

Ruby trunk - Feature #11741

Migrate Ruby to Git from Subversion

11/25/2015 11:44 PM - maclover7 (Jon Moss)

Status:	Rejected	
Priority:	Normal	
Assignee:		
Target version:		
Description		
<h2>Git to SVN</h2>		
<p>Converting Ruby wholesale from Subversion to Git (not necessarily Github!) has been a long time coming, and I think it's finally time to make the switch. Ruby already has an official Git repo up on Github, and the main contributing.rdoc file in that repo is meant for Git, not Subversion. Git is definitely the most popular VCS (version control system) in the Ruby ecosystem, and it's time for the language itself to convert. I propose that Ruby use Gitlab to manage its issue tracker, merge/pull request tracker, and the actual Git repository itself. Gitlab is an open source Ruby on Rails that many large corporations have begun to use for Git repository management + related tools. Gitlab also has a CI toolset built right into the core application, so we could also run CI all on the same set of servers. I have contacted and have a sponsor (that's a major Ruby server hosting company) ready to foot the bill for all servers needed to run a cluster of Gitlab servers for Ruby.</p>		
<p>Below is a preliminary checklist for how to go about the change:</p>		
<h2>Actually convert codebase from SVN to Git</h2>		
<ul style="list-style-type: none">• Either use the <code>svn2git</code> gem• Or clone down the Git repository from https://github.com/ruby/ruby		
<h2>Redmine --> Gitlab</h2>		
<ul style="list-style-type: none">• Contact sponsor [REDACTED] to get GitLab servers spinning, and live (under <code>git.ruby-lang.org</code>, maybe?)• Get CI running on Gitlab (start off with Ubuntu Linux)• Migrate all issues (open and closed, or just open?) from Redmine to Gitlab via Redmine and Gitlab APIs• Begin migrating all pull requests from Github to Gitlab		
<h2>Final Transition</h2>		
<ul style="list-style-type: none">• Post large notice on Redmine website saying that Redmine + Subversion will be deprecated soon• After two months (maybe shorter? longer?) close down old Redmine + Subversion servers		
<p>I am happy to make adjustments as necessary to the timeline listed above, and to take the lead on this project. Let me know if we want to continue the conversation with the server sponsor and the Ruby core team. <3</p>		
Related issues:		
Related to Ruby trunk - Misc #10547: How to move the ruby project to git		Rejected
Related to Ruby trunk - Feature #14551: What's missing to switch to Git inste...		Rejected

History

#1 - 11/26/2015 12:06 AM - naruse (Yui NARUSE)

- Status changed from Open to Rejected

Why you don't read previous discussion when you want to discuss about history management.

#2 - 11/26/2015 12:21 AM - hsbt (Hiroshi SHIBATA)

- Related to Misc #10547: How to move the ruby project to git added

#3 - 11/26/2015 12:29 AM - hsbt (Hiroshi SHIBATA)

You should read discussion of ruby-core archive at <https://bugs.ruby-lang.org/issues/10547#note-5>

#4 - 11/26/2015 12:35 AM - bascule (Tony Arcieri)

It seems like this issue addresses at least some of the points raised in [#10547](#), namely the initial steps of a plan to migrate to Git from SVN.

It also seems like this is an issue that should be periodically discussed, especially if a more concrete plan is forming.

Seeing this issue closed outright without discussion is very frustrating to an outsider.

#5 - 11/26/2015 01:15 AM - naruse (Yui NARUSE)

Tony Arcieri wrote:

It seems like this issue addresses at least some of the points raised in [#10547](#), namely the initial steps of a plan to migrate to Git from SVN.

It also seems like this is an issue that should be periodically discussed, especially if a more concrete plan is forming.

Why this is periodically discussed is because anyone doesn't form a concrete plan addressing previous discussion. I know the plan will include some hard work, and believe they can't do that who hesitate to dig previous discussion.

Seeing this issue closed outright without discussion is very frustrating to an outsider.

If you really want to join the development of Ruby, you are insider.

#6 - 11/26/2015 02:41 AM - maclover7 (Jon Moss)

Hi Yui,

I apologize for not finding previous Git to SVN discussions on the Redmine issue tracker. I did search for 'git subversion' on the Redmine issue tracker, but only 12 results came back, and none of them were the above mentioned issue 10547. I did try searching for just 'git' in the search engine, but for that search over 2,349 results came back, and I think it's unreasonable to expect someone to be able to sort through all 235 pages of those results.

