



 This repository

[Pull requests](#) [Issues](#) [Gist](#)




[Watch](#) 4,832 [Star](#) 34,727 [Fork](#) 13,644


[Code](#) [Pull requests](#) 106 [Pulse](#) [Graphs](#)

Add support for AR5BBU22 [0489:e03c] #17

Closed **WNeZRoS** wants to merge 1 commit into `torvalds:master` from `unknown repository`

 Conversation 188

 Commits 1

 Files changed 1

+3 -0




WNeZRoS commented on May 11, 2012

+ 

No description provided.

  Add support for AR5BBU22 [0489:e03c]

757a6e0

 WNeZRoS closed this on May 11, 2012



 torvalds commented on May 11, 2012

Owner

+ 

I don't do github pull requests.

github throws away all the relevant information, like having even a valid email address for the person asking me to pull. The diffstat is also deficient and useless.

Git comes with a nice pull-request generation module, but github instead decided to replace it with their own totally inferior version. As a result, I consider github useless for these kinds of things. It's fine for *hosting*, but the pull requests and the online commit editing, are just pure garbage.

I've told github people about my concerns, they didn't think they mattered, so I gave up. Feel free to make a bugreport to github.

Linus

...

 33

 31

 7

 8

 8

 1



orblivion commented on May 11, 2012

+ 

How do you feel about merging in things that may include commits downstream that have been pull requested with github? Seems hard to stop that.



jaseemabid commented on May 11, 2012

+ 

Somebody please look at the diff. Thats a simple 3 line code addition. I agree to you @torvalds but you could have excused this time :)

jaseemabid commented on May 11, 2012



By the way, its quite funny that github is sending instructions to @torvalds on using git.



torvalds commented on May 11, 2012

Owner



On Fri, May 11, 2012 at 1:03 PM, orblivion ***@***.***> wrote:

How do you feel about merging in things that may include commits downstream that have been pull requested with github? Seems hard to stop that.

Read my email.

I have no problem with people using github as a hosting site.

But in order for *me* to pull from github, you need to

(a) make a real pull request, not the braindamaged crap that github does when you ask it to request a pull: real explanation, proper email addresses, proper shortlog, and proper diffstat.

(b) since github identities are random, I expect the pull request to be a signed tag, so that I can verify the identity of the person in question.

I also refuse to pull commits that have been made with the github web interface. Again, the reason for that is that the way the github web interface work, those commits are invariably pure crap. Commits done on github invariably have totally unreadable descriptions, because the github commit making thing doesn't do *any* of the simplest things that the kernel people expect from a commit message:

- no "short one-line description in the first line"
- no sane word-wrap of the long description you type: github commit messages tend to be (if they have any description at all) one long unreadable line.
- no sign-offs etc that we require for kernel submissions.

github *could* make it easy to write good commit messages and enforce the proper "oneliner for shortlogs and gitk, full explanation for full logs". But github doesn't. Instead, the github "commit on the web" interface is one single horrible text-entry field with absolutely no sane way to write a good-looking message.

Maybe some of this has changed, I haven't checked lately. But in general, the quality of stuff I have seen from people who use the github web interfaces has been so low that it's not worth my time.

I'm writing these explanations in the (probably vain) hope that people who use github will actually take them to heart, and github will eventually improve. But right now github is a total ghetto of crap commit messages and unreadable and unusable pull requests.

And the fact that other projects apparently have so low expectations of commit messages that these things get used is just sad. People should try to compare the quality of the kernel git logs with some other projects, and cry themselves to sleep.

Linus





torvalds commented on May 11, 2012

Owner + 🗨️

Btw, Joseph, you're a quality example of why I detest the github interface. For some reason, github has attracted people who have zero taste, don't care about commit logs, and can't be bothered.

The fact that I have higher standards then makes people like you make snarky comments, thinking that you are cool.

You're a moron.

Linus



15



13



6



skalnik commented on May 11, 2012

+ 🗨️

@torvalds The GitHub commit UI provides a text area for commit messages. This supports new lines and makes it easy to do nicely formatted commit messages :)



jedahan commented on May 11, 2012

+ 🗨️

@skalnik would be nice if it had an 80-character line to help format things nicely.



paulcbetts commented on May 11, 2012

+ 🗨️

Every time another Pull Request fiasco happens on one of Linus's repos it makes me sad, especially because I want someone whose work I greatly respect, to have a good experience on GitHub - instead he gets dozens of troll comments.

An OS kernel very rightfully demands a very disciplined approach to development that is in many ways not compatible with the goals of GitHub, which is to get as many people *of all skill levels* involved in Free / Open Source Software. We can certainly make improvements though, and I appreciate that Linus has taken some time to detail exactly *why* he doesn't use PRs, even if it's a bit harsh.



11



tubbo commented on May 11, 2012

+ 🗨️

- no sane word-wrap of the long description you type: github commit messages tend to be (if they have any description at all) one long unreadable line.

I think this is only because people who are new to Git are using GitHub and not understanding about Git-style committing. Remember, a lot of these newbies are just out of the gate from using SVN for years. I bet a lot of them don't even realize that `git commit` with the "-m" omitted just opens up COMMIT_EDITMSG in your editor. It isn't even very apparent (to newbies) of the 50-char title rule and 72-char every other line rule with commit messages.

github *could* make it easy to write good commit messages and enforce the proper "oneliner for shortlogs and gitk, full explanation for full logs". But github doesn't. Instead, the github "commit on the web" interface is one single horrible text-entry field with absolutely no sane way to write a good-looking message.

I have to agree with you there. Commit message viewing on Github sucks and I hope they change it soon.



torvalds commented on May 11, 2012

Owner + 🗨️

On Fri, May 11, 2012 at 1:29 PM, Mike Skalnik ***@***.***> wrote:

@torvalds The GitHub commit UI provides a text area for commit messages. This supports new lines and makes it easy to do nicely formatted commit messages :)

No it doesn't.

What it supports is writing long lines that you have not a f*cking clue how long they are. The text area does not do line breaks for you, and you have no way to judge where the line breaks would go.

In other words, it makes it very hard indeed to do "nicely formatted commit messages". It also doesn't enforce the *trivial* "oneliner for shortlog" model, so the commit messages often end up looking like total crap in shortlogs and in gitk.

So the github commit UI should have

- separate "shortlog" one-liner text window, so that people cannot screw that up.
- some way to actually do sane word-wrap at the standard 72-column mark.
- reminders about sign-offs etc that some projects need for project-specific or even legal reasons.

It didn't do any of those last time I checked.

Linus



2



jedahan commented on May 11, 2012

+ 🗨️

I always thought of the title of a pull request as the one-liner ...



jrep commented on May 11, 2012

+ 🗨️

Newbie question I know, but can someone point me to this "nice pull-request generation module" Linus mentions? My google fu, documentation fu, and command-line-help fu all failed.



torvalds commented on May 11, 2012

Owner + 🗨️

On Fri, May 11, 2012 at 1:40 PM, Tom Scott ***@***.***> wrote:

- no sane word-wrap of the long description you type: github commit messages tend to be (if they have any description at all) one long unreadable line.

