

GNOME OS on real hardware

Valentin David

Valentin David: Can you hear me?

>> I can hear you fine.

Valentin David: Great. Okay. Okay. Yes. So, I'm Valentin David. I work for Codethink. And today I will talk about GNOME OS on real hardware. What I mean by real hardware is not virtual machines.

So, I'm going to talk about GNOME OS. I have to click here. Okay. When I talk about GNOME OS, the first question is, what is GNOME OS? Sorry about the green screen. I don't have the process for that. This is -- GNOME OS is a bootable image of GNOME. Which are the distribution of GNOME with OS. That is used -- that can be used for release and continuous builds.

That can be available for application developers to be able to test their application in the next release, upcoming release. It can be available for testing the release itself. For the user experience, for the -- it can be nice not to have any branding from the distribution. And also, it can be also nice when we want to test some hardware. And -- and not having the distribution helps also to be able to choose the dependencies we need for this.

The idea goes a bit further. So, this GNOME OS, the idea of GNOME OS is not that young. I can't change the slide. Slides don't change. Is it frozen? Ah, huh. I think I need to -- it's not -- I can't see the slide. Hm. Okay. So... when I go back to...

>> Hey, Valentin, are you having any issues? [no audio from Valentin]

[Waiting for audio connects issues to be resolved]

>> So, I was trying to get to slide four. Yeah. So, GNOME OS, people are still joining. Hope we can get everyone soon. So, GNOME OS goes -- the idea of GNOME OS goes a bit further than that. In 2012, there was talks about trying to get pre-installed GNOME OS on hardware. And more recently in last Linux Application Summit, there was talk from Tobias and Jordan about the Linux, saying GNOME can be a platform when it becomes an OS. Maybe I can put... webcam on. Let's hope it works. Let's see.

There was a first implementation of GNOME OS which is GNOME Continuous. It was done by Colin Walters. GNOME Continuous is several things. It's a tool to build GNOME OS. And it was also the manifest that describes how to build it. It was based on -- it was build with Yocto so it didn't build from bottom. One thing that was interesting from GNOME Continuous is the start of OS Tree and atomic updates instead of imagine managements. There was a question in the chat before with what package manager it uses. And that's OSTree is for GNOME OS. It's not packages.

But there was one issue of GNOME Continuous. It was adding one more set for how to build stuff in GNOME. So, we had the release that was done using JHBuild and then Flatpak, and GNOME SDK, built with Flatpak Builder. And GNOME Continuous. Instead of building different things with different tools and different descriptions, maybe not always matching. The team transformed this into using BuildStream. And there are two talks. You can find videos. One from '17 and one from '18 from Tristan and Michael about the migration of -- of the metadata to BuildStream. And there will be a talk on Friday. You should problem join this one if you have questions about BuildStream and how it's used to build.

So, when there was an integration, the GNOME OS has been redone completely. When I started to test GNOME OS, it was built with BuildStream. Continuous. It was just something that started to work on VM. And but then I started to try to move things a bit forward so that we can go towards something that works on hardware and gets it a little bit more. So, what is there in GNOME OS? So, here I will describe some features about what it does. Maybe you don't know everything. And if you know it, gives you some indications.

So, first, Boots on UEFI with Systemd-boot. So, because it's probably something that works on many machines and it's something that is also a bit more portable. The Initramfs, it's built with dracut. And the graphical boot using Plymouth, this uses the be bgrt theme. This looks a bit like what the design team wants. So, probably they want to look into this and fix it to get exactly what they want. The boot uses Systemd, of course. It's a dependency from GNOME. GNOME core. So, the core applications. But there is no GNOME core applications. There's Flatpak for all the other applications. So, if you want to test an application on it, you would have to use Flatpak.

There's no X -- there's Wayland, but there's no XWayland so X applications can run. This is -- this is important to have exponent because they are still applications using XWayland and lots of applications on Flathub. Even though many support are not enabled. It's something people should look at the application and try to use Wayland. They can support it. It comes with Mesa drivers. Many work. If you have crowd, it will work with it. If you want the NVIDIA proper drivers, we will talk about that a bit later. But it's still there.

Of course, it starts -- the first boot starts with the GNOME Initial Setup. So, the full story of the -- of GNOME. And -- and the first root will also resite the root partition. So, you can just put that on any disk and it will resize.

As I said before, it uses like in -- in GNOME OS uses OSTree for atomic update. There's no automatic updates, we just update. And we have the eos-updater, which can be used by GNOME software so you can make an update nicely in GNOME Software.