I am happy to make a small chart comparing Git and Subversion conversion pro/cons, see below. I do think that my plan listed above is fairly concrete with just a few kinks to be worked out (DNS / quantity of servers to be provided). I also think that at this point the argument is more focused on ideology of Git vs SVN rather than the economics/money of making this decision, since the sponsor would be taking care of the cost of servers for us.

Git to Subversion Pros (Yay Git!)

- Matz has promised both in his 2014 RubyConf, and now at his 2015 RubyConf talk, that Ruby will be moving to Git "soon"
- The other major implementations of Ruby, Rubinius / JRuby, as well as Matz's own mruby all use Git for their version control system.
- Most Ruby projects use Git, for example:
 - Rails / Sinatra
 - RSpec / Minitest
 - Rake
 - RubyGems / Bundler
- The late Jim Weirich said in one of the mailing list discussions you mentioned earlier that, "I have seen the number of contributions to Rake skyrocket after the switch."
- He also said that, "[...] once past the initial learning curve, I'm very happy to be using git and would choose git over svn for any new project where given the choice."
- Git is much more widely used than Subversion, and the Ruby language itself is the only critical Ruby infrastructure project I can think of that uses Subversion
- Ruby already has a Git mirror, why not go "whole hog" and make the full transition over to Git? Seems a bit odd to be supporting both version control systems
- Most online coding tutorials/bootcamps + companies use Git, and do not teach or use Subversion anymore. The Ruby language itself should be promoting industry practices.

Git to Subversion Cons (Yay Subversion!)

- Ruby has been using Subversion for a while, so all key Ruby contributors know how the tool works
 - Solution: Git is widely used, and there are many tutorials to get up to speed
- Ruby has been using Redmine for a while, so all key Ruby contributors know how the tool works
 - Solution: Gitlab is a tool with great documentation and a great API, so learning should be quick. Gitlab is also much more visually appealing (IMO) as compared to Redmine.
- Subversion is centralized, so all source code is in one place
 - Solution: This makes svn.ruby-lang.org a SPOF (single point of failure). By Git being decentralized, everyone has a copy of all of Ruby's history locally and easily accessible. (Sorry I couldn't come up with anymore cons, I only use Git, and have not used Subversion before)

If you are still on the fence, you can check out [this blog post](#) from DHH (David Heinemeier Hansson) on the Rails blog about their move to Git from SVN. You can also check out a tutorial he links to [here](#).

Please do let me know if there is anything I can do to help out with this ticket.

#7 - 11/26/2015 04:08 AM - normalperson (Eric Wong)

me@jonathanmoss.me wrote:

- Solution: Gitlab is a tool with great documentation and a great API, so learning should be quick. Gitlab is also much more visually appealing (IMO) as compared to Redmine.

Disclaimer: I'm not speaking for anyone else, anywhere.

Fwiw, Redmine also supports git, too.

How well does Gitlab work for users without JavaScript or graphical interfaces? As long as I run Ruby without a GUI or JS, I expect to be able to contribute to it from computers without those things.

Redmine at least mostly works from w3m.

Fwiw, I'd rather not have any bug tracker at all, just a mailing list like the git project itself.

- Subversion is centralized, so all source code is in one place
 - Solution: This makes svn.ruby-lang.org a SPOF (single point of failure). By Git being decentralized, everyone has a copy of all of Ruby's history locally and easily accessible. (Sorry I couldn't come up with anymore cons, I only use Git, and have not used Subversion before)

The problem with all GitLab/GitHub/Redmine is they still are centralized message boards where people may hide their email addresses (maybe Redmine is better than the other 2, here). This is a SPOF when the central server goes down and people cannot communicate.

ruby-lang.org lists are closed to non-subscribers, which is still an SPOF (but at least we have each other's email addresses in case the list server goes down). Email to lists is also replicated nearly instantaneously to other archives (gmane, probably others). Honestly any subscriber has all the data to setup their own archive mirrors (and I might do this, soon).

A better way would be for ruby-lang.org to allow posting from non-subscribers and thus encourage everyone to Cc: everyone else. This mitigates the SPOF component greatly (and is what git and linux-kernel lists @vger.kernel.org do) to the point which nobody really notices or cares.

So if we migrate entirely to git, we ought to adopt the workflows git developers themselves use.