I think this is only because people who are new to Git are using GitHub and not understanding about Git-style committing.

The thing is, even if you *do* understand about git-style committing, it's actually *really* hard to do that with the github web interface.

The best way to do it is *literally* to open up another text editor for the commit message, and then cut-and-paste the end result into the web interface text tool.

Yes, commit messages should have proper word-wrap, with empty lines in between paragraphs, and at the same time sometimes you need a long line without word-wrap (compiler error messages or other "non-prose"

explanation).

And yes, that would almost require some kind of "markup" format with quoting markers etc. And yes, it would be a more complex model of writing commit messages. But if the default is "word-wrap at 72 characters, put empty lines in between paragraphs", then people who don't know about the markup would still on average get better results (even if the word-wrap would then occasionally be the wrong thing to do)

Right now, github simply seems to default to "broken horrible messages", and make it really really hard to do a good job.

And I think it should default to "nice readable messages" with some effort needed for special things.

Linus



technoweenie commented on May 11, 2012



@jrep: I believe he's referring to [git-request-pull](#).



nugend commented on May 11, 2012



I'm not sure I understand why the commit message itself should be hard word-wrapped. Naively, it seems like that should be a display property of the editor used to write the commit message or the tool used to display the commit message.



torvalds commented on May 11, 2012

Owner



On Fri, May 11, 2012 at 1:48 PM, Dominik Dabrowski ***@***.***> wrote:

You might have fun raging on the internet, but I think your goals would be better served if you expressed your thoughts in a clear (maybe even polite) manner that doesn't embarrass the people whose actions you're trying to influence.

Umm. I think I've been able to reach my goals on the internet better than most people.

The fact that I'm very clear about my opinions is probably part of it.

If people get offended by accurate portrayals of the current state of github pull requests, that's *their* problem.

I hate that whole "victim philosophy". The truth shouldn't be sugarcoated.

Linus



20



scommma commented on May 11, 2012



While I do have great respect for you @torvalds and your work, and it's totally valid for the repository of Linux to have rather rigorous standards, have you considered the possibility there could be a lot of GitHub users who don't really need nor care about any of those "features" you try to portray as *objectively* superior?



torvalds commented on May 11, 2012

Owner



On Fri, May 11, 2012 at 1:49 PM, Daniel Nugent ***@***.***> wrote:

I'm not sure I understand why the commit message itself should be hard word-wrapped. Naively, it seems like that should be a display property of the editor used to write the commit message or the tool used to display the commit message.

No it shouldn't.

Word-wrapping is a property of the `*text*`. And the tool you use to visualize things cannot know. End result: you do word-wrapping at the only stage where you `*can*` do it, namely when writing it. Not when showing it.

Some things should not be word-wrapped. They may be some kind of quoted text - long compiler error messages, oops reports, whatever. Things that have a certain specific format.

The tool displaying the thing can't know. The person writing the commit message can. End result: you'd better do word-wrapping at commit time, because that's the only time you know the difference.

Sure, the alternative would be to have commit messages be some non-pure-textual format (html or similar). But no, that's not how git does things. Sure, technically it could, but realistically the rule is simple: we use 72-character columns for word-wrapping, except for quoted material that has a specific line format.

(And the rule is not 80 characters, because you do want to allow the standard indentation from git log, and you do want to leave some room for quoting).

Anyway, you are obviously free to do your commit messages any way you want. However, these are the rules we try to follow in the kernel, and in git itself.

And quite frankly, anybody who thinks they have better rules had better prove their point by showing a project with better commit messages. Quite frankly, I've seen a lot of open-source projects, and I have yet to see `*any*` project that does a better job of doing good commit messages than the kernel or git. And I've seen a `*lot*` of projects that do much worse.

So I would suggest taking the cue for good log messages from projects that have proven that they really can do good log messages. Linux and git are both good examples of that.

Linus



3



tylermenezes commented on May 11, 2012



If you add `.patch` onto this URL you'll get a git-am style patch.

(Github is very silly for not exposing this in the interface, and for not even really mentioning this feature.)

I agree with you on the messages, I wish the text areas were at least monospaced.



torvalds commented on May 11, 2012

Owner



On Fri, May 11, 2012 at 2:01 PM, Prathan Thananart ***@***.***> wrote:

While I do have great respect for you @torvalds and your work, and it's totally valid for the repository of Linux to have rather rigorous standards, have you considered the possibility there could be a lot of GitHub users who don't really need nor care about any of those "features" you try to portray as _objectively_ superior?

Sure.

And when those people with lower standards try to get their commits included in the kernel, I will ridicule them and point out how broken their commit messages or pull requests are.

Agreed?

Btw, the commit message rules we use in the kernel really *are* objectively better. The fact that some other projects don't care that much is fine. But just compare kernel message logs to other projects, and I think you'll find that no, it's not just "my opinion". We do have standards, and the standards are there to make for better logs.

Linus



2



torvalds commented on May 11, 2012

Owner



On Fri, May 11, 2012 at 2:03 PM, Mahmut Bulut ***@***.***> wrote:

So, if you can't "impolite" dear @torvalds, we can say 'why the "linux kernel" is here'?

.. because I think github does some things *very* well.

So sure, you may think I hate github. I don't. I hate very specific parts of github that I think are done badly.

But other parts are done really really well.

I think github does a *stellar* job at the actual hosting part. I really do. There is no question in my mind that github is one of the absolute best places to host a project. It's fast, it's efficient, it works, and it's available to anybody.

That's *wonderful*!. I think github is absolutely lovely in many respects.

And that then makes me really annoyed at the places where I think github does a subpar job: pull requests and committing changes using the web interface.

Linus



3



1



mmorris-gc commented on May 11, 2012



Word-wrapping is a property of the *text*. And the tool you use to visualize things cannot know. End result: you do word-wrapping at the only stage where you *can* do it, namely when writing it. Not when showing it.

Just curious - why is it that the tool used to visualize things cannot know how to wrap text it displays? And if it is the case, isn't that a problem with the viewer itself, rather than a reason to hard wrap?



myfreeweb commented on May 11, 2012



Commit messages must be limited to 140 characters, like tweets. Right in git's core.
(See what I did there? What's "pure garbage" for you is just perfect for a lot of people.)



vertexclique commented on May 11, 2012



@torvalds Thank you for your rational and good opinion. I appreciate you.



brettalton commented on May 11, 2012



Do you guys not understand that this is *Linus'* blessed repository and he can accept and reject whomever and whichever request he likes? He has specific and pertinent rules when it comes to merging that he's learned over *20 years* of maintaining the Linux kernel. He developed git - in case you forgot, he was the initial developer - with features specifically for gpg signoffs, shortlogs, etc. - things he and other intelligent computer scientists find useful for maintaining repositories.

