Wedge 100: More open and versatile than ever



By [Xu Wang](https://engineering.fb.com/author/xu-wang/), [Alex Eckert](https://engineering.fb.com/author/alex-eckert/), [Luis MartinGarcia](https://engineering.fb.com/author/luis-martingarcia/), [Reza Niazmand](https://engineering.fb.com/author/reza-niazmand/)

At Facebook, we put an emphasis on building our data centers with fully open and disaggregated hardware designed by our engineers and open-sourced through the [Open Compute Project](http://www.opencompute.org/) (OCP). By [disaggregating our stack](https://engineering.fb.com/posts/681382905244727/introducing-wedge-and-fboss-the-next-steps-toward-a-disaggregated-network/), we can replace hardware or software as soon as better technology becomes available. This approach has enabled impressive performance gains across the compute, storage, and networking layers.

As the demands on our data centers keep growing, we are continuously pursuing ways to handle data more efficiently and at ever-faster speeds. Our strategy is to build 100G data centers, and Wedge 100, our [second generation top-of-rack network switch](https://engineering.fb.com/posts/203733993317833/opening-designs-for-6-pack-and-wedge-100/), is one of the key components helping us achieve that goal.

Today, we’re excited to announce that Wedge 100 specification has been accepted into OCP. The industry has already built a robust software ecosystem around it, and we hope that this addition to the community will accelerate the pace of innovation and allow others to bring 100G to their data centers as well.

In this blog post, we'll outline the current hardware and software ecosystem, and we'll share some of our experiences using Wedge 100 in production.

**Wedge 100 at Facebook**

At Facebook, we use Wedge 100 in our production environments, and we continue to deploy it at scale across our data centers. It is a component of our 100G data center network strategy, while also allowing us to maintain backward compatibility with existing 40G devices. On the software side, we continue to use [FBOSS](https://engineering.fb.com/posts/681382905244727/introducing-wedge-and-fboss-the-next-steps-toward-a-disaggregated-network/) and [OpenBMC](https://engineering.fb.com/posts/1601610310055392/introducing-openbmc-an-open-software-framework-for-next-generation-system-management/), our own networking stack and baseboard management implementations, which gives us the flexibility to iterate quickly and introduce new features and innovations in our network.

Creating Wedge 100 was a complex endeavor, but we were able to iterate on the hardware and software with confidence because of our learnings from creating and deploying Wedge 40. We reused many of the same hardware components and made key changes to address pain points we felt with Wedge 40. On Wedge 100, we also run the same FBOSS software as Wedge 40, but we expanded it to support the new platform with 100G ASIC chips and optics.

**Hardware updates**

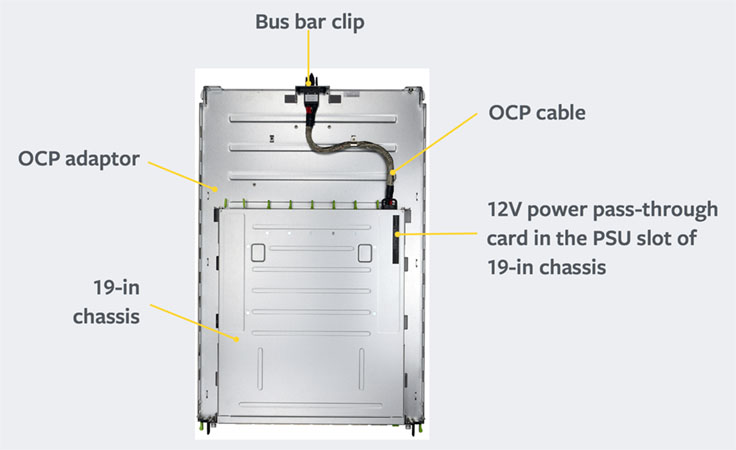
We built Wedge 100 with a COM-Express Type 6 module in the compact form factor (95mm x 95mm) as the microserver; this bodes well for 100G and higher speed switch platforms. Wedge 100 also supports COM-Express module form factor (95mm x 125mm), which allows for a flexible design in case more advanced CPUs are required for special applications.

Serviceability is a key focus for us at Facebook. We have thousands of Wedge switches deployed and only a few data center technicians, which means we need to be able to service and repair a switch quickly and effortlessly, without any tools. We made major improvements to the serviceability of the switch with Wedge 100. The top cover can now be removed without tools, providing easy access to switch internals for our data center technicians. The hot-pluggable fan trays can now be removed by pressing on a clip versus by removing a thumb screw. Additionally, the data center technicians can easily view the fan tray status via a status LED next to each fan tray. The combination of these allows for very quick debug and in field replacement if needed.

