

| | | | |
|-----------------|-------------------------------------|----------------------|------------|
| ステータス: | New | 開始日: | 2010/04/20 |
| 優先度: | 通常 | 期日: | |
| 担当者: | | 進捗率: | 0% |
| カテゴリ: | SCM_3 | 予定工数: | 0.00時間 |
| 対象バージョン: | Unplanned backlogs_23 | 作業時間: | 0.00時間 |
| Redmineorg_URL: | https://www.redmine.org/issues/5357 | status_id: | 1 |
| category_id: | 3 | tracker_id: | 1 |
| version_id: | 23 | plus1: | 0 |
| issue_org_id: | 5357 | affected_version: | |
| author_id: | 13151 | closed_on: | |
| assigned_to_id: | 11192 | affected_version_id: | 6 |
| comments: | 36 | | |

説明

The repository browser displays the commit history sorted by date/time. When developers are working in different time-zones (or have a badly adjusted PC clocks), the commit sequence gets all jumbled up in the repository view.

Reproduction:

Commit something to a GIT repo.

Set PC clock to - <1 day>.

Commit something else.

journals

Same issue as #3567.

"order by date/time" is only valid for centralized system like CVS or Subversion.

I have a patch, attachment:sort-changesets-by-id.diff, to fix this issue, but which is yet well tested.

Thanks, but your changeset does appear to solve the problem for a GIT repo. It did however provide me with enough info to solve it myself:

attachment:sort-changes-by-id-for-git.diff

It appears that similar changes will have to be applied to ./test/unit/repository_git_test.rb as well, but I am not familiar enough with the unit test infrastructure to make these changes.

Oops, I made a typo: Your changeset does +not+ appear to solve the problem for a GIT repo

Crap. attachment:sort-changes-by-id-for-git.diff doesn't work:

With the initial fetch, redmine adds the changesets in reverse commit order.

When additional changesets are imported, they are added to the back of the table, also in reverse commit order.

Sorting by changesets.id therefore results in a discontinuity

I tried to reverse the order with which changesets are added to the database by using the `:reverse` option of `scm.revisions()`, but all the other mechanisms use the `changesets.committed_on` field to determine if something is new, so new changesets are not imported resulting in changeset fetch deadlocks.

Hi, I read #2108 and feel the issue is much tangled than Mercurial's (#3567.)

According to `@git help log@`, it sorts changes in chronological order, and Redmine also orders by `@committed_on@`. So how do they become different?

BTW, your information about git really helps me. Thanks! :)

Not true. The `git log --help` does not seem to be accurate (and does not make sense either). Here is a clear example of a log pulled from GIT (`git log --date-order` and `git log` produces the same results):

```
commit 1820eb8224442f04f055cce31f30e1813bbc61c8
Author: Pieter Smith
Date: Tue Apr 20 03:09:01 2010 -0400
```

```
Also ignore mongrel rails PID files in ./log
```

```
commit 0ff42bcb8069b08c8f558ee1763c96167872a41f
Author: Pieter Smith
Date: Wed Apr 21 18:02:58 2010 -0400
```

```
Applied mongrel --prefix use patch:
* http://www.ruby-forum.com/topic/183811
```

```
commit 12fb96220972e89f8748d81eb7082de1697b12d8
Author: Pieter Smith
Date: Mon Apr 19 02:46:56 2010 -0400
```

```
Added redmine checkout plugin
```

```
commit 10406e90f8797dc4299717f1d1260b9976b23c1e
Author: Pieter Smith
Date: Mon Apr 19 02:37:15 2010 -0400
```

```
Ignore files directory entirely:
* Allow use of symlinks to place attachments elsewhere
```

```
commit 29671188126e774529713847155a57e522f84b4b
Author: Pieter Smith
Date: Mon Apr 19 02:33:57 2010 -0400
```

```
Added --proj-id-match and --update-only options to reposman
```

```
commit fc6ccc1ea2593f29fcbbe3d021880b2212b93639
Author: Jean-Philippe Lang
Date: Sun Feb 28 15:12:09 2010 +0000
```

```
tagged version 0.9.3
```

```
git-svn-id: svn+ssh://rubyforge.org/var/svn/redmine/tags/0.9.3@3515 e93f8b46-1217-0410-a6f0-8f06a7374b81
```

And the same log in redmine:

!Redmine_Repository_View.JPG!

