July 22, 2020 - GUADEC2020

Power management infrastructure in GNOME and contemporary developments

Aditya Manglik

Aditya Manglik: Thank you very much, Neil, for the introduction. I just want to check, can you see the slide? Okay. Great. Great. Thank you. Thank you so much for the feedback in the chat. Yeah. Just a quick check, can you hear me clearly? Is there any echo whatsoever? Great. Thank you. Thank you so much once again. Great. All right. So, I'll just wait for the time and then we'll start precisely on the schedule, okay? Or should I start right now?

I'll start on time. Thank you for the feedback!

In the meantime, in case someone is feeling hungry or if they want to get a cup of coffee. Please, you're welcome. And we are going to start on time. Thanks so much for joining GUADEC. It's really fantastic to see you all personally, I guess, once again. And I really look forward to this presentation.

Hi, it's very nice to see you from GNOME. The audio is perfect. I'm just waiting for the time.

Hey, I'm not sure how to put this. But I kind of forgot to connect the charger and my battery is a bit low. I don't know how long I can go. Will I make it until the end? I'll try my best to make it until the end.

I'm not on Chrome. I'm on Firefox. And I was just kidding. I mean, this is -- this is about power measurement and management. So, obviously I connected the charger. But this presentation talks about the basic premise is, what is happening when your battery is draining. And how do you understand what's happening? That was the whole idea. So, I'm sorry for the poor joke if it was. But I thought it was a cool way to get started for the topic.

So, I guess I'll just start, okay? All right. So, here's -- I hope I'm on time. Maybe it's 5 seconds earlier. Now. All right. So, here's a very brief introduction. I'm just going to skip over it. So, I was a 2018 student at GNOME, and worked on implementing the power possible in GNOME-usage. It was exciting and I was mentored by Felipe Borges and Christian Kellner. I learned a lot from them. Thank you so much for the wonderful experience. This is about the project that I did, the findings and what I plan to do ahead.

All right. So, I'm gonna give a quick overview. First, we're going to understand what's a problem statement. I'm going to turn on the webcam if that's fine. Can you see me?

All right. Yeah. Can you guys see me? Okay. Great. Thank you so much. Christian, I learned so much from you. Thank you so much for erg that you've taught me. All right, let's go. So, the first one I'm going to talk about is, what is the problem statement? What are we dealing with here? And second is once we realize what we are dealing with here, I try to dig into what our competitors -- not our competitors, but what our peers are doing. What is vendor doing? Mac doing, in terms of -- [audio cutting out] and how to -- and then finally, we talk about the system architecture. What I want to build and what I think would make an impact in all of our lives.

Finally, talk about GNOME-usage. We're installing this for all your queries. So, let's go. All right.

So, I think all of us have faced these situations. Using it for 30 minutes and the battery is at 50%. Why are the fans running at full speed in I'm using Chrome, checking maximum. I have no clue what's happening in my system, right? And this one is even more perplexing, I would say. I just charged my laptop full last night, and today morning when I opened it up, it's 70%. This is interesting to understand what is happening. Why is my laptop not sleeping when it should be sleeping, right?

So, I like to -- I like to understand this as the fact that why is a process consuming much more energy than the amount of value that I'm deriving from it? Why is the process running on my system that is taking up so much energy cost as much as I'm not even putting into it, right? That should not happen. And this project aims to solve this problem.

All right. So, let's say that if you had this kind of -- if you had this kind of an application which told you, hey, your screen consumed like 90% of your power and your Inkscape work consumed 40% of the power. Wouldn't that be great? You could what the numbers mean, where is this coming from? Right? So with this is the mockup for GNOME-Usage. And credits go to Allan, thank you so much again for your help.

These were the end user use cases. What about developers? How do these statistics help GNOME community and GNOME developers? Why is my application getting so many bug reports about killing the battery? I mean, what is it? Is there a memory leak? Is it an infinite fore loop? What is happening in my application and how do I quantify the power impact of my code? This is a big question. as a developer, I don't think in terms of power. I think in terms of runtime and memory. But I don't think in terms of what is the power? How can I quantify that? How can I actually put a number to the amount of energy that my application is going to consume?