I've maintained small projects with three developers plus myself and as soon as you become loose with your merging criteria, the entire repository goes to hell. If he wants gpg signoffs, then he'll get gpg signoffs. Try maintaining 20 millions lines of code and merges requests from 2,000 developers, and then you can give Linus advise.



dustalov commented on May 11, 2012



I think @torvalds is a pretty cool guy. eh scolds githubs and doesnt afraid of anything.



MostAwesomeDude commented on May 11, 2012



While I do have great respect for you @torvalds and your work, and it's totally valid for the repository of Linux to have rather rigorous standards, have you considered the possibility there could be a lot of GitHub users who don't really need nor care about any of those "features" you try to portray as objectively superior?

"GitHub is the best place to share code with friends, co-workers, classmates, and complete strangers." As long as GH actually, genuinely cares about making this statement true, they should be providing these features.

Roman, in the future, you should follow the kernel's guide for submitting patches. I believe that drivers/bluetooth is covered by the list at linux-bluetooth@vger.kernel.org and you can submit your patch to them, with a proper Signed-off-by tag.

FWIW, Reviewed-by: Corbin Simpson MostAwesomeDude@gmail.com, but there's no way to confirm that since GH is going to hide my email address and I can't easily sign this message.

(As an example of broken UI, while writing this message, I split my screen between Firefox and vim, vertically. Linus' messages, being wrapped, were perfectly readable, but because Github has a massive minimum width, I had to scroll back and forth in order to read everybody else's messages.)



ivyl commented on May 11, 2012



@mmorris-gc

Sure, tools can do that, but at what cost?

Mostly messages are read in terminal, not via web interface.

How to distinguish part which should be wrapped from ones that don't? Add extra tags?

Commit logs are mostly viewed in terminals, which tends to use monotype fonts.

What about quoting? ">" are clean and indicates level of quoting.

This ideas are used for years in emails and guess what? They work!



factormystic commented on May 11, 2012



@mmorris-gc It's open source. Fork it and write a custom viewer for yourself. Problem solved.



mephux commented on May 11, 2012



Amen for the "victim philosophy" comment. If you want to commit or suggest features get ready for feedback. People need to seriously stop crying when others are blunt with them; It's pathetic. (not everyone has time to consider the infinite ways you may interpret something)



KorvinSzanto commented on May 11, 2012



I'd have to say I fully agree with @torvalds, I've worked in very strict commit standards, and in very loose standards, and by far my entire experience was a lot better with well formatted standard commit messages. Github does not handle this at all.

Some say that "people don't care", it's mostly because they don't know what they are missing, if it were more convenient to use good standards, everyone would use them.



jite commented on May 11, 2012



Sometimes I wonder if the ones who like a massive one-liner as commit message are Windows users...



1



mmorris-gc commented on May 11, 2012



@ivyl

Sure, tools can do that, but at what cost?

I don't know what the cost is, but I'd be interested to know! That's why I was asking what prevents the tool from doing this rather than requiring that the user handle it.

@factormystic Not sure what this has to do with my question. I was just wondering if there was a reason that the viewer couldn't handle it; I wasn't complaining or asking someone to fix it for me.



jnavila commented on May 11, 2012



Sad that there is no option to disable pull requests via github



skalnik commented on May 11, 2012



@torvalds It is indeed a text area. 

On top of this, vim/emacs/\$EDITOR does not usually enforce the commit format either. In both cases it's up to the end user to write a well styled commit message.

That being said, I agree it could be better. Perhaps if it was more like the commit form that the [GitHub application](#) has.

Since this seems so important, perhaps git should enforce this style by rejecting any commits with a message that does not adhere to your specification?



camdez commented on May 11, 2012



why is it that the tool used to visualize things cannot know how to wrap text it displays?

@mmorris-gc That was actually covered by @torvalds above when he said:

Some things should not be word-wrapped. They may be some kind of quoted text - long compiler error messages, oops reports, whatever.

Not only would it be a tremendous burden for every viewing tool to try and determine which items meet the above definition (and do so correctly), many of the tools we use are generic whereas the formatting rules might depend what domain the material came from, making it literally impossible to display things correctly under all conditions.



mmorris-gc commented on May 11, 2012



@camdez Interesting. Still seems like a problem that could be solved by better tooling, but I appreciate you taking the time to point that out. Thanks!



leobalter commented on May 11, 2012



@jnavia there's a way to disable pull requests in Github, they call it private repos.

So sad seeing someone who made a great system raging like a child because no one and no system can be like him or how he wants.



antirez commented on May 11, 2012



@torvalds other than "form" of pull requests what I'm even more worried about is that this new model of contributing code bypasses the former interaction that there is in a mailing list. If the hub of a project is the ML there are better chances that things are discussed before turning into code that will be refused. Even when the approach starts with a patch, it gets publicly discussed by interested parties, and a long term trace remains in the ML archive. It's a pretty different way of doing this, that was used to build a lot of code with success, and one that works better for a project where patches and new ideas are scrutinized in depth before being accepted.



JeremyARussell commented on May 11, 2012



@torvalds I would like to take this opportunity to say thanks for Linux and git. For without both of those this great coding community wouldn't have had a chance.

I'd also like to point out something else GitHub does do really well. This. What we are doing right now. Socially coding in an open environment. Talking about things, being connected. Hell when I was growing up I never thought I'd get a chance to say something that Linus effing Torvalds would get to read and possibly comment on, and now here I am, able to put in my two cents (in a flood of thousands of pennies). So thankyou. Thankyou Linus for making git and Linux, and thankyou GitHub for making coding social.



jnavila commented on May 11, 2012



@leobalter No : disabling pull requests does not mean making a repo private. As many other opensource projects, the linux kernel has its own workflow, so why not follow it? At GH, they are aware of it, they even mention in the progit book.

And before "raging like a child" about his comments, read them again: he just does not care or bother.



evanmoran commented on May 11, 2012



My own preferred solution would be if GitHub kept to one commit message box but live previewed how it would appear below with 72 character wrap. Then you could see clearly what the short and long messages would look and could adjust accordingly (this is done in Stack Overflow and is very helpful).

The last issue is that monospace is required to view / wrap correctly. A natural way to handle this is to use the markdown four space indent syntax, but since this could get annoying it might be better to have an input type pulldown (text vs markdown) in the same way editing GitHub wikis allows.



leobalter commented on May 11, 2012



@jnavila github has its pull requests as they are. Maybe no one follow "high standards" Linus but it's great in my workflow.

My point is: raging like a child is unnecessary. Turn of pull requests notifications and don't answer.

If this github pull requests mess your day off, start thinking about using other code hosting.

The community doesn't need to be blamed for not being such highness standards followers, we just need people collaborating, because it's open and many visions are still great on any project.



drogus commented on May 11, 2012



I'm not sure why this topic is about **pull requests** not the feature of editing files online. Most of the people create pull requests out of branches prepared locally, I've prepared tons of pull requests and I've used online editor only once.