And the missing file:

Hmm, I'll see it later how git considers, but attachment:git-changeset-order-workaround.diff may partially fix the problem. (though I'm not sure.)

It changes the insertion order to be "ORDER BY id" as possible. (so please run @clear_changesets@ to regenerate changesets table, and apply also attachment:sort-changesets-by-id.diff patch)

Note: Git backend checks the last 7-day logs everytime calling fetch_changesets.
Then deletes disappeared changesets from database, and finally inserts missing ones.

Thank you!

Further testing is needed, but these two patches (attachment:sort-changesets-by-id.diff and attachment:git-changeset-order-workaround.diff) appear to fix the issue. Will this be incorporated in a near-future release, or will I have to patch future versions?

P.S. How should I run @clear_changesets@? This didn't work:

```
@ruby script/runner "Repository.clear_changesets" -e production
```

I deleted the contents of the changesets table in the MySQL database to test this, but I would prefer not to circumvent the application by messing with the database. ----- Hi, thank you for testing it!
Pieter Smith wrote: > Will this be incorporated in a near-future release, > or will I have to patch future versions? I'm not sure but since it requires to refetch all changesets, it might be harder to be merged into Redmine release. Anyway, I need to understand more about Git. (I mostly use Mercurial and its Hg-Git extension.) > P.S. How should I run @clear_changesets@? Oops! I completely misread the comment of repository/git.rb saying "The repository can still be fully reloaded by calling #clear_changesets." There seems no way to run it from command-line. You can regenerate changesets table by: 1. delete repository from settings tab; 2. recreate the same repository, though it's really indirect way. :) ----- FYI: git 1.7 supports "git commit --date DATE". <http://www.kernel.org/pub/software/scm/git/docs/RelNotes-1.7.0.txt>

----- Interesting... The commit date can be overridden. Not quite sure how it could practically help with the display ordering though. :(----- I think we can create test repository, unit test and functional test. * source:tags/1.0.0/test/fixtures/repositories/git_repository.tar.gz * source:tags/1.0.0/test/unit/repository_git_test.rb * source:tags/1.0.0/test/functional/repositories_git_controller_test.rb ----- If you need any assistance, I'd be glad to help.
----- Yuya is working now. Please see <http://www.redmine.org/issues/4455#note-123> and <http://www.redmine.org/issues/4455#note-130>.

----- Toshi MARUYAMA wrote: > Yuya is working now. > Please see <http://www.redmine.org/issues/4455#note-123> and <http://www.redmine.org/issues/4455#note-130>. Ah, but I've just tried minimizing the impact of #4455 patch, to make a difference _only to mercurial backend_. ;) As I said before, my workaround patch for git requires database regeneration. I'm not sure how the maintainers and redmine+git users think, though, it sounds awful to me. ----- There is a patch for time-zone of git adapter at #6346.

----- Toshi MARUYAMA wrote: > There is a patch for time-zone of git adapter at #6346. No, this is not related. The problem here is that redmine assumes "younger" revisions to be the "newest", which isn't the case with git. The commit date in git is just metadata, it's neither authoritative nor an indication of the ordering of the revisions. The revision order in git is provided only by the linked list nature of revisions, i.e. a revision holds a reference to its parent revision, which in turn holds a reference to its parent, and so on. This discrepancy between "redmine assumes commit dates in consecutive commits always increases" and "the commit date in git is just metadata" is I think the problem, but that's one that will take more work than "just" fixing the git adapter. ----- Toshi MARUYAMA wrote: > FYI: git 1.7 supports "git commit --date DATE". > <http://www.kernel.org/pub/software/scm/git/docs/RelNotes-1.7.0.txt> Git has a @AuthorDate@ and @CommitDate@. * #2108 * Pro Git "Maintaining a Project":<http://progit.org/book/ch5-3.html> "--date" option changes @AuthorDate@. @AuthorDate@ and @CommitDate@ can be changed by @GIT_AUTHOR_DATE@ and @GIT_COMMITTER_DATE@ environment variables.

```
$ GIT_AUTHOR_DATE="1990-01-01 00:00:00 +0100" /
  GIT_COMMITTER_DATE="2000-01-01 00:00:00 -0500" /
  git commit -a -m "test."
```

```
$ git log --pretty=fuller -n1 | cat
commit da940a9ed13807bddc63b7bf5c295291d1a7f6ba
Author:      Toshi MARUYAMA
AuthorDate: Mon Jan 1 00:00:00 1990 +0100
Commit:     Toshi MARUYAMA
CommitDate: Sat Jan 1 00:00:00 2000 -0500
```

```
test.
```