So, this data will give developers the ability to see how and where actual battery power is being consumed, right? And let's take an example of games. All of us love to play games. A game should not ideally be using a lot of heavy accesses. It's a C (U application. If your game is running slow, the objective is it's being bottlenecked by RAM and not by the CPU or the GPU. That is one of the use cases for this kind of data. And this data can also allow system administers better control, better management, everything. Talking about system administers, let's take a look at how this data, useful for our friends in enterprise systems.

So, how can this data be useful for enterprise folks? Whenever we talk about cloud vendors, for example, Red Hat, and IBM and everyone else. Cloud is a data center. And data center means the total cost of ownership. So, I got these numbers, which basically say that the energy cost of running your servers is a 55% of your total cost of ownership. By the way, this total cost of ownership is somewhere in the billions. It's easily billions. So, you are spending almost 500 million on just running your servers. That is a huge amount of money, right?

And then you're also spending so much energy, which is burning so many fossil fuels and everything. So, how about we actually understand where this energy is going? Power attribution. Attribution means this. So, this is what we refer to as power attribution. Attribution means breaking down the power consumption by the devices and the applications which are actually using that power. That is what I refer to as attribution. So, power attribution can enable data-centers to monitor high energy-cost applications. And it can help improve the scheduling across your cluster. Now you can actually schedule applications which need high amounts of power for high performance clusters. Right?

So, then you end up spending your energy budgets much more wisely. And that way, you can save on your carbon footprints. There's a great tool. And enterprise can do all cool things with this data. This data is really useful for everyone involved in our community. Okay. That is the problem. But why is this problem there in the first place? What is the source for this problem?

The source for all power problems is the battery. Your battery is a finite resource, right? At the beginning of the presentation, I was like, I don't have battery and now I do. So, your battery is a source of all your power. And unfortunately, Moore's Law doesn't apply to batteries. Software engineers have always depended on Moore's Law for the huge power gains, for huge performance gains. But Moore's Law doesn't apply to batteries, unfortunately. This is the reason why your smartphone batteries have not grown in capacity at all. You buy one 3 years ago, 3,000 milliamp hours, and today, 3,000 milliamp. Does that make sense? The screen size has grown to 4.5 energies. You're spending more and more energy with the same amount of budget. This is known as the power envelope. It's an industry term that your device is supposed to operate within a particular budget. Let's say that it's 45 watts. So, let's take an example as a tank, as a reservoir, which has some fuel in it. So, power budget can be understood as the capacity of this tank and the ability of this tank to deliver this fuel to your device.

Okay? So, this is the power envelope. And today, if you go to any industry which has consumer-facing systems, mobile devices, laptops, tablets, anything. Power envelopes are your number one design constraint. That is a proven fact. All fancy technologies that you guys want, VR, AR, everything is constrained but power. And solving this problem can actually enable us to develop much more functionality wide-spread devices.

Okay. How does power attribution help solve this issue? Okay, you can do everything with power. But how can power attribution fix this problem? I would say this is a bit of cliche, but obviously you cannot improve what you cannot measure. Please don't kill me for that, it's a management cliche. But works for this concept. And we need to understand where our power is allocated and where they're going. If you're not aware of something, how do we fix this? How do we know what to fix?

Despite immediate user impact in all industries, be it our end users, developers, be it our enterprise customers, power attribution systems and infrastructure is really, really undeveloped. And it's hard to understand why is the system in development in the first place? I have been wondering. I have been thinking about this question for the past 2 years and then decided, okay, let's bother these guys also with this question and get their thoughts on this.

So, one more thing, I would say this is not a good comparison point. But this only serves to say that our peers have understood the importance of this feature. They have understood the importance of power for all modern device development. And they have attempted to close the gap. Linux solution is still there. That's a big challenge in front of us.

All right. So, what is the challenge that we're facing? How do we fix this? How do we actually get this done? Again, one more cliche, hardware is hard. But why is this problem hard? We know that process education run at 3 gigahertz plus. That's a million in a second. Humans run at 1, plus, minus. Some run at 10, some at 1 hertz. Basically, they do -- in one [audio breaking up] -- and I recommend you drink a bit of coffee to increase your processing speed from 1 hertz to 10 hertz. Please forgive me for the poor jokes.