#8 - 11/28/2015 04:20 PM - maclover7 (Jon Moss)

Hi Eric,

I think it's unreasonable to ask a website to degrade and work normally without any GUI or use of Javascript -- these are the core foundations of the web today, and in order to use the web, you must use these technologies. I also think it's uncalled for to ask a website to work through a terminal window -- again, this is not what the modern web was designed for. If you do need Gitlab to work without a GUI or JS, Gitlab has a great API and you can build a small Sinatra app for yourself to consume Gitlab's data.

Most software projects these days (especially ones created in Ruby) do not use mailing lists for contributions. Only the very large ones, like Git and Linux as you have mentioned previously, use mailing lists for contributions. I think using a mailing list as a way to accept contributions would create a very large learning curve for beginners, and would increase the bar for knowledge needed to be able to contribute. I don't agree with you that just because Ruby would be changing to Git, "we ought to adopt the workflows git developers themselves use." I don't think the toolset that the core Git developers use would be appropriate here, since Ruby is something that's much higher up the technical stack (more abstractions).

I think using something like Gitlab, as compared to Github, is compromise enough for a distributed system, and something that would meet your demands. I am not trying to be rude by saying this, but I don't think the ideology of one person should be holding back a migration like this. I am happy to try and find common ground, but I think some of your requests are a little too much (example would be needing site to be viewable over terminal, with no JS).

Please do let me know if there is anything I can do to help out with this ticket.

#9 - 11/28/2015 10:29 PM - normalperson (Eric Wong)

me@jonathanmoss.me wrote:

I think it's unreasonable to ask a website to degrade and work

normally without any GUI or use of Javascript -- these are the core foundations of the web today, and in order to use the web, you must use these technologies. I also think it's uncalled for to ask a website to work through a terminal window -- again, this is not what the modern web was designed for.

I have no interest in a "modern web" which requires hundreds of megabytes of memory to exchange small pieces of text. That is discriminatory to users who cannot afford to upgrade.

Ruby isn't known for being memory-efficient, but nearly all of my Ruby processes are under 50MB of RAM, most are under 20MB. Ruby certainly never requires a GUI to run; thousands of servers run Ruby that way.

Given an Internet connection, anything that can run Ruby should be able to contribute to it.

And given things like NoScript are popular, so I would say working without JS is still important.

If you do need Gitlab to work without a GUI or JS, Gitlab has a great API and you can build a small Sinatra app for yourself to consume Gitlab's data.

Maybe that's an option; but signing up isn't intuitive.

I tried signing up for GitLab.com with w3m and I see 4 unlabeled fields which requires manually inspecting each element to know what it is.

The terms of service are less strict than GitHub (a good thing); I would assume any self-hosted GitLab instance would not have to abide by the same laws which govern GitLab B.V.

Redmine at <https://bugs.ruby-lang.org/> has no terms-of-service at all and the signup is clearly labeled.

Most software projects these days (especially ones created in Ruby) do not use mailing lists for contributions. Only the very large ones, like Git and Linux as you have mentioned previously, use mailing lists for contributions. I think using a mailing list as a way to accept contributions would create a very large learning curve for beginners, and would increase the bar for knowledge needed to be able to contribute. I don't agree with you that just because Ruby would be changing to Git, "we ought to adopt the workflows git developers themselves use." I don't think the toolset that the core Git developers use would be appropriate here, since Ruby is something that's much higher up the technical stack (more abstractions).

You're right about the current state of things. But do you wonder why projects like git and Linux have attracted so many contributors? Perhaps the lack of registration barrier has something to do with it.

Back in the early days, git was heavily implemented in shell and Perl scripts which were very high-level. I wouldn't say Ruby is any higher up the technical stack than using the git plumbing.

I think using something like Gitlab, as compared to Github, is compromise enough for a distributed system, and something that would meet your demands. I am not trying to be rude by saying this, but I don't think the ideology of one person should be holding back a migration like this. I am happy to try and find common ground, but I think some of your requests are a little too much (example would be needing site to be viewable over terminal, with no JS).

I have no more influence in this matter than you do. In fact, I have always refused to ever have any more influence than anybody else, anywhere.

And maybe GitLab works reasonably well from a terminal.

Hopefully it stays that way, maybe we'll know in a few years.

OTOH, I'm really not sure it's worth changing two things at once is worth it.