git rebase changes @CommitDate@ and keeps @AuthorDate@.

```
$ git log --graph --all --pretty=fuller -n2
* commit 585b4abad53aa65fd6bf6109ef6e7ffdd06808a8
| Author: Toshi MARUYAMA
| AuthorDate: Wed Sep 22 21:26:24 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Wed Sep 22 21:26:24 2010 +0900
|
| test01.
|
| * commit d0331012c8ac9c3c588723547a98695d77ae013a
|/ Author: Toshi MARUYAMA
| AuthorDate: Wed Sep 22 21:24:08 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Wed Sep 22 21:24:08 2010 +0900
|
| test00.
|
$ git checkout d0331012c8ac9c3c588723547a98695d77ae013a
$ git rebase 585b4abad53aa65fd6bf6109ef6e7ffdd06808a8
$ git log --graph --all --pretty=fuller -n2
* commit 1809eacbcd8844fab5df80fcd95a7f7315a0c6c4
| Author: Toshi MARUYAMA
| AuthorDate: Wed Sep 22 21:24:08 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Wed Sep 22 21:35:52 2010 +0900
|
| test00.
|
| * commit 585b4abad53aa65fd6bf6109ef6e7ffdd06808a8
| Author: Toshi MARUYAMA
| AuthorDate: Wed Sep 22 21:26:24 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Wed Sep 22 21:26:24 2010 +0900
|
| test01.
|
$ git log --graph --all -n2
* commit 1809eacbcd8844fab5df80fcd95a7f7315a0c6c4
| Author: Toshi MARUYAMA
| Date: Wed Sep 22 21:24:08 2010 +0900
|
| test00.
|
| * commit 585b4abad53aa65fd6bf6109ef6e7ffdd06808a8
| Author: Toshi MARUYAMA
| Date: Wed Sep 22 21:26:24 2010 +0900
|
| test01.
```

Yuya Nishihara wrote:

Toshi MARUYAMA wrote:

Yuya is working now.

Please see <http://www.redmine.org/issues/4455#note-123> and <http://www.redmine.org/issues/4455#note-130>.

Ah, but I've just tried minimizing the impact of #4455 patch, to make a difference only to mercurial backend. ;)

As I said before, my workaround patch for git requires database regeneration. I'm not sure how the maintainers and redmine+git users think, though, it sounds awful to me.

I referred Yuya's "mercurial implementation":

http://bitbucket.org/yuja/redmine-mq-issue4455/src/a7f2ec8a9198/hg-changeset-order_patch#cl-22 ,
then I implement ordering only for git.

I pushed to my github and I attach a new patch.

- <http://github.com/marutosi/redmine/commits/git-order>
- <http://github.com/marutosi/redmine/commits/6806ea5af21e75fc1>

And I post logs and images of following test repository.