KorvinSzanto commented on May 11, 2012



@leobalter, you're missing the point, this isn't about downplaying the current workings of github, it's about suggesting better workings for github. Just because you are fine with having pull requests on doesn't mean there shouldn't be an option to turn them off.



denhamcoote commented on May 11, 2012



@leobalter He's not blaming 'the community', he's pointing out what he thinks needs improving *in GH*. Raging like a child? If you don't like his 'childish' opinion (read: high standards), don't open a pull request. I'm quite happy to see the conversation that's followed as a result.

I work at a financial institution where a single line code change can be backed with 50 page specs, 200 lines of test code, 2 weeks of testing, etc. Asking for a decent commit message *on your own repo* isn't *that* big of a deal.



nugend commented on May 11, 2012



@camdez Are we talking about *only* the situation where some text shouldn't be word wrapped though? Are there other wrapping related formatting concerns with plain text?



SixArm commented on May 11, 2012



I agree, especially the identify verification via confirmed email addresses, digital signatures, or a mix.



orblivion commented on May 11, 2012



@torvalds I think you missed my point. I'm not just talking about people using Github to host. You don't merge everything in Linux yourself, you defer 90% of that through a trust hierarchy (as you eloquently described in your Google talk about Git). Unless you somehow enforce that everybody under you also refuses Github pull requests, your logs could still get soiled.



jsanders commented on May 11, 2012



@antirez How is the discussion of a pull request on GitHub different than the discussion of a patch on a mailing list? Is it that you end up with two different places to discuss things - mailing list for things without patches, GitHub for things with patches? Or is it that subscribing to see pull requests for a project is not as elegant as subscribing to a mailing list?

My company has had quite a bit of success having in depth discussions about both experimental and more straightforward patches on pull requests, and treating them as the long term trace of discussion, much like you're suggesting - what would we gain from using a mailing list instead?



torvalds commented on May 11, 2012

Owner



On Fri, May 11, 2012 at 4:12 PM, orblivion ***@***.***> wrote:

Unless you somehow enforce that everybody under you also refuses Github pull requests, your logs could still get soiled.

I'm not a "rules over everything else" kind of black-and-white person.

I'm basically describing what *my* requirements are. Not all Linux sub-maintainers are necessarily as critical as I am, and yes, there are ugly commit messages in the kernel too (and some of them are very much about lacking proper word wrapping, for example).

So things slip through occasionally. I'm not German - rules are good, and they set a standard that people should really try to strive for (and quite frankly, hopefully exceed: the "formatting rules" should preferably go with "really good and readable message that really explains what is going on"), but rules are not some kind of absolute thing that have to be 100% guaranteed.

In the kernel, see commit [cb8722d](#), for example. That's a case of "oops, that's one long line". It happens, and I got it through David Miller, who usually doesn't have those kinds of issues. I suspect the patch came from somebody who used an annoying editor or MUA that has problems wrapping lines properly (sometimes, you have to disable word-wrap in the MUA for it to not corrupt patches, but then some MUA's have a horrible editor that doesn't help you wrap lines when you want to!).

So I don't worry about "still get soiled". Crap happens, we try to minimize it.

What I dislike about the github thing is that it's not "crap happens, we'll try to minimize it", it's "crap absolutely **WILL** happen". Instead of trying to minimize it, the commit message editor actively **reveals** in it, and makes it hard not to make a crappy message.

Similarly, the pull request interface of github makes it literally **impossible** to make a good pull request. You literally cannot make a good pull request using the github web interface.

So right now, I encourage people to use github as a hosting site, but as a hosting site **only**. Don't create commits there, and don't use github for pull requests. Do your commits on your own machine, push them to github, and then when you're ready for a pull request, again do it on your own machine and email it to the maintainer that way.

So I'm really not trying to hate on github. I only despise a few of the small details of github.

Github as a hosting site for open-source (or closed, for that matter) projects is **wonderful**.

Github as a place to generate commits and pull requests? Not so much.

Linus



dysoco commented on May 11, 2012



@johnmetta Oh, you must be new to the internet, or to @torvalds rants :P



braneed commented on May 11, 2012



Linus, I love your rants and your code. @torvalds.



SkaveRat commented on May 11, 2012



I like how @torvalds rants on a high niveau ;)
nice read, and I have to agree (tho the "moron" comment really wasn't necessary)



jespada commented on May 11, 2012



Worth read : <http://tbaggery.com/2008/04/19/a-note-about-git-commit-messages.html>



sirlancelot commented on May 11, 2012



Did you see about adding `.patch` to the end of the pull request URL like so:
<https://github.com/torvalds/linux/pull/17.patch>

I'm no git-expert, but doesn't that have all the information?



holdenweb commented on May 11, 2012



Not sure what all this fuss is about. @Torvalds points out that due to definite weaknesses in GitHub's UI he won't accept pull requests, and the world starts whaling on him. It's simple: if you want him to pull your

changes in, don't use GitHub to generate the request. This would probably be easier than trying to change his mind.



torvalds commented on May 11, 2012

Owner + 🗨️

On Fri, May 11, 2012 at 5:05 PM, Jorge Espada ***@***.***> wrote:
Worth read : <http://tbagger.com/2008/04/19/a-note-about-git-commit-messages.html>

Yes, that has all the basic rules.

In the kernel, we have the extra sign-off rule (and "Reported-by"/"Acked-by" etc to give credit to people who have been involved in other ways than just moving the patch around), so some projects will have extra rules about what a commit message should contain.

But the basic rule about "one-liner header" (which really is used by a lot of tools) and then nice word-wrapped textual bodies is really good.

We *often* have 20-50 lines of explanation for even a *single* line of changes. Sure, it can be overkill, but on the other hand it really can come in very useful too. And git makes it so easy to make shortlogs and other "condensed" versions, that it should never be a question of "I don't want to bother people with too much information".

Linus



torvalds commented on May 11, 2012

Owner + 🗨️

On Fri, May 11, 2012 at 5:05 PM, Matthew Pietz ***@***.***> wrote:
Did you see about adding `.patch` to the end of the pull request URL like so: [#17.patch]()

I'm not git-expert, but doesn't that have all the information?

Umm. I read *email*. I don't look at web interface.

And no, I don't want the patch either (it might be thousands of lines long).

Look here for a good example of a recent *valid* pull request:

http://groups.google.com/group/linux.kernel/browse_thread/thread/c3de7bbe9bb73cf5/1d61f01ea9ec3c67?show_docid=1d61f01ea9ec3c67&pli=1

where that pull request contains:

- the real person with a real email asking me to pull
- the *explanation* of why I should pull
- a shortlog of the changes (a single line)
- a proper diffstat

it doesn't have silly links to other information, it *has* the information.