So, I wanted to look at trying to boot that on hardware. First I looked at the ARM. And the first thing I tried is I got the Pinebook Pro. It's a laptop which is on ARM which is very cheap. And to me it was a nice thing to just play with to try to see, can I actually boot this on real hardware? I worked on this because I had a Rock 64, on autobot, to port that. And Tom Pollard helped me with the Raspberry Pi4. The Rock 64 and the Raspberry Pi, they are small developers for ARM that are cheap and it's very popular.

So, the first thing that was a bit to deal with is that those boards didn't support UEFI. So, UEFI, for those who don't know, it's like the boot loader that comes with your hardware normally. These don't support that. And it was not possible to boot the GNOME OS images that we had. But those boards all have different bootloaders. So, it's not possible -- it's not that easy to just handle all of them. So, but if you go for more expensive boards from ARM, you will have UEFI support. So, this can work. But it's just -- in those cases, it didn't work. So, for those cases, you have to do something. Is there a used EDK II? A UEFI implementation. Or U-Boot which is more popular and works on more boards and also can boot UEFI.

The -- the firmware is not part of the OSTree. It's part of the image. To boot, we have to put this in the disk. It will never be updated. We need to have different images for the different boards. But this is not a big issue. The OSTree repository will be the same. We have to only handle one. But the problem is it's not always true. But I will talk about that later.

So, in ARM board there is also another thing is that there's something called a device tree. So, we need a distinction of what the hardware is so that we can know what they can use. It can not just discover all of it like that. So, there's something that needs to be described. This is -- on the PC computer, in PC, there is something, there are tables that comes on the computer that the kernel would be able to load and figure out what there is. But for many boards, there is no such thing as this. And we have this thing that is device

trees which is a fuel that comes with the kernel, which is a bit weird thing. And also device tree doesn't really support that very well. They are debts. So, you need to have lots of device trees for lots of different boards which is a bit difficult to handle. But thankfully, this can be hidden by the UEFI firmware. And this can just probably have that Linux kernel. This is something we can hide behind.

But the question is after, how do we handle the updates of this firmware? Because we have something that the manufacturer didn't provide. A way to provide this device tree that we provide. And we would have to handle this. This is something, I don't know. So, another problem is that these boards don't use upstream kernels except one. The Rock 64. Which is a good thing. But many of the boards that get out, they don't work with an upstream kernel and it takes quite a lot of time before it happens. And sometimes never happens. This is really annoying because the OSTree provides one kernel for one architecture.

And having to support a dozen or 20 boards, you would have to have 20 kernels. That means 20 different trees and also 20 different kernel configuration, which is not very nice. So, I think that there are people who are looking at trying to handle that. To have a nice solution. But for now, maybe we should just forget about the idea of supporting such boards. You don't have a mainstream kernel, then we forget about it.

How well does it run? Well, the ARM processor, they are range from very slow to very powerful. And that means that not all ARM processors will work. So, in the case of the Pinebook Pro, and the Rock 64 and the Raspberry Pi, they are cheap things, but it is usable. It's very nice. We had the Raspberry Pi 3 A before and it had too little memory. You need more than 256 megabytes. If you don't have that memory, you probably don't have enough to do anything else. And that doesn't work. So, if you want to try to run GNOME OS, you need memory.

And also, note that those boards we add some VR connection. So, that looked fine and it was usable. The video as well. Playing video was usable. But this is not always -- that might not always be working well.

So, when I submitted for this talk, I just managed to get the Pinebook Pro to work and I was thinking, I will use that until it comes and then I will just report on my -- on my usage. What I managed to do with it. And what I needed to fix to make it work. But the truth is that I don't like this thing because even though it works fine, it's not snappy as a PC. So, what I decided is that I just installed GNOME OS on all the computers I could find. So, I could not use any more, any other distribution.

What I use for several machines that I have at home. But anything that is desktop or laptop, I put on GNOME OS and I was forced to use it every day for all my tasks. So, if there is a bug, I should fix it. Because I can't -- I will be stuck. So, the first thing to do this is something that I started before doing this installation. And starting to work on making an ISO installer.

So, the image provided, it's many forms of them, and you give it to the VM and use it and it's already installed. But if you want to put on the hardware, you would have to unplug the disk and -- and just flash it. And it's not practical. Really not practical. So, I worked on making a bootable GNOME that was less things and just added one application which is Endless OS installer. And the context proof of concept. That was a mistake.