- http://github.com/marutosi/redmine_test_git_repository/commits/marutosi00
- http://github.com/marutosi/redmine_test_git_repository/commits/ce4e78fcffa81e538

```
$ git log --graph --pretty=fuller -n3 | cat
* commit ce4e78fcffa81e538ad3ef41065a285da462c94c
| Author: Toshi MARUYAMA
| AuthorDate: Wed Jul 11 00:00:00 1990 +0100
| Commit: Toshi MARUYAMA
| CommitDate: Sun Jul 11 00:00:00 2010 -0500
|
|     date test 3.
|
* commit 8312fa08b40540b0b301241a6a6b032fac08d3f3
| Author: Toshi MARUYAMA
| AuthorDate: Tue Jul 10 00:00:00 1990 +0100
| Commit: Toshi MARUYAMA
| CommitDate: Sat Jul 10 00:00:00 2010 -0500
|
|     date test 2.
|
* commit 3329c803bc79b6657660b7e045a8d08802ca32f7
| Author: Toshi MARUYAMA
| AuthorDate: Thu Jul 12 00:00:00 1990 +0100
| Commit: Toshi MARUYAMA
| CommitDate: Mon Jul 12 00:00:00 2010 -0500
|
|     date test 1.
```

```
$ git log --graph -n3 | cat
* commit ce4e78fcffa81e538ad3ef41065a285da462c94c
| Author: Toshi MARUYAMA
| Date: Wed Jul 11 00:00:00 1990 +0100
|
|     date test 3.
|
* commit 8312fa08b40540b0b301241a6a6b032fac08d3f3
| Author: Toshi MARUYAMA
| Date: Tue Jul 10 00:00:00 1990 +0100
|
|     date test 2.
|
* commit 3329c803bc79b6657660b7e045a8d08802ca32f7
| Author: Toshi MARUYAMA
| Date: Thu Jul 12 00:00:00 1990 +0100
|
|     date test 1.
```

I hate to quote myself, but:

Felix Schäfer wrote:

The revision order in git is provided only by the linked list nature of revisions, i.e. a revision holds a reference to its parent revision, which in turn holds a reference to its parent, and so on.

So they are ordered by database insertion instead of commit date, still a workaround rather than a solution.

(and don't get me wrong: I appreciate the work you do here, but I think it goes in the wrong direction)

Felix Schäfer wrote:

(and don't get me wrong: I appreciate the work you do here, but I think it goes in the wrong direction)

I think better solution for DVCS is creating new tables for "DAG":http://en.wikipedia.org/wiki/Directed_acyclic_graph.

Please see <http://www.redmine.org/issues/4773#note-18>.

Rebased multi revisions @CommitDate@ changed same time.
So, ordering with @CommitDate@ is strange.

I pushed these revisions to my github.

- http://github.com/marutosi/redmine_test_git_repository/commits/rebase-test-00
- http://github.com/marutosi/redmine_test_git_repository/commits/57a99e05c3ef7bfd43c87

```
$ git log --graph --all --pretty=fuller -n 3 | cat
* commit c65291727d4d674f0ea4796a8a666cf6817494dc
| Author: Toshi MARUYAMA
| AuthorDate: Tue Sep 28 20:32:19 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Tue Sep 28 20:32:19 2010 +0900
|
|     rebase test 10.
|
| * commit de31c2ab3b20188b2b09e168026805688d65ef65
| | Author: Toshi MARUYAMA
| | AuthorDate: Tue Sep 28 20:28:42 2010 +0900
| | Commit: Toshi MARUYAMA
| | CommitDate: Tue Sep 28 20:28:42 2010 +0900
| |
| |     rebase test 01.
| |
| | * commit 27a6e34d47c90d632bd2207fb812c8eb49107923
| | / Author: Toshi MARUYAMA
| | | AuthorDate: Tue Sep 28 20:28:19 2010 +0900
| | | Commit: Toshi MARUYAMA
| | | CommitDate: Tue Sep 28 20:28:19 2010 +0900
| |
| |     rebase test 00.
|
$ git checkout de31c2ab3b20188b2b09e168026805688d65ef65
$ git rebase c65291727d4d674f0ea4796a8a666cf6817494dc
```

```
$ git log --graph --all --pretty=fuller -n3 | cat
* commit 57a99e05c3ef7bfd43c87146f728e51665234a28
| Author: Toshi MARUYAMA
| AuthorDate: Tue Sep 28 20:28:42 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Tue Sep 28 20:39:11 2010 +0900
|
|     rebase test 01.
|
| * commit dd6129a06b9a38d7d8a80e140cf804d5c76e0e3b
| Author: Toshi MARUYAMA
| AuthorDate: Tue Sep 28 20:28:19 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Tue Sep 28 20:39:11 2010 +0900
|
```

```
| rebase test 00.
|
| * commit c65291727d4d674f0ea4796a8a666cf6817494dc
| Author: Toshi MARUYAMA
| AuthorDate: Tue Sep 28 20:32:19 2010 +0900
| Commit: Toshi MARUYAMA
| CommitDate: Tue Sep 28 20:32:19 2010 +0900
|
| rebase test 10.
```

I'm far away being a git expert but I think rebasing on a public "official" repository is evil anyway. Please see [git-rebase\(1\) NOTES](#) and [RECOVERING FROM UPSTREAM REBASE](#). I'm not sure if this could be fixed in redmine with reasonable efforts.

