A better extension experience -- extensions rebooted
Sriram Ramkrishna

Neil McGovern: Hi, Sri. Are you around? Had [trying to get audio]

Sriram Ramkrishna: This is a better extensions experience, extensions rebooted, and I'm your host, Sriram Ramkrishna. So, let's start.

So, let's start up a little bit just to level-set. And what is extensions? Right? So, extensions are essentially JavaScript. Or written using the JGS which is our JavaScript bindings for the GNOME platform. And what they do is they extend the GNOME Shell through a process called monkey patching. This is so that we can add, or people could add things that normally is not in the experience of GNOME Shell. Now, the term "Monkey patching," I'm going to kind of give a slightly inaccurate visual on what monkey patching is. But think about a car going down the street at speed. And people changing parts out.

And replacing them, adding new things while it's still on the highway. So, that's -- and if that sounds scary to you, then, yes, that's exactly why this is also scary. So, but new code is -- and more technically, new code is inserted into running code. And you pretty have the entire codebase open to you. There's no private or public code. And anything can be modified and used. Just like that car -- car on the road.

So, extensions can break. If anything changes, that could mean not just the shell itself, but the platforms underneath it, the libraries underneath it. And it's not intentional. Code changes. And so, it gets reflected. So, that's -- that's a short on what the extensions are. And a little bit about the extensions community. I would like to thank Yuri who provided me with some of the statistics. We have 500,000 installed extensions. That's through the browser.

We don't track how many are just down loaded or anything like that. Or how it's coming through your distro or any other methodology used to get extension. This is just through the website and through the extension. You can see there's 321 using Chromium to do this. And then 134,000 using Firefox. I don't know why one is more popular than the other. But this is over a 4 year period.

I actually expected the number of extensions to be more for some reason. Given the amount of noise out there. I expected more. But in any case, and so the some other numbers, you have top extension president top four is user themes at 310,000, applications menu, 248,000. Dash to dock is 218, and Windows List is 168. One tool thing is of the four, three of them is officially supported by GNOME. So, that's great. And then dash to dock is the third-party software. So, once again, thank you, Yuri, for providing those numbers. So, moving on.

What's our situation right now, right? If -- and this is sort of why this initiative got started. So, one, the most popular extensions tend to break at the end of each release. And a lot of those extension owners, they don't even know that their extension's gonna break until we release or their distro migrates to new GNOME version. They -- they're somewhat ignorant about the development cycle of GNOME Shell.

So, just one day, they said it's broke. Somebody says it's broken and then they take their time to get it done. But it's not there at release day. The community itself doesn't know who is responsible. If you look at the community, they'll be like, well, thing GNOME is responsible. And we're not really responsible. We're partially responsible in the sense that there's no guidance. But -- but as far as the community is concerned, this is our doing.

And then authors, there's not -- there's not a huge pool of documentation. We've actually seen a flurry of YouTube videos showing how to do extensions and things like that. But a lot of it people just look at the older extensions or they look at the code to take a look.

And then I thought this was kind of odd. But the most common place authors go for help is Reddit. Not exactly my favorite platform to ask questions. Especially on something like extensions. So, but in my estimation, right? This from a community perspective, from what we see in terms of -- of issues, this is a -- I don't want to use the perennial -- but it's a repeating problem that happens every time around release. And up till now, we've sort of let it fester. Because unaddressed. But as you can see, there's like -- if there's 500,000 people using extensions, and some portion of that are out in the weeds. That's not good for how brand or anything else.

So, and we should do something about that. So, to mitigate this, we really want to -- it's to improve what's happening today. And that solution is not just a technical solution, right? But it's a combination of many things. So, the solution needs to be both about engineering process. It needs to be about policy. It needs to be about great documentation. And -- and really, we want to create a sustaining community to handle this. So, that we can take pressure off GNOME Shell developers in doing this, right?

So, this is kind of like the high-level objective for this initiative. Is to get the pressure off of GNOME Shell developers. But the best way to fix a bad situation is through leadership. It is absolutely essential that we lead on this because there's nobody else who can. So, that's -- so, that -- now we'll talk about it a little bit. But what the goals of the extensions rebooted initiative is. Some short-term goals. So, we have -- we just started working and building a CI pipeline to do basic load tests. We actually were able to run GNOME Shell headless. And be able to test a load on extension, check to see if worked or not. So, a lot of that part is complete. We have some upcoming challenges in regards to building a container that will take code from -- from Git, right?