Linus



javajosh commented on May 11, 2012

+ 🗨️

If the rules for writing good commit messages are that mechanistic, then @skalnik made a good suggestion: provide a way for maintainers to specify a validation function on commit messages. Could be a nice feature.



fogleman commented on May 11, 2012



Why is Linus word wrapping his comments in this thread? Looks silly.



SkaveRat commented on May 11, 2012



On 12.05.2012 02:35, Michael Fogleman wrote:
Why is Linus word wrapping his comments in this thread? Looks silly.

Reply to this email directly or view it on GitHub:
[#17 \(comment\)](#)

because he's writing replies via email, which, with the right client,
gets auto wrapped (like this reply)



reinaldons commented on May 11, 2012



I fully agree with [@torvalds](#). GitHub is a UI that replaces an important feature with inferior version, have no excuses.



sp4ke commented on May 11, 2012



I think github devs should really take [@torvalds](#) remarks seriously for two main reasons.

Git was built for kernel and designed by [@torvalds](#), so even if it might not seem important for some new people to git and github, there is a reason for git commit messages and pull request to respect some rules which might not be evident, and Github has a great responsibility in teaching these rules to new comers.

Second, Gh might be the best tool/platform to start using Git, so with all the possibilities given by a modern web service like Github and how easily UI can be tweaked, it seems like a waste not to build on top of best practices. Seriously how hard is it to make a text field validator for respecting git commit messages ?



rtomayko commented on May 11, 2012



I just want to get on the record as one of the original pull request developers that we've been aware of these issues for a long time and certainly take them seriously. There are a number of problems we need to address that would make maintainers's lives a lot easier.

There's no question Linus's feedback is warranted. I could add considerably to his critique, even. (Mail headers anyone?) Nobody hates these issues more than we do.



luckydev commented on May 11, 2012



[@torvalds](#) is very clear on what Linux needs if the contributors wants to send him pull requests. I think Github should just look into this and fix the problems.



AlekseyKorzun commented on May 11, 2012



Github was made for 'easy & fast' code management, perhaps this is not the right tool for this job.

I don't agree that there should be rigid pull request standards in place, it works fine for 99% of the smaller projects that are hosted here.

The ticketing system on other hand.. is another story.



petdance commented on May 11, 2012



(not everyone has time to consider the infinite ways you may interpret something)

I can think of only one way to interpret Linus telling someone "You're a moron." There is no subtle nuance there.



Bilge commented on May 11, 2012



Sure is my way or the highway in here.



pirtlj commented on May 11, 2012



My comments keep getting deleted lol



pirtlj commented on May 11, 2012



I hate that whole "victim philosophy". The truth shouldn't be sugarcoated.

By truth Linus is of course referring to his own opinion.



holdenweb commented on May 11, 2012



No doubt his manner is abrupt. Possibly curt. Probably rude. It's fairly obvious @torvalds doesn't "suffer fools gladly". For all I know, this may be a necessary strategy, though it's certainly not one I would find productive. But luckily for him, I'm not him. I do know prominent open sourcerers who are, I wouldn't say harassed, but certainly imposed upon mercilessly. The "nice guys" end up conscientiously dealing with at least some of the traffic, which takes up time that could be spent working or with their families and friends.

But the most important points have almost been lost in the noise: a) @torvalds made explicit complaints about the github pull request, with cogent reasons why it was unsatisfactory; b) Github responded (nice to know they watch their logs) explaining that they are aware of the shortcomings, and others not mentioned, and are working towards fixing it.

It's pointless to argue and bicker
Linus doesn't respond to a clicker
So just make a note
He's a crabby old goat
And then we'll all get along quicker

The Miss Manners conversation can now continue :)



ghost commented on May 11, 2012



Why is it possible to commit using the web interface anyway? I agree that it's difficult to write decent commit messages using the web interface, but it's even more difficult to write decent *changes* using the web interface.

People often either don't have knowledge of the organization of the project, or they are half asleep when writing the changes using the web interface.

Anyway, this discussion about wraps is getting me hungry.

n3storm commented on May 12, 2012



I started reading this thread with no opinion. Then I realised I started reading @torvalds nice newspaper column like posts with ease and skipping non wrapped texts, and said "uhm, that's the point!"

So now I do have an opinion, *all* comments should be line-wrapped.

Thank's for the lesson :)



sitaramc commented on May 12, 2012



@torvalds You have a lot more patience than I have. I rarely even log in to github (website); once in a while I'll go in and just blindly delete all the pending pull requests unless I recognise the name of the person.

I've long had a policy of "no pull requests, no issues, no comments on code via github; everything on email only" and if people don't know that it's their problem.

[edited to change "tolerance" to "patience" in first line]



JonDum commented on May 12, 2012



@sitaramc Maybe the solution is for Github to convert actions that were created with the web interface into an email friendly format and send it out like it does for comments. That way all parties are satisfied regardless of which interface they prefer (web or terminal).



jammycakes commented on May 12, 2012



In defence of Linus's attitude here:

There is one thing you need to bear in mind about the Linux kernel. It is an operating system kernel -- the most fundamental, critical software component of your entire computer. If it goes wrong, everything goes wrong. On top of that, it is probably the most widely deployed OS kernel in the world, being used from everything from transport to logistics to medicine to the military to aerospace. Many of these are applications where people could be killed if things went wrong.

A system of that nature requires much more care and attention to detail than your average vim setup or pet weekend IOC container. If that extends to issues as seemingly trivial (to some people) as word wrapping on check-in comments, then so be it. And if the lead developer of a project such as that does get sharp with people, it's not unfriendliness and political incorrectness, but simply due care and attention with regards to the bigger picture and the stakes being so much higher.



JackieJ commented on May 12, 2012



@AlekseyKorzun, "easy and fast" development is based upon code that's easy to manage. Loose pull requests really hurts efficiency in code management, no matter for big project or small project. A rigid pull request standard would make the development easier and faster, especially for projects involving multiple contributors:).



sitaramc commented on May 12, 2012



@JonDum sounds like a nice idea but consider this sequence. I get an email from the website. I reply to it, cc-ing someone outside github. The original requestor (who is interacting only via github's web interface) sees my reply and replies to that. At this point I believe the guy I added in my CC does not get cc-d and is out of the loop.

At least that's my recollection of this; maybe they fixed it...

It's not hard to fix; qa.debian.org does it ok I think. So does bugzilla, IIRC and probably many other such systems.



robermorales commented on May 12, 2012



I think that if @torvalds does not like github, he can move "his" project to another site. I like github web interface. We are on 2012, not on 1980. Probably Linus use a 80-char green-on-black display. The real people mostly not.

Furtherhand, @torvalds cannot say "you are a moron" while his minions clap. It is hateful.



jaseemabid commented on May 12, 2012



@robermorales He explained already why he hosted "his" project here in a very sensible manner. Read comments.

There are quite a lot advantages using the 80 chars convention even in 2012.

He is the only reason why we at least have a sensible "80-char green-on-black display", respect him for that.



richo commented on May 12, 2012



@robermorales move "his" project?