There's a bunch of Redmine-specific tools the maintainers use (I haven't checked in depth); and as I said before: Redmine works with git, too.

So perhaps migrate to git first and stick with Redmine for now.

#10 - 11/29/2015 05:58 AM - duerst (Martin Dürst)

Hello Jon,

On 2015/11/29 01:20, me@jonathanmoss.me wrote:

Issue [#11741](#) has been updated by Jon Moss.

Hi Eric,

I think it's unreasonable to ask a website to degrade and work normally without any GUI or use of Javascript -- these are the core foundations of the web today, and in order to use the web, you must use these technologies.

I think this is wrong. Graceful degradation has always been an important part of Web technology. So has Accessibility. If a Web site isn't as flashy or slick when switching off JavaScript (or CSS, or something else), that's just fine. However, if a Web site is unusable/very difficult to use if you don't use JavaScript, then that's a design/implementation problem that can and should be fixed.

[This may not apply to all Web sites, e.g. not to in-browser gaming, but it definitely should apply to sites such as redmine/github/gitlab.]

I also think it's uncalled for to ask a website to work through a terminal window

There are special Web browsers that take care of this.

-- again, this is not what the modern web was designed for.

See above.

If you do need Gitlab to work without a GUI or JS, Gitlab has a great API and you can build a small Sinatra app for yourself to consume Gitlab's data.

How would a blind user, or a user that's otherwise challenged, build a Sinatra app? Why would she have to build a Sinatra app just to cover for design mistakes of the creators of the Web site.

Most software projects these days (especially ones created in Ruby) do not use mailing lists for contributions.

"do not" -> "do not only". Ruby uses mailing lists, and I guess we would accept a contribution on a mailing list if there were some interest in the contribution among the committers.

I am not trying to be rude by saying this, but I don't think the ideology of one person should be holding back a migration like this. I am happy to try and find common ground, but I think some of your requests are a little too much (example would be needing site to be viewable over terminal, with no JS).

A Web site such as gitlab to be viewable and usable (not necessarily with exactly the same bells and whistles as in a graphical browser) in a terminal-based browser without JavaScript would just be to adhere to best Web application design practice, so I have strong difficulties to understand why such a request would be "too much".

Regards, Martin.

#11 - 11/29/2015 04:16 PM - maclover7 (Jon Moss)

Hi Martin,

I think my previous response may have come out wrong -- I do think that a service like Gitlab should degrade gracefully, but then again I think the extreme of viewing webpages through a terminal window should not be supported. I downloaded w3m onto my computer and used it for a short while to see how it worked with Gitlab, and it seemed fine to me. Eric, would you want to try Gitlab out with w3m before the switch?

#12 - 11/30/2015 12:38 AM - normalperson (Eric Wong)

me@jonathanmoss.me wrote:

Eric, would you want to try Gitlab out with w3m before the switch?

The main site's Terms of Service (or any ToS) is still a problem for me. Currently there is no ToS at all on our Redmine instance and registration is completely non-intuitive on GitLab w/o JS.

I've been completely supportive of git since 2005; never cared for Redmine much, but GitLab seems worse than Redmine.

What happens when GitLab is acquired by another company? GitLab just acquired Gitorious this year and shut it down along with all it's development resources. I fully expect somebody like GitHub or Atlassian to buy GitLab one day and do the same thing they did to Gitorious.

Redmine's been through a lot, but hsbt et. al. seem to be doing a good job of maintaining it so far.

We'd also have to do a lot of work to migrate from Redmine to GitLab; which doesn't seem to have any tangible benefits[1], only drawbacks I've noted above. Even just the SVN-to-git migration could be a lot of work alone, but adding a needless transition to GitLab is completely unnecessary.

If people aren't willing to register for a Redmine instance; I doubt they'd care to register for a GitLab instance, either. Then we'd also need to migrate accounts and probably reset everyone's passwords, too to the annoyance of existing users.

[1] "visually appealing (IMO)" as you noted does not count. Especially when the registration page is hostile to non-JS users.

#13 - 11/30/2015 01:19 AM - sytse (Sytse Sijbrandij)

GitLab Inc. would be very happy to support this effort in any way possible. We can help with the conversion, add features to GitLab of anything else needed.

Eric Wong wrote:

me@jonathanmoss.me wrote:

Eric, would you want to try Gitlab out with w3m before the switch?