And yes. Endless OS, it's not a very well-finished product. But I'm talking about the installer, not Endless OS. Endless OS is fine. But Endless OS is not that nice. It's based on the GNOME Initial Setup. And it's very simple and exactly what we needed. But I think someone needs to look into that and do a bit better. Maybe from that or go back from GNOME Initial Setup and try to add some support. This is not merged yet. But there's a branch and you can get an image for that.

So, my experience with it. First of all is does GNOME core work? Most of the application worked. Some exceptions here that I have. So, there are things that I fixed. For example, builder needed to have Flatpak Builder. Boxes, depends on the libvirt. And the way libvirt was bullet was missing lots of dependencies so

nothing work would. And Tracker too was broken. So, that means that photos and music were not working. But most of the other applications were working.

There are still some things that are broken that I found. Probably the stuff that is broken, but in my uses, this is what I found. Orca for visually impaired people. So, it's a screenreader. This is not working. And you would need help there. Someone should look at that. I think there is some problem with Wayland. Or other problems. It's not good to have it not working because I think it's a very important piece of software here. Web and video are not working. They are missing some connect because of a issue with patterns and we need to find a way to install -- download and install from Cisco servers.

then I wanted to do this talk through GNOME. But when I tried it today, I just figured out that the microphone didn't work and it was a bit too late to rebuild WebKit. So, now I'm using Firefox. I'm on GNOME OS, by the way, because, of course, you've noticed before I said that, I changed all my computers to use GNOME OS. So, I'm currently using GNOME OS.

Let's see. Next page. There are all the things that I need to fix. The power management was broken. There was missing integration. This is another thing that you don't necessarily see when you use a VM because you don't really have power management. Printing was not working. I mean, the idea of using a VM and, oh, I want to try to print something. You completely forget about printing. But in my professional life, I need to print sometimes.

And I will discover, oh, but we don't have printing. So, there was not that easy. It was a month to get the printers working. There's probably other printers that work, I didn't test that. But I don't buy printers where you really need drivers. So, if you have this kind of purchase, that maybe can be helpful. There were things -- small things like Flatpak portals were missing for GTK. So, Flatpak applications didn't work. For the hardware, lots of things to fix. The Bluetooth wasn't there, the Wi-Fi didn't work.

And the Wi-Fi is a very important fix because if you want to test -- to be able to test things like GNOME Initial Setup on GNOME OS, you need Wi-Fi to have the full story. It's very important. The sound and microphone. Like you can hear me now, I hope. The microphone works. That was a bit not ideal. I have details about, for example, how to get sound through HDMI. Not always of use. Touchscreens. I have two laptops that work with touch genes. Think touch pad is easy. When it's I2 C, it's not that of use. And lot don't work, many don't work. And web Cams. You can see that works. That is being tested right now.

For the development. So, I needed to develop GNOME OS on GNOME OS. So, GNOME OS is build with BuildStream. I needed BuildStream. It's not part of GNOME OS. I make it a Flatpak. This is a problem because Flatpak is not a tool for console applications. This is meant for desktop applications. That means there's lots of missing things. So, I don't know if it's in the interest of Flatpak to fix those things. Or if they are that easy. Consoles are not that easy. Details are very specific.

So, that was a bit of a problem. I needed tools like for programming objdump. This is very, very useful. Gdb and valgrind. What do I do? I get it through the case of Flatpak. And think that I can make this work. But it's not very nice. That's the truth. So, what about Git and Flatpak-builder? Those are dependencies of GNOME Builder. They come in GNOME OS. Because GNOME Builder cannot work without that.

But when we see that, we know that there is something we have to change. So, it is something we have changed recently. So, we have two trees now. Like GNOME continuous add, but we didn't have that in the beginning. First, when you start, you have the user tree. The first image. As everything of core, but it doesn't have GNOME Builder. And if you want to do development, you will have to switch your tree to something that has GNOME Builder. And something with SDK so it has compiler and development times. Headers, manual pages and things like that.

So, for the moment we didn't put BuildStream in this. But I think it should be part of it because GNOME OS is built with it. Another thing that I think we should add, probably, is GHBuilder. Because I think there

are some use cases not covered by BuildStream. I'm not completely sure. So, maybe something.

So, in this SDK, there is no debuginfo files. So, the question is, what do we do with this? And I think we should have servers that serve all the debuginfo files. This is huge. We don't want to have an image with all the debugging forms. It's nice to have them available somewhere so we can download them on demand. So, we don't have that. This is something we should have.