You're here because you use git right.. who specifically do you think wrote git?



AnthonyAkentiev commented on May 12, 2012



Linus wrote: "For some reason, github has attracted people who have zero taste, don't care about commit logs, and can't be bothered."

It seems like github is written using C++ :-))



yobert commented on May 12, 2012



I'm fascinated by how many comments say things along the lines of "why not enforce the commit message formatting" or "github should add validation to the commit message tool". The point is that you can't validate or enforce good formatting, since only the author knows which parts of the text should be wrapped nicely and which parts shouldn't.

On a side note, doing a good text editor in a web browser that looks nice and works well is very very challenging.



greg0ire commented on May 12, 2012



@yobert no, you can't (or should) not enforce the commit message formatting, but you sure could validate it. Look at how vim does this with colors when you use it as your commit editor.



bootc pushed a commit to bootc/linux that referenced this pull request on May 12, 2012

mm: compaction: check pfn_valid when entering a new MAX_ORDER_NR_PAGE...

9da11af

...



MattKetmo commented on May 12, 2012



@n3storm @SkaveRat : I don't share your opinion about auto-wrapped email.

First, I feel sick reading those auto-wrapped comments in this thread because I always make a "stop breath" after each end of line. It's not natural, especially with non-monospace font. Imagine reading a book where text is not justified and text width is 3/4 of the page. That'd be weird.

It's not your responsibility to format the text displayed in a web page. If you feel uncomfortable with reading all other comments, maybe the designers at GitHub should change the font-size or the `.discussion-timeline` div width. The only reason is if you read that thread on your terminal... which leads to my second point.

My second point is : **I don't read my emails on a terminal!**, like most of people. I hate when somebody send me (wrapped) text-only email that are very hard to read on a smartphone (need to scroll horizontally), and often looks weird on the web interface, like **@fogleman** noticed it here.

That could be compared to a width-fixed content like PDF which is not adaptable (/responsive) for all supports, whereas HTML is.

On the other hand, most people use a terminal or a monospaced text editor/IDE to code and use git. That's why I totally agree on wrapping message commits, but not in emails or in comments.

So my opinion is: don't use a web UI as your main tools to make commits -- most people work locally anyway, this is just here to provide a quick (and crappy, ok) solution when you don't have your usual working environment. But Github pull requests are really awesome and are much more user-friendly than emails.



dysoco commented on May 12, 2012



Wait a minute, I'm going to invite Tanenbaum to this conversation.



poke commented on May 12, 2012



@torvalds

Look here for a good example of a recent *valid* pull request:

http://groups.google.com/group/linux.kernel/browse_thread/thread/c3de7bbe9bb73cf5/1d61f01ea9ec3c67?show_docid=1d61f01ea9ec3c67&pli=1

To be fair, pull requests on GitHub are not that different to that. Pull requests (and issues, which are very related) here are a replacement for mailing lists. It's where the discussion is going on. All the data you mention is available in a pull request as well, just not *that* visible. Instead you have to look at the commits appended to the request, or the diff view that's next to the discussion tab.

Obviously that's not how you do things. You are an email person, using mailing lists as the main (if not only) way to discuss and propose changes to your projects. And I think that is perfectly fine, especially looking at how well it works with your projects.

But I don't think that makes pull requests on GitHub inferior. They are different, yes, they require a different workflow, but that workflow works extremely well for many projects, especially those that are not using other means for communication (like mailing lists).

A bad style for commit messages or reasoning of commits and pull request does not come automatically with pull requests on GitHub, the same way as a good style does not come automatically with mailing lists. I've seen many perfectly described commits in a well-reasoned pull requests on GitHub, and I've also seen as many bad requests in mailing lists. You can do both good and bad things with either (or any) system, and I personally think GitHub offers a great system for projects that are not as busy as the kernel or Git itself.



kbarber commented on May 12, 2012



Github should supply a mechanism for disabling pull requests from the Admin interface, so these conversations aren't required and people like **@torvalds** can make his own decisions on how he wants to receive commits. At the moment, pull request capabilities are always on so one needs to constantly close

them, explain to people why, rinse & repeat.



n3storm commented on May 12, 2012



@MattKetmo, just to give you a clue, ever wondered why you prefer reading books in a tablet or pad and not in a 29" screen?
Even in year 2030, human eyes will have the same comfortable eye scanning and skimming range, proportionally to font size, of course.



benatkin commented on May 12, 2012



Just add a feature to disable pull requests on a per-repo basis, GitHub. Since it's an *option*, it doesn't need a majority of users to want it, to justify adding it.

It's also a good first step to take in fixing pull requests.



MattKetmo commented on May 12, 2012



@n3storm Sure, I agree with you about having a "comfortable eye scanning and skimming range".

I'm just saying breaking lines at 80 chars for that kind of content doesn't solve the problem (except in a terminal). Text should be displayed at fullwidth in a pad, and in a column of "xxx" px max in a 24" or 42" screen. Line breaks won't be the same depending on the support, so manually breaking lines can make reading harder.



ostenlag commented on May 12, 2012



Maybe @github could add a per-repo option to enforce the @torvalds recommended commit messaging style?



SteveJones commented on May 12, 2012



I can't believe this whole discussion has gone on with no mention of `format=flowed`, probably the best thing Apple ever did. Not that it really applies to the question of how git commit messages should be formatted (or does it?), but you shouldn't be commenting on formatting of plain text emails unless you've read that rfc.



shepik commented on May 12, 2012



I love it that you can easily know just by looking at word wraps of the comment whether a person is supporting Linus or not



team aqua commented on May 12, 2012



For future reference, here are the HN discussions on this topic:

<http://news.ycombinator.com/item?id=3960876>

<http://news.ycombinator.com/item?id=3964252>

344 items not shown

[View more](#)



oelmekki commented on Oct 10, 2015



@nkeck720 It's a bit hard to imagine that this one who is a great programmer, can write drivers without specs or kernel level code and is willing to contribute to linux kernel is not able and willing to learn how to use git itself extensively to achieve this :)

Also, the linux kernel is probably way past the point where maximizing contributors is an issue.



conradjones commented on Oct 10, 2015



How about a great amazing programmer who isn't prepared to use bash, GCC, make etc. Should Linus accommodate building the kernel in visual studio?

...



nkeck720 commented on Oct 10, 2015



I can see where you are coming from. What I am saying is though that there are people out there who may have never heard of git. When they start using git, the github interface is what they learn to use instead of the terminal interface. While they may be new to git, they still know C inside and out. That is my point, that at one point or another everyone was new to git but knew at least what they were doing in their programming.

...



nemobis commented on Oct 10, 2015



nkeck720, 10/10/2015 18:31:

everyone was new to git but
knew at least what they were doing in their programming.