The main site's Terms of Service (or any ToS) is still a problem for me. Currently there is no ToS at all on our Redmine instance and registration is completely non-intuitive on GitLab w/o JS.

1. Is this about the terms of service for GitLab.com or something else? Since I think a self hosted GitLab instance is proposed I think there are no ToS.
2. If better non-javascript functionality is needed for the conversion to happen we would be happy to add that.
3. If GitLab Inc. is acquired by another company (which is unlikely) GitLab the project will still continue. It is in use by more than 100,000 organizations and has over 800 contributors and is MIT licensed.

#14 - 11/30/2015 11:48 AM - rosenfeld (Rodrigo Rosenfeld Rosas)

Em 28-11-2015 20:25, Eric Wong escreveu:

me@jonathanmoss.me wrote:

... There's a bunch of Redmine-specific tools the maintainers use (I haven't checked in depth); and as I said before: Redmine works with git, too. So perhaps migrate to git first and stick with Redmine for now.

Hi Eric, I understand your point of view (even though I don't support it and have no problems with JS being used) but there's something I'd like to bring to the attention here.

I don't think migrating to GitLab (or even GitHub) should change Ruby's workflow significantly. Also, if the Ruby-core team wants to it could even maintain a git mirror in their own servers without using any front-end like GitLab or GitHub.

I still think GitLab and GitHub interfaces are very poor to handle issues management and I much prefer Redmine for that and I think we should keep Redmine as an issues tracker, not because it works on w3m but because it's vastly superior to any other issues tracker I've known.

Having said that, even if the Ruby-core team decides for not maintaining two official repositories (by supporting an official mirror), you can interact with GitHub or GitLab only once, to sign up and send your public keys. I guess you can opt to get any pull requests notices by e-mail so even reviewing them could be done using the git standard tools set.

The workflow with git won't really require any interactions with the GitLab or GitHub UI, unless we move issues tracking to those. But I really think Redmine is a much more powerful tool for handling issues management and I think we should keep using it for that purpose. In that case, maybe it would make sense to disable the "issues" feature from GitLab or GitHub.

Also, it's important to note that one is not attached to any vendor, like GitHub, by opting for it. Changing the main repository should be really easy with Git, anytime we want, so we wouldn't have to stick with GitHub or GitLab or any other option we choose now.

Also, maybe you could get in touch with GitHub and ask them if they would agree in accepting a custom CNAME for the git url so that it would be even easier to move the git url to point to another server in the future... I don't think it would worth because I don't think they would provide us all public keys once we decided to move out from there. But at least it wouldn't break automated recipes using git to download the source as the source url would remain the same...

I'd just like to point out that moving from SVN to Git shouldn't imply in moving from Redmine to other UI for issues management. I'm a big fan of Redmine and would love to keep using it as an issues management tool.

I remember Gitorious didn't provide any issues management interface and they had set up an instance of ChilliBeans (a fork of Redmine) to track their own issues when they were still in development. I really prefer separate tools for handling issues and VCS. Also, it's much simpler to move the VCS repository. It's not that simple to move issues tracking.

Please keep Redmine while moving to Git.

#15 - 12/01/2015 08:58 PM - normalperson (Eric Wong)

web2004@syntse.com wrote:

Eric Wong normalperson@yhbt.net wrote:

The main site's Terms of Service (or any ToS) is still a problem for me. Currently there is no ToS at all on our Redmine instance and registration is completely non-intuitive on GitLab w/o JS.

1. Is this about the terms of service for GitLab.com or something else? Since I think a self hosted GitLab instance is proposed I think there are no ToS.

OK for the proposed instance. But if we find a problem with the hosted

instance and want to report the problem to GitLab itself, the reporter would still have to deal with the ToS of GitLab.com

A bigger problem is none of these centralized messaging systems have any interoperability with each other. Different Redmine/GitLab instances cannot talk to each other; and Redmine instances can't talk to GitLab instances, etc. Bugs do cross project boundaries occasionally...

Plain-text email is still the only interoperable way to communicate. Debian's BTS actually gets this right and interoperates well with existing mailing lists. From my limited experience, RT does as well.

1. If better non-javascript functionality is needed for the conversion to happen we would be happy to add that.

For one, the registration page I noted earlier only shows a bunch of unlabeled input boxes on w3m (which has no JS or CSS support).

Does GitLab have mailing list integration? Fwiw, I would not have bothered contributing to Ruby if we did not have ruby-core mailing list integration with Redmine (but I'm only an occasional contributor to Ruby).