So, now we stop with the GNOME experience and now I want to talk about things a bit what I do with my computer. First thing is gaming. So, there are things that I needed that are in GNOME OS. There's some media files for devices. And so, Gamemode. That is from Feral Interactive, a service that tweaks your CPU performance when you play games. Which is very nice for people that play games. Yes, team devices, VR works. I have to say this quickly, but if you want to play VR on GNOME OS, you have to be ready for a very bad user experience. Because Wayland doesn't support drm-lease. It can not just take over one screen, which is the screen of your VR headset.

So, how something will be fixed there? It can be used. There is for people who do -- we are still missing things to do. I don't know voltage control. I never did that. But at least the fan control is something I have been missing. And since monitoring, we don't have that. But this is not something that's easy to have out of the box. But something that we should have that would be nice. I have only used AMD and Intel graphics. Who are the NVIDIA? If you want to do games, not that good unless you play small games. So, we have glvnd, dispatched to OpenGL, the way that EGL can and opencl. So, it should not be very difficult to add those.

The only thing is we need to be able to deploy software that are not part of the tree. It's the same problem as the video connect for GNOME OM, I think. So, once we have a solution that we can deploy extra layers of things that don't come from there, that would be nice.

how die make these slides? I used a background originally. Or a long time ago. I want to add live into GNOME OS. And using Flatpak is not nice. I built TeX Live with BuildStream with the BuildStream shell. But it's not very nice. We need something there so we don't have the OCI container system. We have a system, and it's important for people who know. But this is even worse to use in this case. It's not really the thing.

I think it would be helpful, but put money on something similar to Docker for people who don't know. But it's not there. And it would be nice. Something, maybe. One last thing is that if you have a key -- security key for signing things and logging, this is added. It's not in by default. But you can enable the user service for that. That I needed that. So, it's still something out of the context of GNOME. Because GNOME itself doesn't use that.

So, how to get it? First, should I use it? Well, first, you have to know that this is still in development. And there's no documentation and it's going to be very difficult to use. Lots of fixes that I talked about before, they are not managed. They are still something that we need to go through. So, there are still things that I said were fix and were broken on master.

And also, it's in master. So, that means it's not a release. So, it's not available. At the time you update, you will have things that will break. As a virtual machine, yes, soon. Once we have things a bit setup. But it's moving and it's getting there. So, people will be able to use that for testing. Should you use it on hardware? I don't think so. Unless your point is to contribute to make it work on hardware. If that's your point.

Security-wise, it will be -- it's not very good to do that. Because you -- it's tricky for anyone to know if you have all security patches there were and there we tried to track -- the detail -- I will not talk about that, maybe. But then it's partly not -- there are things that we don't track security.

So, yes. So, soon we will have -- hm. This is not a very well-formulated, sorry about that. We didn't have OSTree repositories yet. We are setting them up right now. In a few days I hope that we get something and then you can get the update instead of building yourself everything manually.

Some build from master you can find in the GNOME Build Made project. There's help in the Wiki. If you need more help, comment the GNOME OS on GIMPNet. But, yeah. For the month it might be a bit difficult. And we are getting it in real-time.

I want to give -- I want to say thank you for the -- to the release team and especially to Abderrahim who have been making lots of big built requests that are done a because I tried to go fast and had a lot of helpful reviews to make this much cleaner. So, very nice. The work I've done in testing GNOME OS on hardware and fixing things that I found on the way is sponsored by Codethink. So, thanks for that.

That was it. So, I think that I caught up with the time and we have --

>> Thank you. Thank you, Valentin, for the talk. And thank you. Thank you, Valentin for the talk and thank you for the shoutout. There are -- under the shared notes you can find the chat. It has been moved to our Rocket chat. And the question on the pad. All the -- we can -- everybody can try to add questions to the pad. Everything is in the shared notes.

Valentin David: Okay. So, question: What package manager does GNOME OS use? I think I answered this question. It's a new package. It's just OSTree. So, you're just -- that's your whole tree for application is Flatpak. There are Flatpaks for web and videos. Can they be shipped out of the box? CodeX be protect? Maybe. I don't know. This is something probably we should ask Michael about that. I mean, I could test also that it works, maybe.

Have you considered development tools like toolbox for example? So, toolbox, I don't know. Might be useful. But I think we need to have that kind of basic Python environment. PEEP. So, there's PEEP in this case, if you have the development case, you should have PEEP. Yeah.