For the uplinks of Wedge 100 in Facebook's data centers, we use optics that come with a lower case temperature limit of 55 C, as opposed to the standard 70 C commercial temperature range. We put a lot of effort into the system thermal design, including the addition of one more fan tray, air baffles to separate the air paths between the PSU and the main switch board, and the front panel opening design to maximize the air flow.

The Wedge 100 chassis is designed for an industry standard 19” rack, so it can be easily adopted by the networking community. However, our data centers use [Open Racks](http://www.opencompute.org/projects/open-rack/), and we also would like to power Wedge 100 using the highly efficient, Titanium-rated Open Rack V2 Power Supply Units through the 12V bus bar at the rear of the rack. So we designed a 21” Open Rack adapter tray that helps mount Wedge 100 in such a rack. We also designed a PSU pass through module that connects the 12V bus bar to the 12V input of Wedge 100 main switch board. It is in the same form factor as the standard AC/DC PSU, and contains a 12V hot swap controller.

You can find the design specs for Wedge 100 [here](http://www.opencompute.org/wiki/Networking/SpecsAndDesigns#Facebook_Wedge_100_-_32x100G).

Wedge100 Open Rack V2 Adapter

**Software updates**

Our software stack for Wedge100 is nearly identical to the stack for Wedge 40. We run the same FBOSS switching and routing daemons, and we use the same set of tools to manage the switches in production. There's always work that needs to be done to support a new platform, but one of the guiding principles for FBOSS is to build the minimal feature set we need for our environment. This allows us to keep our code lean and easily adapt it to new environments and platforms, instead of starting from scratch or managing parallel code bases. Getting the set of protocols we support — NDP, DHCP, ARP, LLDP, ECMP, ICMP — and the features we need to operate — warm boot — working on Wedge100 was more of an iterative exercise since we had experience running Wedge 40 in production. This allowed us to hit the ground running on the Wedge100 project and use it as an opportunity to harden our FBOSS software stack.

Because we had been managing Wedge 40s in production for almost two years and had the infrastructure in place, we started testing Wedge100 in a production environment almost immediately. Running switches serving production traffic gives us the best signal for changes we need to make. As it turns out, we found our biggest challenge was on the operational side — specifically the configuration, provisioning, and managing of Wedge 40 and Wedge 100 in parallel in our network. At Facebook, the FBOSS team writes both the software that performs packet forwarding and the suite of tools we use to manage the switches operationally. Because of this, we had a positive feedback loop and, in the end, lessons learned in production informed software — and even hardware — decisions. In this way, we were able to focus on the features we actually needed and get Wedge 100 switches forwarding production traffic very quickly.

Another challenge we faced when building Wedge 100 was making our software platform more flexible. When we deployed Wedge 40, we benefited from the fact that the set of configurations we needed to support were pretty constrained, and we could easily support them. With Wedge 100, this wasn't the case. We wanted to drop a Wedge 100 at any location in our network that had a Wedge 40. Downlink speeds and cable types can vary. Uplink speeds can vary between old and new clusters. Correct configuration is especially important for 100G links as they have a much smaller optical power margin and can present more challenges in establishing link and optimizing performance, requiring much more careful configuration than 40G did. Supporting these new environments required changes from the physical layer all the way up to our monitoring stack. We expanded our support of the SFF spec to be able to operate CWDM4 optics running at various speeds by changing the power class, CDR, rate-select, FEC, pre-emphasis and others dynamically to support these different settings. We also reworked a lot of our configuration and provisioning workflows to be able to support all these possibilities.

Lastly, owning our own software stack makes fixing bugs or adding functionality to these devices much faster for Facebook. For example, we observed that 100G optics achieve higher operational temperatures, so we changed the fan control logic on our board management controller, OpenBMC, to achieve a better thermal profile when those modules are present. Other times, we encountered kernel panics on the microserver. Since we have an engineering team that manages our own kernel releases, we could leverage their expertise to debug the issues. Owning the entire stack also allows us to do other interesting things like simplifying device maintenance procedures by modifying the meaning of front panel LEDs on these switches, based on feedback that we get from the data center operations team.

The FBOSS code is all open sourced and available on [GitHub](https://github.com/facebook/fboss/tree/master/fboss). We open-sourced our software last year and have since moved to a continuous release cycle, pushing internal diffs automatically to GitHub. If you'd like to learn more about what we're working on, please check out the code.