False. I exhibit myself as counterexample: proper usage of git and code review tools allowed me to learn coding, from being a total coding n00b.



kaber commented on Oct 10, 2015



Am 10. Oktober 2015 17:43:36 WESZ, schrieb nemobis <notifications@github.com>:

nkeck720, 10/10/2015 18:31:
> everyone was new to git but
> knew at least what they were doing in their programming.

False. I exhibit myself as counterexample: proper usage of git and code

review tools allowed me to learn coding, from being a total coding n00b.

Reply to this email directly or view it on GitHub:

[#17 \(comment\)](#)

Guys could you take me off this crap please? I'm not the slightest interested and have no idea why I'm CCed at all.

--

Diese Nachricht wurde von meinem Android-Mobiltelefon mit K-9 Mail gesendet.



alias-mac commented on Oct 10, 2015



@kaber, you are receiving updates on this because your commit in:

kaber@ 4533703

has (in the commit message) #17, so github relates that automatically and therefore you are automatically subscribed to the same thread you referenced (in this case this one).

I complained to github about a very similar issue: I was receiving notifications from things happening on the forked repo that I don't care about, since I'm not maintaining them.

You need to manually click the unsubscribe button (on the bottom right part) in the #17 page.



0day-ci pushed a commit that referenced this pull request on Oct 11, 2015

usb: dwc2: gadget: fix a memory use-after-free bug ...

44583fe



quant67 referenced this pull request in MopTym/gojuon on Oct 13, 2015

description: make description appropriate #2

Merged



0day-ci pushed a commit to 0day-ci/linux that referenced this pull request on Oct 21, 2015

Btrfs: add a check of whether fs_info->fs_root is NULL in btrfs_async...

a5a4ec2



0day-ci pushed a commit to 0day-ci/linux that referenced this pull request on Oct 26, 2015

reporting stuck on s390 in dso__split_kallsyms_for_kcore ...

aa5b596



0day-ci pushed a commit to 0day-ci/linux that referenced this pull request on Oct 26, 2015

reporting stuck on s390 in dso__split_kallsyms_for_kcore ...

5fe286e



jhofstee added a commit to victroneenergy/linux that referenced this pull request on Nov 10, 2015

ARM: 7668/1: fix memset-related crashes caused by recent GCC (4.7.2) ...

a982c99



0day-ci pushed a commit to 0day-ci/linux that referenced this pull request on Nov 10, 2015

xfs: give all workqueues rescuer threads ...

7a29ac4



This was referenced on Dec 17, 2015

Drivers Description #226

Open

Change device tree like sunxi's .fex file #224

Open



joshix referenced this pull request in coreos/docs on Dec 28, 2015

README: change pull request description #708

Merged



sajadtorkamani added a commit to sajadtorkamani/dotfiles that referenced this pull request on Jan 16

Use editor to enter commit message rather than using the the -m flag ...

317493b



Ismael-VC commented on Jan 29



And when those people with lower standards try to get their commits included in the kernel, I will **ridicule them** and point out how broken their commit messages or pull requests are.

Wow, really!? I respect @torvalds work, but I can't see how anybody could respect him, I guess that as long as his arrogance and his geniusness are not mutually exclusive, lots of people don't care at all and even idolize him for that. 😞

Even you Torvalds were a n00b at some point, and your rudeness and willingness to publicly ridicule other newbies is totally unnecessary, why not just *point them how broken their commit messages or pull requests are* and link them to your standards guide (which I agree is a good standard) if you are taking the trouble to answer or just ignore them instead?



pixelrebel commented on Feb 1



...his arrogance...lots of people don't care at all and even idolize him for that.

I, for one, would be honored if Linus took the time to call me a moron.

why not just point them how broken their commit messages or pull requests are and link them to your standards guide...if you are taking the trouble to answer

Then golden threads like this one wouldn't exist. :)



Ismael-VC commented on Feb 1



I, for one, would be honored if Linus took the time to call me a moron.

Good for **you**, but the point is that not all the people would be willing to take insults from **anyone** with *pride* like you and some others may do.

Insulting over the internet is super easy, insulting back is also super easy. I wonder if he would also still be at least that rude with each and everyone of *those people with lower standards that try to get their commits included in the kernel* in person ...at fist range?

So much for a **benevolent** dictator for life... if such a thing even exists.



mucamaca commented on Feb 1



I am pretty sure that **you** wouldn't tell him all this in person..



leonklingele commented on Feb 1



Can you please stop spamming my inbox with this useless junk? Thank you



Ismael-VC commented on Feb 1



Sure, why not? He is just a man like you and me. Are you implying that you would also be OK if he insulted you? And it doesn't mean that I would be disrespecting him. Not all the people likes to be disrespectful and I would also ask him first if he is still like that and why (you know ...talk), if he still does, in that case.

I'm just wondering, because I cannot truly believe that each and everyone he insults just lets him be, either because they idolize him or *fear* him?

I would expect at least one of them to fight back, that's all, I'm done with my comments here, I think I'm pretty clear of my opinion.



alias-mac commented on Feb 1



@Ismael-VC, I think you are seeing things in the wrong perspective.

It seems to me that you assume wrong things and if you read carefully above, Linus explained that in the first comment. He got upset when other ppl jumped into the thread and said "open an exception", which I must agree, makes **no sense** to open an exception just because it is 3 lines of code...

There are standards that Linux project is following and all of that is well documented already, thus making statements/comments like those are just a waste of Linus and other ppl's time. If the maintainers of the project (and community) already spent their own time to write those standards and to share them with the community (giving it for free), I strongly believe that all we (including n00bs) should do is: RTFM!

I often see n00bs on forums asking things that could clearly be found by researching a little bit more. It is a my time vs your time problem.

Seems like that you did the same mistake and judged the entire thread.

I would probably burst too and would have the same reaction when I read the comment:

[#17 \(comment\)](#)

Don't you think that was being disrespectful? I do 😊

@leonkingele, please unsubscribe the thread (it is easier to press that button than it is to make a comment here).

Bengt referenced this pull request in jaypelps/git-blame-someone-else on Feb 9

Pretend to be Linus e5cfe4b



nemobis commented on Feb 21

I'm surprised there isn't a link yet, so here it goes: https://mako.cc/writing/hill-free_tools.html

torvalds pushed a commit that referenced this pull request on Feb 25

KVM: arm/arm64: vgic: Ensure bitmaps are long enough ... 236cf17

mdjurfeldt added a commit to mdjurfeldt/linux that referenced this pull request on Feb 28

KVM: arm/arm64: vgic: Ensure bitmaps are long enough ... fe1d5e3

Noltari pushed a commit to Noltari/linux that referenced this pull request on Mar 3

KVM: arm/arm64: vgic: Ensure bitmaps are long enough ... d62cca1

barracks510 referenced this pull request in src-its/ca-web on Mar 10

content: git_forking.md: Content describes GitHub and not Git. #15 Closed

fedux pushed a commit to fedux/linux that referenced this pull request on Mar 19

KVM: arm/arm64: vgic: Ensure bitmaps are long enough ... e720b3b

Noltari pushed a commit to Noltari/linux that referenced this pull request on Mar 20