So, there's no point using a distro supply GNOME Shell here. We need to be able to do this from a just released codebase. So, that's gonna be a challenge. Building that CI pipeline container is gonna be a challenge. But the general idea that we can do a load test on a headless system, that -- I think -- I feel like that's a lot of the way forward. We are thinking about providing guidelines for extension developers. I'll talk a little bit about that from the slides. Feedback mechanism for extension developers. Give them feedback on the things they write. Create a brain trust. We want to think about, as a matter of policy, centralizing extensions on the GNOME GitLab. But not necessarily you have to develop your extension on GitLab. That might be a little bit controversial. I will talk about that as well.

And redesign the extensions.GNOME website. Yuri has done work on this already. And so, that seems to be moving apace.

Okay. And then so, on the matter of providing clear extension guidelines. And this has come from

Andy Holmes who has been doing a lot of the reviews. And I would also like to credit Neil for also reviewing extensions. Both of them put a lot of work did -- and if I left your name out, I'm sorry. Partly through ignorance. But a lot of them have put a lot of work into reviewing extensions and making them -- truly the lag between how extensions get approved and put on the website has been slow. And we're hoping that reviews could be faster by providing some good guidelines. But maybe also look at engineering processes.

So, some of the things -- these are things that Andy came up with. Cleaning up your code after your extension has been disabled. Don't download external code from somewhere. And put your extension at risk. Definitely don't be executing binaries inside your extension. Don't do strange custom updates to update your extensions with bypassing our review system. And don't obfuscate your code. Which I think is good. There's a link here that talks about how the -- how the extensions are. So, something to think about there.

There -- I'm going a little faster than I intended here. So centralizing extensions. I wonder if this is controversial or not. For me, this is something that I felt was the best way to -- to create a process, right? If we want to do efficiencies, we have to have a single -- singular place to -- to get our extensions tested through the pipeline. Have ways to work with each other. Be able to look at each other's code in a way. And there's all kinds of other benefits. If we get our CF pipeline working the way that we could even test how many extensions break on a code commit, right?

And not because that -- that's important, but rather we can get some idea where code is being touched on an extension, right? So, if we're dumping a piece of the code in GNOME Shell and then suddenly we have 15 extensions broken, then we know there is that many people at least using this piece of the code, right? So, I think -- I think those are nice ways to know. Maybe we want to model an API. So, you get some idea of risk, right?

There are some burdens on this that I recognize. We might file issues on the GNOME GitLab where welfare to look at two big bug databases. They may have to change their release process to use our GNOME GitLab. But, again, I'm not -- I'm not insisting that people develop extensions on the GNOME GitLab. But, you know, because it's Git, you can -- you can move it anywhere, honestly.

So, in this case, I think over time, the benefits of centralizing extensions in one place is going to be something that's gonna be beneficial in the end. So, this -- this is -- this is -- this is probably -- I'm not completely married to this. But if somebody has a better -- better methodology, I'm totally willing to think about that. But as a matter of policy, I think centralizing extensions. Also, policy around, you know, what can go up on the extensions website if they don't pass the test, things like that.

But I fully intend to get the GitLab as a gateway to the extensions at GNOME.org. This is a way to makes sure that our quality and release process is understood to extension writers. And where they stand on that. Especially if they're not going to maintain their -- their extensions anymore after a period. Then I don't see why a broken extension should be on our website. I do understand that there are complications with LTS and extensions that used to target that. So, understanding the different versioning and what we target with extensions is something we have to think about also going forward.

So, community improvements. And we can't really do this on our own. It's well understood that we don't have resources. This is -- this is where building a community and having them manage

them is gonna be key. But to do that, we have to build -- the onus is on us to -- to build it, right? To build the community. So, one, we want to improve the atmosphere for everyone. Which, you know, a friendly place where there are experts to be able to talk to. Have very good collaboration between extension writers and GNOME Shell developers. Better API documentation if such a thing is really possible. Going back to my comment about monkey patching. Those are -- those are things that we can think about. But not sure we have a plan going forward. A channel for them to congregate. Real-time chat.

Maybe a place in our Discourse to talk about it. So, these are -- these are great things. Come up with analytics so that -- so, a lot of this is just making sure that we can build a community that can -- can sustain itself. And, again, take pressure off the GNOME Shell developers. And also, we can't ignore the fact that new developers working at GNOME Shell can come become part of the extensions community. I have no doubts that people who work on popular extensions like the dock software or things like that, already know the code -- the GNOME Shell codebase as well as anybody, right?

