 and #5; one hour each.
```

The result is that:

- issues [#1](#), [#3](#) and [#4](#) are referenced, all three have added a **one**-hour timelog and all three have the done-ratio set to 50%.
- issues [#2](#) and [#5](#) are closed, both have added a **one**-hour timelog and both have the done-ratio set to 50%.

tests:

I've made sure that I didn't broke the tests provided by the original patch. I've changed one test according to the addition of the timelogcomment-keyword.

After this patch applied all tests pass...

Though I haven't been able to come-up with some good tests for my other additions. I will try to take a look at them lateron...

Finalisation

I hope someone can profit (in use) from this extended patch. I also hope that patched features will become available in the core in the future. Though, looking at my skills, I don't think this is the best implementation... :-)

If someone has further questions let me know... ;-) HTH...

#15 - 2008-10-06 02:22 - Mischa The Evil

Chris Platts wrote:

Perhaps the time, ratio and (later) activity type values in the commit log could be tokenised somehow

This looks a good option to me. Though I guess it will require more changes in the code.

For now I've modified the patch to solve my issues. See above... :-)

For example, is 1,5 1 and a half hours, or is it 1 hour, 50 minutes?

1.5 equals 1 and a half hours. This is default Redmine behaviour. See here: http://www.redmine.org/wiki/redmine/Time_tracking

Note though that the time-formats accepted by this patch are limited to All except 12 hours and 30 min(utes).

#16 - 2008-10-06 18:58 - Jonas von Andrian

Thank you Mischa for your great Input and your extended Patch. Especially your work on the journal seems useful.

In about a week I will have time to work on it again and look at your changes in detail. It looks like you finished what I started out to do. 2 months in India and voila, my work is finished.

I'm not sure I like the tokenising Idea though. Seems a bit too condensed for my taste.

#17 - 2008-11-11 10:35 - Jean-Philippe Lang

- Target version deleted (0.8)

#18 - 2008-12-09 04:51 - Kihyun Yun

Hi, Mischa.

Can you update your patches for 0.8rc1 or recent trunk version?

Thanks a lot.

#19 - 2008-12-10 01:52 - Mischa The Evil

- Status changed from Resolved to New

Hans Yoon wrote:

Can you update your patches for 0.8rc1 or recent trunk version?

Thanks for pointing this out. I did not noticed it since I'm +/- 400 commits behind current trunk... :-S

I'll try to update the patch to be/stay compatible with the 0.8.0-stable branch. I have already tracked-down the conflicting code-changes in the trunk. It's broken by the user-mapping implementation in [r2006](#).

Probably this weekend I'll have a working patch. I already have it working again but are testing it to be fully compatible with the user-mapping. Some current issues:

- I noticed that a new user is made in the DB as anonymous without login (it isn't showed neither). Currently I can't remember if that was also the case with my patch against [r1926](#). Can you confirm?
- How to handle the anonymous commits at all (user is not Redmine-user and not mapped to a Redmine-user) after the user-mapping implementation?
- All the core-tests must be made at least compatible with this patch.
- Some further general tests...

Greetings,

Mischa.

#20 - 2008-12-10 02:31 - Mischa The Evil

- File *logtime_changeratio_via_commit-message_extended-r2118-Alpha.diff* added

I have wrapped-up the working **alpha** version against [r2118](#) of the trunk for the "edge-addicts". This is without the test-changes. I'll look at that later. Please only test on a dedicated, non-production deployment of Redmine [r2118](#).

I'll finish it upcoming days... HTH.

Greetings,

Mischa.

#21 - 2008-12-12 01:55 - Kihyun Yun

Thank you for your quick response.

I think new patch have to include any changes after last patches was out.

#22 - 2008-12-12 06:18 - Kihyun Yun

I've tested it.

It works well.

All my company developers will use this features after my announcement.

Next week, I will post a more detailed result.

#23 - 2008-12-22 23:39 - Mischa The Evil

- % Done changed from 80 to 50

Mischa The Evil wrote:

- I noticed that a new user is made in the DB as anonymous without login (it isn't showed neither). Currently I can't remember if that was also the case with my patch against [r1926](#). Can you confirm?

After some more tests and tries I noticed that this isn't due to my patch, but default by Redmine. So no problem here...

Mischa The Evil wrote:

- How to handle the anonymous commits at all (user is not Redmine-user and not mapped to a Redmine-user) after the user-mapping implementation?

This got a bit trickier. The current behaviour I stick with is as follows:

- When the SCM-user is not a Redmine-user **and** not mapped to a Redmine-user the static ("personalized") issue-journals and timelogs are getting the Anonymous-user as author.
- When the SCM-user is not a Redmine-user **but** is mapped to a Redmine-user the static ("personalized") issue-journals and timelogs are getting the mapped username as author.
- When the SCM-user **is a Redmine-user** the static ("personalized") issue-journals and timelogs are getting the redmine username as author.

Since were talking about "static" DB-entries here (issue-journals and timelogs) the given author-name doesn't change after user-mapping changes, after Redmine has fetched the commits due to the fact that these items are created while the commit is fetched by Redmine.

Mischa The Evil wrote:

- All the core-tests must be made at least compatible with this patch.

This got even some more difficult. The tests initially provided by Jonas on his first try on this patch weren't covering my personal changes at first. Though they succeeded combined with my changes. Second, the new user-mapping functionality (combined with my modified patch) breaks the tests from Jonas.

Most likely this is not due this patch. AFAICS the three failing tests are incorrect, not the functionality. The failing tests are:

1.1) Failure:

```
test_log_time_should_not_create_duplicate_logs(ChangesetTest)
  [./test/unit/changeset_test.rb:131:in `test_log_time_should_not_create_duplicate_logs']:
<1> expected but was
<2>.
```

2.2) Failure:

```
test_set_committer_identified_by_email(ChangesetTest)
  [./test/unit/changeset_test.rb:201:in `test_set_committer_identified_by_email']:
<#<User id: 1, login: "admin", hashed_password: "d033e22ae348aeb5660fc2140aec35850c4da997", firstname: "redMine", lastname: "Admin", mail: "admin@somenet.foo", mail_notification: true, admin: true, status: 1, last_login_on: "2006-07-19 22:57:52", language: "en", auth_source_id: nil, created_on: "2006-07-19 19:12:21", updated_on: "2006-07-19 22:57:52", type: "User">> expected but was
<#<AnonymousUser id: 6, login: "", hashed_password: "1", firstname: "", lastname: "Anonymous", mail: "", mail_notification: false, admin: false, status: 1, last_login_on: nil, language: "", auth_source_id: nil, created_on: "2006-07-19 19:33:19", updated_on: "2006-07-19 19:33:19", type: "AnonymousUser">>.
```

3.3) Failure:

```
test_set_committer_identified_by_login(ChangesetTest)
  [./test/unit/changeset_test.rb:217:in `test_set_committer_identified_by_login']:
<#<User id: 1, login: "admin", hashed_password: "d033e22ae348aeb5660fc2140aec35850c4da997", firstname: "redMine", lastname: "Admin", mail: "admin@somenet.foo", mail_notification: true, admin: true, status: 1, last_login_on: "2006-07-19 22:57:52", language: "en", auth_source_id: nil, created_on: "2006-07-19 19:12:21", updated_on: "2006-07-19 22:57:52", type: "User">> expected but was
<#<AnonymousUser id: 6, login: "", hashed_password: "1", firstname: "", lastname: "Anonymous", mail: "", mail_notification: false, admin: false, status: 1, last_login_on: nil, language: "", auth_source_id: nil, created_on: "2006-07-19 19:33:19", updated_on: "2006-07-19 19:33:19", type: "AnonymousUser">>.
```

Currently I'm not able to rewrite Jonas' tests to run successfully with my modified patch against [r2118](#). That is a bit up my current skill sadly :(But I'm currently working myself through some thick books (The pragmatic and the RoR-bible) so let's see what the future will bring...

Mischa The Evil wrote:

- Some further general tests...

Ok, now I've had some time to test this more extensive I can conclude that the alpha-version of my modified, extended patch (uploaded here <http://www.redmine.org/issues/show/1518#note-20>) is functioning correctly.

The feature-set is the same as described in <http://www.redmine.org/issues/show/1518#note-14> with the exception of the user-mapping thing which I've explained above.

Mischa The Evil wrote:

I have wrapped-up the working **alpha** version against [r2118](#) of the trunk for the "edge-addicts". This is without the test-changes. I'll look at that later.

Please only test on a dedicated, non-production deployment of Redmine [r2118](#).

Given the circumstances I've decided to currently keep the modified test-files out of the patch. This leads to the "conclusion" that the alpha-patch is ok for now. Therefor I won't update the patch for now...

Though in future I'll try to keep the patch at least compatible with recent releases/dev-line.

Regarding the warning: this keeps standing... :-)

HTH...

#24 - 2008-12-22 23:52 - Mischa The Evil

Jean-Philippe: Would it be possible to schedule this feature for the 0.9-release? I'd really like to see this in the Core and since it can be turned-off (which is actually the default state) in the current implementation I don't see much obstacles for it...

I don't see how a feature like this can be implemented using a plugin to keep it out of the core but that could be my innocence and inexperience...

What's your expert-opinion on this?

#25 - 2009-02-01 18:06 - Jean-Philippe Lang

This is a requested feature that could be added in 0.9 indeed.

But this patch seems to do more than allowing time log via commit message.
It breaks a few tests too.

#26 - 2009-03-30 17:55 - Mischa The Evil

Jean-Philippe: at first my apologies for my slow response to your comment... It's been tough to keep up with the quick development of Redmine... :)

Jean-Philippe Lang wrote:

But this patch seems to do more than allowing time log via commit message.

Indeed, additionally to the timelog creation via the use of the timelog-keyword in a commit message it also provides a way to change a issues done-ratio using a dedicated done-ratio keyword.

This is supported with the following additional changes to handle the above mentioned features:

- added default timelog-comment setting if time is added in commit-message but timelog-comment is missing (blank [default] or changeset-ID)
- added default timelog activity-type setting
- changed issue-journal creation behaviour in commit-message processing; now a journal is only created when:

```
[core] an issue is fixed using fixing-keywords (thus status (AND possibly done-ratio) is changed)
OR
[ext] advanced commit parsing is active AND
      issue is not fixed using core fixing-keywords AND
      issue-ratio is changed explicitly in commit-message using ratio-keyword (this requires ref-keyword to be configured explicitly)
OR
[ext] advanced commit parsing is active AND
      issue is not fixed using core fixing-keywords AND
      * is configured as reference-keyword AND
      issue-ratio is changed explicitly in commit-message using ratio-keyword
```

- changed used issue-journal content when only an issue's done-ratio is changed using ratio-keyword in commit-message;
 - when only an issue's done-ratio is changed: "Partially applied in rxxx"
 - when an issue is fixed using fixing-keywords: "Applied in rxxx"

Jean-Philippe Lang wrote:

It breaks a few tests too.

Indeed, you can see my initial notes about them [here](#) which I've explained further in [this issue-journal](#).
Jonas? Can you maybe take a look at them too?

HTH...

#27 - 2009-06-20 21:38 - Jens Goldhammer

+1

#28 - 2009-06-22 23:22 - Mischa The Evil

- Assignee set to Mischa The Evil

Let's see if I am able to rebase the patch against current trunk... Then we can maybe create a stable solution for this issue...

#29 - 2009-07-22 18:02 - Ken Sands

patching fails on the repositories.rb against the 0.8.4 stable, anyone able to give me a hand on it, I've just finished writing a tortoisestv plugin so I can feed back hours during the commit with a nice ui, now I just need to get this bit working, I had it working with a source build of redmine from a few months ago but other things in that build made it unusable.

by the way if anyone is interested in the tortoisestv plugin then I'll upload the source somewhere for you all.

#30 - 2009-07-25 18:19 - Mischa The Evil

Ken Sands wrote:

patching fails on the repositories.rb against the 0.8.4 stable

The last version of the patch ([logtime_changeratio_via_commit-message_extended-r2118-Alpha.diff](#)) should be compatible with Redmine 0.8.x (including release 0.8.4). I haven't patched the file repositories.rb neither. Are you sure this isn't a typo and were you actually meaning the file `_repositories.rhtml` (the view-template)?

Please let me know since I want to maintain this patch for 0.8.x at least (until I have ported it to current trunk also).

Ken Sands wrote:

by the way if anyone is interested in the tortoissvn plugin then I'll upload the source somewhere for you all.

Is that something like the one already mentioned at [ThirdPartyTools](#)?

Kind regards,

Mischa.

#31 - 2009-08-27 12:02 - Ken Sands

The last version of the patch ([logtime_changeratio_via_commit-message_extended-r2118-Alpha.diff](#)) should be compatible with Redmine 0.8.x (including release 0.8.4).

I'll try it again, I was probably making a mistake

Is that something like the one already mentioned at [ThirdPartyTools](#)?

yes, pretty much, but I've included a box for entering the time spent and a dropdown for percentage complete, plus it filters the possible issues down to those associated with the repository you are committing to. I'll tidy up the code and put it up as another entry on that page, could be helpful for some.

#32 - 2009-08-27 12:22 - Ken Sands

The last version of the patch ([logtime_changeratio_via_commit-message_extended-r2118-Alpha.diff](#)) should be compatible with Redmine 0.8.x (including release 0.8.4). I haven't patched the file repositories.rb neither. Are you sure this isn't a typo and were you actually meaning the file _repositories.rhtml (the view-template)?

the following is in that diff, it's where it failed for me, but I'll do some testing and make sure it's not because I had other patches there already.