KVM: arm/arm64: vgic: Ensure bitmaps are long enough ... b29de09



























This was referenced on Mar 21

Create gitcommit template Deltaworx/machine-setup#1 Closed























Add client-side hook to validate commit messages qTox/qTox#3089 Open



























sashalevin added a commit that referenced this pull request on Apr 11

nfs: skip commit in releasepage if we're freeing memory for fs-relate... a3a32bd


- ...
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  nfs: skip commit in releasepage if we're freeing memory for fs-relate... f10731a
- ...
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  nfsd: nfsd_open: when dentry_open returns an error do not propagate a... e7bac3a
- ...
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 9b9aea2
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 80a1788
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 9a2f0d2
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  nfsd: nfsd_open: when dentry_open returns an error do not propagate a... f63540d
- ...
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... dd4d4b5
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  nfs: skip commit in releasepage if we're freeing memory for fs-relate... 5fd842a
- ...
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 02b550b
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  nfsd: nfsd_open: when dentry_open returns an error do not propagate a... e327508
- ...
-
- ★  sashalevin added a commit that referenced this pull request on Apr 11
- 🔗  nfs: skip commit in releasepage if we're freeing memory for fs-relate... a59aa34
- ...
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 6c9251a
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  nfs: skip commit in releasepage if we're freeing memory for fs-relate... 54b55e5

...


-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  nfsd: nfsd_open: when dentry_open returns an error do not propagate a... 72acb33
- ...
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  nfs: skip commit in releasepage if we're freeing memory for fs-relate... f72a6a1
- ...
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 3ea3656
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  nfs: skip commit in releasepage if we're freeing memory for fs-relate... 6a16174
- ...
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  nfs: skip commit in releasepage if we're freeing memory for fs-relate... 55dad63
- ...
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  nfsd: nfsd_open: when dentry_open returns an error do not propagate a... ed470d6
- ...
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 9d24636
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 8793be1
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  nfsd: nfsd_open: when dentry_open returns an error do not propagate a... ffd6d60
- ...
-
- ★  sashalevin added a commit to sashalevin/linux-stable-security that referenced this pull request on Apr 28
- 🔗  mm, vmscan: Do not wait for page writeback for GFP_NOFS allocations ... 457ce28
-
- ★  0day-ci pushed a commit to 0day-ci/linux that referenced this pull request on May 20
- 🔗  f2fs: fix to update dirty page count correctly ... ca052c4
-

-
- ★  Oday-ci pushed a commit to Oday-ci/linux that referenced this pull request on May 20
- 🔗  f2fs: fix to update dirty page count correctly ... 0f3311a
-
- ★  Oday-ci pushed a commit to Oday-ci/linux that referenced this pull request on May 26
- 🔗  BCM4331 reset leads to wl lockup ... 7fdc0ea
-
- ★  Oday-ci pushed a commit to Oday-ci/linux that referenced this pull request on May 28
- 🔗  x86: Add early quirk to reset Apple AirPort card ... e3c7188
-
- ★  FichteFoll referenced this pull request in divmain/GitSavvy on Jun 1
- Commit message: Visually represent first line limit of 50 characters #423** [Open](#)
-
- ★  Oday-ci pushed a commit to Oday-ci/linux that referenced this pull request on Jun 6
- 🔗  mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes ... 339981d
-
- ★  Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request on Jun 7
- 🔗  mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes ... 85761d0
-
- ★  Oday-ci pushed a commit to Oday-ci/linux that referenced this pull request on Jun 8
- 🔗  mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes ... 60a8d58
-
- ★  jpoimboe pushed a commit to jpoimboe/linux that referenced this pull request on Jun 8
- 🔗  x86/quirks: Add early quirk to reset Apple AirPort card ... 625a99d
-
- ★  Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request on Jun 9
- 🔗  mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes ... b3c4d9e
-
- ★  l1k added a commit to l1k/linux that referenced this pull request on Jun 11
- 🔗  x86/quirks: Add early quirk to reset Apple AirPort card ... fac9f31
-
- ★  l1k added a commit to l1k/linux that referenced this pull request on Jun 11
- 🔗  x86/quirks: Add early quirk to reset Apple AirPort card ... 4113c30
-
- ★  l1k added a commit to l1k/linux that referenced this pull request on Jun 11
- 🔗  x86/quirks: Add early quirk to reset Apple AirPort card ... 48d0972
-
- ★  FadeMind referenced this pull request on Jun 12
- The description shouldn't contain a example, when the example is directly below. #296** [Open](#)
-
- ★  Oday-ci pushed a commit to Oday-ci/linux that referenced this pull request on Jun 15
- 🔗  mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes ... a29ee07

- ★




0day-ci pushed a commit to 0day-ci/linux that referenced this pull request on Jun 21




mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...

49ddb05
- ★




0day-ci pushed a commit to 0day-ci/linux that referenced this pull request on Jun 23




mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...

3fb0216
- ★




Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request on Jun 24




mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...

24bb18e
- ★



Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request on Jun 27



mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...

2b3111f
- ★

This was referenced on Jul 10

simplified getting previous and next node in red-black tree for vmalloc #302

might bug in file_system or bash #301

Update ahash.c #300

Serial: 8250: Add 7173 (WCH CH355 4S card) #287

cpu/hotplug: handle unbalanced hotplug enable/disable #291


Open

Open


Open

Open

Open
- ★




Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request 29 days ago




x86/quirks: Add early quirk to reset Apple AirPort card ...

abb2baf
- ★




Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request 27 days ago




mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...

d8f64b3
- ★




Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request 26 days ago



mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...


a460acf
- ★




lenovouser referenced this pull request 24 days ago

fixed error message to use decimal instead of hex #305

Open
- ★




Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request 21 days ago



mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...


a3028d3
- ★




lenovouser referenced this pull request 19 days ago

u8pdating #306

Open
- ★



Reichl pushed a commit to Reichl/linux-odroid that referenced this pull request 18 days ago



mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes

...

c3d31d2

Labels

None yet










Milestone

No milestone

Assignees

No one assign


119 participar



and others


Notifications



You're not rec from this thre

★  lenovouser referenced this pull request 17 days ago

Add --output option to perf-top #307

Closed

★  0day-ci pushed a commit to 0day-ci/linux that referenced this pull request 16 days ago

  mm-frontswap-convert-frontswap_enabled-to-static-key-checkpatch-fixes eb062e9

...




Write Preview

AA B i “ <> 🔗 ☰ ☷ ☹ ↶ @ ★

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

 Styling with Markdown is supported

Comment

💡 **ProTip!** Add comments to specific lines under [Files changed](#).



Labels

No labels yet

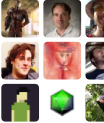
Milestone

No milestone

Assignees

No one assigned

119 participants



and others

Notifications

You're not receiving notifications from this thread