So, humans run at 1 hertz. So, the -- attempt to build the fact that it's very hard to get numbers from a -- a system which is performing a billion operations a second. And interpret them for a human which is running at 1 hertz per second. Does that even make sense? Compare these two guys. They are at -- and sometimes it's quite hard. It's really very -- it's -- is what makes it inaccurate. This was the first challenge. The second challenge is that power is a hardware concept. As a developer, as I said, when you write code, when you write a -- you don't take that, oh, that is going to tack lots of power. We should reduce the power.

No, no. This guy is running and it's going to take enough memory -- right? How do I reduce the run tomb? So, power is considered as a hardware concept. And devices use -- they are used by software simultaneously. For example, your game is in your CPU. It's using your GPU. It's using your display, it's using your memory. So, how do you quantify the power distributed across some components?

And even if you want to quantify, say you have a [inaudible] -- using lots of power. And your GPU, 20, display is using 50 milliwatts of power. And you have 80 milliwatts of consumption. That's one way to calculate it. Here comes the best part. Hardware chips do not measure the information. This is 10, this is 20. Manufacturers do not give out this information.

So, if you have the information in the first place, how are you going to calculate it, right? We see this is an important requirement. But you're not on board with this requirement because of high costs. The only reliable measure polymer team members values are for processer wattage. Which means your CPU power. And only for folks post Nehalem. microprocessors made by Intel after twelve. Only systems made after 2012 expose this value to the user. And we have the available battery charge.

So, we don't even have -- in the first place, right? Okay. So, with that, we don't have -- try to still go deeper into the problem and see what numbers do we actually need? We need for CPU, it depends on your cores, your frequency, depends on the P-states and your core states. Don't worry about it.

And for our GPU, with thousands of cores and high-bandwidth, we are really -- can you hear me? I got -- I don't know what happened. Can you hear me clearly? Hi, can you hear me now?

>> Hi, we can hear you. I just wanted to tell you to turn off the video because the sound is a bit laggy.

Aditya Manglik: I'm really sorry for that. Now is it better -- I don't know what -- all right. Can you hear me now? I'll continue. And please tell me if you cannot hear. Okay. So, yeah, display consumes a lot of -- and we have no clue how it exactly --

Up next we have --

>> One second.

Aditya Manglik: Yes.

>> Can you basically switch off and on again your audio chat connection. It might fix it. It's the button at the below part where -- where you have the headphones are and microphone displayed.

Try connect again.

[Standing by for audio]

Can you hear? Hey, can you hear me? Oh, my god. Thank god. I'm really sorry for this. Thank you so much for waiting on me. I don't know how long I can go on it. But I will speed through it. Please let me know whenever I should stop.

I don't know. I had to restart my router and now it's working fine. The presenter, and as soon as I have that, I'll go through. Skip the case studies for now and try to understand what's the solution to this problem. I'm sorry for all these issues. I checked all this beforehand. But I guess it's Murphy's Law. Whenever something has got to go wrong, it will go wrong.

>> Also, you're almost out of time.

>> Yeah, I understand. It's like 15 seconds now. I don't know. What do you guys say? Should I continue? Or drop it right now?

>> You can continue for a couple minutes. Two, three minutes and then we start the questions.

>> Thank you. Thank you so much. I got the privileges, thank you so much once again. All right. So, I'm gonna skip this. So, basically, these are the hardware devices which need these kind of computations in the first place, okay? And I think one interesting tidbit that I would like to cover here is that until 2016, Apple could only release MacBooks which supported a maximum of 16 gigabytes of RAM. Can you believe it? Why was this so? This was because they were unable to power more RAM than 16 gigabytes. That was due to the DDR3 power requirements.

And that is really -- I mean, it's really amazing to see that everyone as such a big company, they could not solve this issue. This is a serious concern which is facing all manufacturers and programmers. And it's best that we solve it really well.

All right. So, given all these challenges, how do we solve it? The idea solution, I mean, as a logical solution, we would think that you should determine the time spent on the CPU, the input-output, the memory. Okay, say you're spending 10 milliseconds on the CPU and 20 milliseconds on the I/O and 30 milliseconds on the memory. Multiply by the power. 10 milliseconds by 10 millivolts gives you 100 millijoules, right? This is the power for the components. Why is it not as simple as it sounds? First of all, how to compute the time that is spent? Processes need to actually calculate how much time they're spending in each component during each context switch.