```

Index: app/models/repository.rb =====
--- app/models/repository.rb (revision 2118)
+++ app/models/repository.rb (working copy)
@ -92,10 +92,6 @
@latest_changeset ||= changesets.find(:first)
end

- def scan_changesets_for_issue_ids
- self.changesets.each(&:scan_comment_for_issue_ids)
- end
- # Returns an array of committers usernames and associated user_id
def committers
committers ||= Changeset.connection.select_rows("SELECT DISTINCT committer, user_id FROM #{Changeset.table_name} WHERE repository_id =
#{id}")
@ -142,11 +138,6 @@
find(:all).each(&:fetch_changesets)
end

- # scan changeset comments to find related and fixed issues for all repositories
- def self.scan_changesets_for_issue_ids
- find(:all).each(&:scan_changesets_for_issue_ids)
- end
-
def self.scm_name
'Abstract'
end

```

#33 - 2009-11-04 18:55 - Jean-Philippe Lang

People interested in this feature should have a look at [#4155](#).
I'll wait for some feedback before choosing one of the syntax for time logging via commit message.

#34 - 2010-01-11 12:50 - Chris Platts

Jean-Philippe Lang wrote:

People interested in this feature should have a look at [#4155](#).
I'll wait for some feedback before choosing one of the syntax for time logging via commit message.

I definitely prefer the syntax of [#4155](#). It's clearer and doesn't try to do too much (at least for my purposes - of course others will disagree!).

Unfortunately, I can't get either this patch or [#4155](#) to apply cleanly to the latest trunk revision I'm running, so it's all a bit hypothetical at the moment :)

#35 - 2010-05-16 01:16 - Mischa The Evil

Just to temporarily drop the thing I came around which might be interesting in this perspective. It shows how related issues are partially solved in Retrospectiva.

- Issue: <http://retrospectiva.org/tickets/86>
- SCM-URL: http://retrospectiva.googlecode.com/svn/extensions/1-1/scm_ticket_update

#36 - 2010-12-29 22:13 - Jean-Philippe Lang

- Status changed from New to Closed

See [#4155](#).

#37 - 2011-08-24 21:03 - Szymon Nowak

Mischa The Evil wrote:

Just to temporarily drop the thing I came around which might be interesting in this perspective. It shows how related issues are partially solved in Retrospectiva.

- Issue: <http://retrospectiva.org/tickets/86>
- SCM-URL: http://retrospectiva.googlecode.com/svn/extensions/1-1/scm_ticket_update

Is there any way to make it works in latest Redmine (1.2.1)? I think it's extremely useful feature.

#38 - 2011-08-24 21:06 - Szymon Nowak

Chris Platts wrote:

Jean-Philippe Lang wrote:

People interested in this feature should have a look at [#4155](#).
I'll wait for some feedback before choosing one of the syntax for time logging via commit message.

I definitely prefer the syntax of [#4155](#). It's clearer and doesn't try to do too much (at least for my purposes - of course others will disagree!).

Unfortunately, I can't get either this patch or [#4155](#) to apply cleanly to the latest trunk revision I'm running, so it's all a bit hypothetical at the moment :)

Is there any way to make it works in latest Redmine (1.2.1)? I think it's extremely useful feature.

Ps. I've quoted wrong post :P

Files

patch_commit_messages.diff	21.9 KB	2008-07-03	Jonas von Andrian
patch_commit_messages2.diff	21.9 KB	2008-07-03	Jonas von Andrian
patch_commit_messages3.diff	22.6 KB	2008-07-03	Jonas von Andrian
patch_commit_messages4.diff	24.1 KB	2008-07-06	Jonas von Andrian
logtime_changeratio_via_commit-message-r1926.diff	23 KB	2008-10-06	Mischa The Evil
logtime_changeratio_via_commit-message_extended-r1926.diff	27.1 KB	2008-10-06	Mischa The Evil
logtime_changeratio_via_commit-message_extended-r2118-Alpha.diff	15.8 KB	2008-12-10	Mischa The Evil