Yes, because there is Flatpak and Flatpak comes with what's called -- program is that -- the new version of BStream will need some other dependencies which are Buildbox which we've wrapped the wrap thing. So, it's called -- so, I'm missing something. So, yeah. Can you -- can we use GNOME OS for marketing? I suppose. Yes. Can we have a little demo, please? Yes. Can we -- ah. Where is my Pinebook. I'm not sure. Let me find that.

So, this is a Pinebook. I hope it has battery, we'll see. It's probably an older version of GNOME OS. It's booting. Just to say I wanted to do a demo on the desktop that I'm currently using. But with Wayland, it's difficult because I was not able to use GNOME Web. I'm not sure if you're discussing with GNOME Web. And you need 0.2, and I have 0.3. So, you think the battery is off. And I don't know where the charger is. So, I will skip to the next question. Is it possible to customize after install via flatback? Only if you have Flatpak? So, you are pinged on to the OSTree. You can have your own OSTree. This is what I do because I need to fix things. And make a request and it goes to the GNOME meta review. This might take time, but I need a fix here. You can do the master. So, you can do like me if you want. This is not that difficult. If you use GNOME 2 or something like that, it's probably very easy. You can compose an update. Someone has an answer.

Compose. Okay. Soon. That would be interesting. Yeah. Okay. But I think that was all the questions. Yeah, I'm sorry about not having a demo. That's the a but the of an issue. I think if I share some -- yeah. I think it's on the -- on the application that I can share. And the entire screen is black. So, it's not something I can show. See if there is more questions? Epic make GNOME OS it possible soon.

Yeah. I can do a recording. That's very easy. I think it works. I think I tested that quickly today. But I didn't see the results. But it didn't complain. Let's see. I can check for some videos -- yes. Should work. So, I can do that and I can post that later.

But what you will see is just GNOME. I mean, there is nothing special. I mean, the only thing is you will see in the -- it's GNOME OS. I can show the update with the GNOME software. That might be interesting. But then you don't see the boot. The reboot. One thing I could have done is used GNOME boxes to run GNOME OS on GNOME OS. The problem is GNOME boxes doesn't support that because it doesn't put in

the UEFI. I'm told it will be fixed. But not today. Maybe next week.

Apps which are also -- this is something I don't have an answer. Maybe. I think the people who own those applications should say whether they want it in Flatpak or in GNOME OS. Because I think it's true. Some applications don't need that. Can we have a build guide? Yeah. I think if you -- I think you need to learn to use BuildStream. It's very easy. So, you have to check out the GNOME Builder and there's a few commands. `vm/image.btsd` and this will build it. Yes. So, GNOME boxes -- yeah. I mean, we could also have OSTree as like the minimum that doesn't come with any application. I don't know. This is something to discuss. I don't have anything about that.

Oh. We don't have a pre-installed Flatpaks. Something we could do. I think in theory. It's possible. Ah, no. On the image, yes. Yes. Because core doesn't come with OSTree. But I think we can -- we can do that. We just have to add them into the AR directory. We needed to a deployment. Find deployment and get the VAR directory for that. But it's possible. It's very possible. Ah. Would be nice. I think if we get help with that. Someone -- I have not tested the peer-2-peer with Flatpak or I think there's also even for OSTree, the updates itself should work as well.

Somehow, but since it's not been tested, I don't care it works out of the box. There's probably some tricks to do. Please read the repeated questions. I'm not sure which one. Before answering. Ah. Please read -- I'm sorry. So, the question is, I will use -- yeah. Sorry about that. How useful would GNOME OS be at testing other software stacks such as PipeWire? I think very useful. I have been trying to test -- I wanted to test Jack replacement library because one thing we want is -- something I do is some recording of -- with MIDI track audio. But I don't want the Jacket Software. PipeWire would be nice. How hard would an OCI/podman environment to add? How hard to add podman? So, I think it would be easy.

I think we have a build of podman using in BuildStream in some Docker image we have for the CI. So, this we can just copy that there and build that. So, we can add podman. Possibly to install -- would it be possible to install OSTree based package manager like `rpm-ostree`? Then we have to have a package manager. The problem is that you need to have the whole system to be packet managed. I think it's a bit difficult to just mix the trees that are existing with the package management and just add packages. I don't think that works.

>> Sorry to interrupt. Time is up. You can continue the discussion in the -- the chat. There is a link in the chat notes.

Valentin David: I was in the chat right now. Didn't have the time.

>> And thank you again for the presentation.

Valentin David: So...

[Break!]