1. If GitLab Inc. is acquired by another company (which is unlikely) GitLab the project will still continue. It is in use by more than 100,000 organizations and has over 800 contributors and is MIT licensed.

It's likely the Gitorious guys were thinking the same a few years ago :)

Really, I remain unconvinced GitLab is any better than Redmine.

What I can comfortably say is git is better than SVN in most aspects:

- robustness/data integrity
- disconnected operation
- speed
- scriptability with a stable plumbing interface
- ease-of-contributing (open-to-everyone mailing list)

Of course git likely loses in terms of usability and learning curve[1], and maybe still doesn't work as well on non-Free OSes.

[1] Side note: I recommend anybody new to git to learn the blob/tree/commit data relationships with the plumbing, first; but maybe that's just me.

#16 - 12/27/2015 11:13 AM - sytse (Sytse Sijbrandij)

Eric Wong wrote:

web2004@sytse.com wrote:

Eric Wong normalperson@yhbt.net wrote:

The main site's Terms of Service (or any ToS) is still a problem for me. Currently there is no ToS at all on our Redmine instance and registration is completely non-intuitive on GitLab w/o JS.

1. Is this about the terms of service for GitLab.com or something else? Since I think a self hosted GitLab instance is proposed I think there are no ToS.

OK for the proposed instance. But if we find a problem with the hosted instance and want to report the problem to GitLab itself, the reporter would still have to deal with the ToS of GitLab.com

If reporting bugs about GitLab by the ruby tema is a problem we would be happy to provide this via email like we do for our customers.

1. If better non-javascript functionality is needed for the conversion to happen we would be happy to add that.

For one, the registration page I noted earlier only shows a bunch of unlabeled input boxes on w3m (which has no JS or CSS support).

Thanks, I've created <https://gitlab.com/gitlab-org/gitlab-ce/issues/4316> for this. We would be happy to fix this.

Does GitLab have mailing list integration? Fwiw, I would not have bothered contributing to Ruby if we did not have ruby-core mailing list integration with Redmine (but I'm only an occasional contributor to Ruby).

We're thinking about this in <https://gitlab.com/gitlab-org/gitlab-ce/issues/4272>

1. If GitLab Inc. is acquired by another company (which is unlikely) GitLab the project will still continue. It is in use by more than 100,000 organizations and has over 800 contributors and is MIT licensed.

It's likely the Gitorious guys were thinking the same a few years ago :)

Of course I can't predict the future but we have over 40 team members, are well capitalized and are close to cash flow break even this month. The situation is not similar to that of Gitorious during their lifetime.

#17 - 06/02/2016 09:43 AM - ferdinandrosario@gmail.com (ferdinand rosario)

I know this feature request already reject but still posting in case if guys are still considering to move git.

my motivation is not that, python is moving so ruby also has move, instead I love to see ruby in git so that people can contribute to ruby easily.

<http://www.infoworld.com/article/3077452/application-development/van-rossum-promises-python-36-will-move-to-github.html>

#18 - 06/06/2016 05:41 AM - duerst (Martin Dürst)

On 2016/06/02 18:43, ferdinandrosario@gmail.com wrote:

Issue [#11741](#) has been updated by ferdinand rosario.

I know this feature request already reject but still posting in case if guys are still considering to move git.

Well, Ruby is already on github (see <https://github.com/ruby/ruby>), and many developers already use git (not me yet :-). See e.g. <https://github.com/shyouhei/ruby/wiki/committerhowto> (for committers) and <https://github.com/shyouhei/ruby/wiki/noncommitterhowto> (for non-committers).

<http://www.infoworld.com/article/3077452/application-development/van-rossum-promises-python-36-will-move-to-github.html>

This article mentions "moving Python repositories from Mercurial to GitHub". I guess the situation for Python (Mercurial) is different from the situation for Ruby (svn/redmine).

#19 - 07/19/2016 09:21 AM - naruse (Yui NARUSE)

Note:

On a Subversion repository imported into GitHub the commit author mail address is username@repository-UUID. Once people can assign such mail address, but GitHub reject people assign such pseudo address now.

#20 - 07/19/2016 09:51 AM - normalperson (Eric Wong)

naruse@airemix.jp wrote:

On a Subversion repository imported into GitHub the commit author mail address is username@repository-UUID. Once people can assign such mail address, but GitHub reject people assign such pseudo address now.