Bernhard Furtmueller wrote:

I'm far away being a git expert but I think rebasing on a public "official" repository is evil anyway.

Yes. In Git and Mercurial, rebasing and history editing on public repository is rare case. But, rebasing and history editing on private repository and pushing these revisions to public repository is popular. Please see <http://www.redmine.org/issues/4455#note-146>.

These are <https://github.com/edavis10/chiliproject/tree/ticket/master/30-upstream-code-review> image and log.

Eric seemed to rebase or cherry-pick.
Latest CommitDate changed "Sat Feb 5 21:08:10 2011 -0800".

!git-rebase-20110208.png!

```
$ git log --graph -b br-chili-eric --pretty=fuller -n3 | cat
* commit dc016d226d1afce84aa7ea478434c7064e5d13f1
| Author: Jean-Philippe Lang
| AuthorDate: Thu Dec 23 10:04:08 2010 +0000
| Commit: Eric Davis
| CommitDate: Sat Feb 5 21:08:10 2011 -0800
|
| Set VERSION to 1.0.5.
|
| git-svn-id: svn+ssh://rubyforge.org/var/svn/redmine/trunk@4568 e93f8b46-1217-0410-a6f0-8f06a7374b81
|
* commit 3659b7164c6f15971bfd5a7384fa517f5248e8c0
| Author: Jean-Philippe Lang
| AuthorDate: Thu Dec 23 10:03:32 2010 +0000
| Commit: Eric Davis
| CommitDate: Sat Feb 5 21:07:14 2011 -0800
|
| Updated INSTALL for 1.0.5
|
| git-svn-id: svn+ssh://rubyforge.org/var/svn/redmine/trunk@4567 e93f8b46-1217-0410-a6f0-8f06a7374b81
|
* commit 7d152dd51635e2e19edb0c55b0cb2cad98ad17da
| Author: Jean-Philippe Lang
| AuthorDate: Thu Dec 23 10:02:08 2010 +0000
| Commit: Eric Davis
| CommitDate: Sat Feb 5 21:07:07 2011 -0800
|
| Updated CHANGELOG for 1.0.5
|
| git-svn-id: svn+ssh://rubyforge.org/var/svn/redmine/trunk@4566 e93f8b46-1217-0410-a6f0-8f06a7374b81
```

This is a patch for r5074 .

<https://www.chiliproject.org/projects/chiliproject/repository/show?rev=69bad846048e6b854cda272e2701771f26172234>

!chili-pdf-issue-61.png!

related_issues

relates,Closed,7146,Git adapter lost commits before 7 days from database latest changeset
relates,New,7047,Git adapter very slow when a commit modifies a lot of files
relates,New,7648,Git: Repository Statistics Incorrect
relates,New,4657,Wrong timestamp for import of entries of the last ChangeSet.
relates,Closed,7821,Git "previous" and "next" revisions are incorrect
relates,Closed,8049,Git: fetch_changesets not updating project's repository
relates,New,2799,Support for Bazaar's shared repositories (created with init-repo)
relates,New,6566,Convert the git adapter to rugged
relates,Closed,5501,Git: Mercurial: Adding visual merge/branch history to repository view
relates,New,11710,Git: Repository view author/date inconsistency
relates,Closed,2108,git repository browser uses author time instead of commit time for ordering
duplicates,Closed,13311,In repository view, git commits are ordered by date, not by commit orders

履歴

#1 - 2022/05/10 17:22 - Admin Redmine

- カテゴリ を SCM_3 にセット

- 対象バージョン を Unplanned backlogs_23 にセット