So, we should also think of it as a recruiting -- recruiting tool to bring more developers into the GNOME Shell community. So, that's a -- that's another bit.

So, on that note, we have a BoF that is on July 26th. And we're gonna talk about the future of extensions with the rest of you. And a lot of it's going to be centered around developer tooling. Definitely documentation. The whole bit about bringing those -- the load tests, working on containers and things like that. And really the idea of testing, which I find really exciting about is that is, is getting better testing, better Q&A, that kind of thing. How to make the website work better for us. And can we or can we not do accessor API brainstorming? Is that something we can do? I realize some that may be a difficult option. But definitely if we can do something like that, that's going to be efficient for us.

So, and so! This is a call to join our initiative. And the caption is, join with us and let us end this destructive conflict. So, we don't have a formal meeting time. We meet every weekend for a few hours. We do have a Matrix channel. Everybody's welcome. We need the skill-sets of every kind. Regardless if you know things about containers or DevOps. How to get Systemd working in a Docker container is -- would be fantastic. I don't -- I -- we have been hitting our heads on this one. I prefer using podman and Buildah, but, you know, the current GitLab CI runner steps seems to be supporting the Docker bit.

Non-developer skill-sets, you know, project management. Investing in the community, doing engagement. Documentation, absolutely. If you are an extension writer we would love to talk to you because something like this should not be done in a vacuum. You can't just be GNOME developers. If we had extension people, we would love to have conversations with all of you. Understand some of the issues you have been having. All of those. All of those things would be -- would be pretty, pretty important.

So, one last slide, I believe. So, I -- the initiative right now is -- there's four of us, actually maybe five. Yuri there. It's Evan, Andy and Ruth. I would like to really thank them for helping work on these slides and having great conversations about how we can improve extensions. So, yeah. So, hang out on our Matrix channel. We're not formal people. So, come on, hang out and let's talk extensions or anything else. All right? Thank you very much. I will hang out and answer questions. I don't know

how much time we have left.

>> I think you can choose three questions of that big, big list of yours. Because there's a lot of them. I think the rest of them --

Sriram Ramkrishna: I haven't even looked. So, I'm afraid.

>> Yeah, because -- we have to -- we have to let go people who actually do captioning. So, yeah. Choose three and the rest of them will stay for BoF, for the second session.

Sriram Ramkrishna: Okay. Let me... good golly! Okay. Evan and Andy will also help answer questions on there. Is moving extension reviews to GitLab in scope? Yes. I -- we have not really thought about that. I would like to -- because we're talking about centralization, I believe that extension reviews should be in scope. I will let the -- what does cleanup code when disabled mean, I'll leave that up to Andy. Let's see. What do I see here...

So far most of these questions are being answered by Andy and Evan. Let's see. Is there -- so, how do we deal with modified versions of GNOME downstream? And I'm gonna say that's -- if not -- we can't -- there's no resources to look at every -- everything downstream modifies. We can only work with the codebase we have. Especially, you know, if you're centralizing everything and doing our QA on the GNOME GitLab. If they brought their own containers, or they brought their own version of GNOME downstream and provided with us a container that we can test against, then they're welcome to do that. But I believe that the responsibility is on them to do that.

So, I think -- I think pretty much everything else is being answered. Oh, somebody said -- are some of the frustrated extension writers optimistic about this initiative? Unfortunately, I have been trying to do an outreach to extension writers and they seem to be a hard group to get a hold of. We are -- we are going to try to reach out to some of them. I'm sure they are frustrated. Given the delays we get in reviews for any of these other things. So, I absolutely believe that this would be -- this initiative would be welcome. Because, again, leadership is -- providing that leadership and having -- having them be invested in it is something that I can imagine that they would not embrace. all right.

>> Okay.

Sriram Ramkrishna: I guess I'm done.

>> Okay. Thank you so much Sri, about this excellent talk. This is -- it seems to be topic really, really sitting on many people's hearts. That kind of wraps up today. Day three at GUADEC 2020. Thank you all for showing up. Tomorrow we will have the last day. But today, after about 45 minutes we will have a first time ever stand-up comedy. And there will be the hilarious comedy show. Please, it's worth it. Yeah, see you around having fun later tonight! And thank you for coming. See ya!

Sriram Ramkrishna: See ya, everyone! Thank you!

[End of captioning for day three -- enjoy the comedy show!]