So, the process is required to do an accounting update for each task switch along with all the other housekeeping. This infrastructure, I'm not exactly sure if this infrastructure is already present in processing accounting infrastructure and Linux. But that is needed to enable the solution to work. Now comes the second and tricky part. All right. So, we got the time. Now, how do we obtain the power, right? Heisenberg's principle. If you measure the velocity, you change the position, if you measure the position, you change the velocity. If you measure the power of your CPU, you measure the pour power of your CPU.

Say, for example, you say, what are you running at right now? It was sleeping and was consuming 1 volt. And now to answer your question, it has to wake up. It wakes up and says, hey, I'm using 10 watts right now. This is skewing your measurement completely, right? You don't get the right number in the first place. The right way to get this is using a current and a voltage probe and actually getting that value in an oscilloscope. But we can't expect our users to have oscilloscopes dragging around with the system. You think about these by ACPU and data-sheets. But these data-sheets are so conflicting. One data-sheet specifies one UC's as 10 watts, another one says 12 watts. It's the same IC, just from a different manufacturer. It's supposed to do the same thing. How to align the differences and make a sense of this? There's a big challenge in this problem.

So, the friends -- our friends which we saw earlier have managed to solve this problem. What they do is that they collect all this data from the devices running. From their devices. Okay. So, all right. This is a case studies which basically tells how Microsoft, Apple and others are solving this problem. I'm going to skip this part in the favor of finishing this up. I'm really sorry -- you guys can download the presentation. Please mail me for any questions.

Okay. Windows runs something known as the Energy Estimation Engine which is mining all the data on your device for building its power models. That is the short and sweet version of what is written here. Microsoft has stated that 85% accuracy can be achieved for these power values using these software models.

And extremely few PCs in the market have dedicated chips to obtain these power values in the first place. Okay. So, that's Microsoft solution. This is how they present it to the user. The most interesting part is Microsoft doesn't tell you how much your CPU or your memory or your I/O is consuming. Your mail is consuming 6%, Cortana, 6%. They don't consider the hardware parts. That was one part. The second one is the task manager. If you have seen the recent task manager, they're showing you these power usage trends.

>> Excuse me, but you're out of time. Please look at the questions and try to answer the questions and we can go to the next speaker. Please.

Aditya Manglik: All right. Sure. All right. Thank you very much. Okay. So, this is what Windows is doing. And I'm gonna skip to the questions now. The back-end. How they're actually getting these values. And this is what macOS is doing. Here's a bit of detail. Here's what Android is doing. Android is getting it for hardware and software both.

And what I wanted to do, what is GNOME's power measurement system architecture? And data collection and we need to develop power models. Basically, Gaussian distributions. incorporate them in a regression model. This is a multi-armed bandit problem. All right. So, do you agree? Would you guys be willing to give your data? I don't know. And finally, I would like to present all of this in this format.

And before ending I would like to give a shoutout to Felipe Borges and Christian Kellner for really teaching me a lot and for encouraging me for this project. It's a beautiful project. It's a difficult project. And I would like to thank them for this -- and [cutting out] -- actually do this in the first place. And any questions, please. Please share them. I would love to get some questions. And you can always mail me.

>> Do you have any [inaudible] available?

Aditya Manglik: Thank you very much for the feedback. It's really great to see everyone here and the encouragement. That's why I love in community so much. That's what stuck with me with GNOME so far. And it's absolutely amazing to see you all once again. I really hope to meet you again next year.

>> The questions -- see in the public notes of the left.

Aditya Manglik: Yeah. I can see the questions. But I -- one question that I really like is that plugging a dongle like the WhatsApp in between the machine and the wall socket. Okay. So, what you are doing here is you are measuring the total system power using this WhatsApp dongle. And that's combined in one go, your CPU, GPU, memory. Everything in one go. You don't know where the power is going. We want a breakdown of how much each process is consuming, how much the CPU is consuming. Why is my system waking up at night to check my mail when I want to sleep for my really important presentation next day? I don't obtain those statistics using the WhatsApp monitor. That's why we need an attribution mechanism in the system itself.

Another question that I can see it, Freida is working on a daemon application in the background. I'm not aware of the daemon, but I would love to be part of the development. Thank you so much. Thank you so much for bearing with me. I'm really sorry for this issue. And I hope to present this once again. Thank you so much once again. Good-bye.

>> Thank you for your talk, Aditya. Now we're going to the next one which is GNOME OS on real hardware by Valentin David from Codethink.