I think svn.authorsProg of git-svn can help map users to @ruby-lang.org addresses easily:

```
AUTHORS_PROG=/path/to/authors-at-ruby-lang.org
cat >"$AUTHORS_PROG" <<\EOF
#!/bin/sh
username=$1
echo "$username <username@ruby-lang.org>"
EOF
```

```
git config svn.authorsProg "$AUTHORS_PROG"
```

```
# Then run "git svn fetch" as usual
```

I haven't used this feature myself :x
but let me + git@vger.kernel.org know if it breaks.

#21 - 07/19/2016 07:42 PM - normalperson (Eric Wong)

Eric Wong normalperson@yhbt.net wrote:

naruse@airemix.jp wrote:

On a Subversion repository imported into GitHub the commit author mail address is username@repository-UUID.
Once people can assign such mail address, but GitHub reject people assign such pseudo address now.

I think svn.authorsProg of git-svn can help map users to
@ruby-lang.org addresses easily:

```
AUTHORS_PROG=/path/to/authors-at-ruby-lang.org
cat >"$AUTHORS_PROG" <<EOF
#!/bin/sh
username=$1
echo "$username <username@ruby-lang.org>"
EOF
```

```
# oops, forgot executable bit:
chmod +x "$AUTHORS_PROG"
```

```
git config svn.authorsProg "$AUTHORS_PROG"
```

Then run "git svn fetch" as usual

I haven't used this feature myself :x
but let me + git@vger.kernel.org know if it breaks.

It will be documented in the next release, at least:
<https://public-inbox.org/git/20160719100927.GA19702@whir/>

#22 - 02/25/2018 02:40 PM - Eregon (Benoit Daloze)

- Related to Feature #14551: *What's missing to switch to Git instead of using Subversion? added*

#23 - 02/25/2018 04:59 PM - shevegen (Robert A. Heiler)

I know this feature request already reject but still posting in
case if guys are still considering to move git.

I think it was said that the net benefit may have been too small
at that moment in time, but if I remember correctly, it was also
said that this can be re-evaluated at a later time. More than
two years have been gone by now. :-)

(If it is decided to switch to git, I think it should ideally
happen before ruby 3.x comes out and probably after mjit has
been smoothly integrated.)

#24 - 01/15/2019 07:38 AM - steve102 (steve smith)

Thumbs up for this incredible discussion of yours. I've truly appreciated perusing this discussion today and I figure this may be extraordinary
compared to other discussions that I've perused yet. If it's not too much trouble prop this work upon
<http://www.onedollarwebhostings.com/1-euro-hosting-domain.html> the equivalent quality.

#25 - 03/16/2019 09:52 AM - mamad (mamad safa)

<http://15news.ir>
<http://9news.ir>
<http://afexo.ir>

<http://aloofile.ir>
<http://aminpayam.ir>
<http://andishmes.ir>
<http://arikehvip.ir>
<http://azargo.ir>
<http://bassirat.ir>
<http://belton.ir>
<http://birligh.ir>
<http://brmm.ir>
<http://chagheri.ir>
<http://fagups.ir>
<http://felfeel.ir>
<http://filegallery.ir>
<http://fnevis.ir>
<http://q2pracles.ir>
<http://harikagap.ir>
<http://iranspeleo.ir>
<http://irnhost.ir>
<http://is20.ir>
<http://j75.ir>
<http://kalakalaa.ir>
<http://kishfairs.ir>
<http://miadweb.ir>
<http://moeinazoafa.ir>
<http://mostak.ir>
<http://newshap.ir>
<http://padidefans.ir>
<http://pears.ir>
<http://popich.ir>
<http://popmac.ir>
<http://ramkaadv.ir>
<http://rejimkade.ir>
<http://rezervbambo.ir>
<http://samatarh.ir>
<http://seablog.ir>
<http://serendypaper.ir>
<http://shikopet.ir>
<http://sna-kala.ir>
<http://sobheyasoj.ir>
<http://takvinarman.ir>
<http://tepstone.ir>
<http://topsoftz.ir>
<http://zanich.ir>
<http://cssschool.ir>
<http://jsschool.ir>
<http://tpitgroup.ir>
<http://appmail.ir>
<http://8host.ir>
<http://digi220.ir>
<http://digi103.ir>
<http://seotis.ir>
<http://seotis.com>