**Hardware and software ecosystem**

The Wedge 100 switch is [now available](http://www.edge-core.com/productsList.php?cls=1&cls2=5&cls3=67) as a commercial product from [Edgecore Networks](http://www.edge-core.com/) and its channel partners worldwide. Edgecore’s Wedge 100-32X is fully compliant with the Wedge 100 OCP specification and is manufactured by Edgecore’s parent company, Accton Technology, which also manufactures the Wedge 100 for our network deployment. The Wedge 100-32X hardware switch includes a three-year warranty and is shipped with diagnostics, OpenBMC firmware, and the Open Network Install Environment universal NOS loader.

We've also seen strong interest from transceiver vendors to qualify their modules on the platform. To meet this demand, we are partnering with the [University of New Hampshire InterOperability Laboratory](https://urldefense.proofpoint.com/v2/url?u=https-3A__www.iol.unh.edu_&d=DQMFaQ&c=5VD0RTtNlTh3ycd41b3MUw&r=dPdJLtc8aRHWZ4xyIUT9oQ&m=5VPs40SENQDvU9QN4N9bGTN2HQZAJyID-ARJrfwwih4&s=n1lD4jPZnYnXUfQ38VLXa6TzJ0UjD5CJfPFZWVCrpfI&e=)(UNH-IOL) so any modules can be professionally tested at their facilities. The UNH-IOL continues to be a leader in providing neutral, broad-based testing and standards conformance services for the networking industry.

On the software side, multiple companies are building their solutions on top of the Wedge 100 platform. On the operating system layer, we have Big Switch Networks and Canonical; and on the upper parts of the stack, we have SnapRoute, FRINX and Apstra.

* [Big Switch Networks](http://bigswitch.com/) continues to drive [Open Network Linux](https://opennetlinux.org/) (ONL), the first open-sourced network operating system to support the original [Wedge 40](http://opennetlinux.org/wedge), and now Wedge 100; Facebook's FBOSS is also available right now on top of ONL on Wedge 40 and soon Wedge 100. ONL is part of the Open Compute Project and supports 30+ different OCP and non-OCP open networking switches. They also use ONL internally as the basis of their commercial products Big Monitoring Fabric and Big Cloud Fabric.
* [Canonical](http://www.canonical.com/) will bring [Ubuntu Core](https://developer.ubuntu.com/en/snappy/), their new operating system for cloud and IoT devices, to the Wedge 100 platform. Ubuntu Core can run a number of different network stacks like FBOSS or SnapRoute as snaps, and enable bare-metal provisioning for big software like OpenStack, Hadoop, and Kubernetes on the compute layer of the data center from the top-of-rack switch.
* [SnapRoute](http://www.snaproute.com/) announced the availability of [FlexSwitch software](http://l.facebook.com/l.php?d=AQGwyAG7mojvIHbpyCytYaCn3CjcBJp03Zsoii2xZZqM1wgdY0pdGsHjDio&u=http%3A%2F%2Fopensnaproute.github.io%2Fdocs%2Findex.html&h=NAQF2YXvS&s=1) on top of the Wedge 100 platform. FlexSwitch is an open source L2/L3 networking stack that can be run across multiple hardware platforms. It aims to change the economics of network operation by providing a fully customizable and programmable control plane and offering a comprehensive framework to do lifecycle automation and network analytics. It is built around the concepts of full modularity, promoting the idea of running only the set of functionality that the network requires, as opposed to the traditional one size fits all approach.
* [FRINX](https://frinx.io/) is focused on integrating [OpenDaylight](https://www.opendaylight.org/), an open source SDN platform, on Wedge 100. They are working towards the creation of a supported out-of-the-box solution for data center deployments, based entirely on open source components.
* [Apstra](https://urldefense.proofpoint.com/v2/url?u=http-3A__www.apstra.com_&d=DQMFaQ&c=5VD0RTtNlTh3ycd41b3MUw&r=2GrspeJj4mfrImGKkkk_JjROsr-z5aF0MleuKEKwviE&m=aILqFzjEN_qMjIYH89ZfYbYf66PLlhItcFsHo0zi0CY&s=8imZaMO8rVlsJdtWN_uuA3XvvfptHIs306ZH9JeB--8&e=) is operating at an upper layer with the [Apstra Operating System](https://urldefense.proofpoint.com/v2/url?u=http-3A__www.apstra.com_products_&d=DQMFaQ&c=5VD0RTtNlTh3ycd41b3MUw&r=2GrspeJj4mfrImGKkkk_JjROsr-z5aF0MleuKEKwviE&m=aILqFzjEN_qMjIYH89ZfYbYf66PLlhItcFsHo0zi0CY&s=M2EKSkVzTKLIYROJmD6ft7wyP0NClHLzjyBqKbvNhKI&e=)(AOS), a vendor-agnostic distributed operating system that can translate user intent into a continuously validated infrastructure. They are working on a [solution](http://www.apstra.com/apstra-aos-and-facebooks-wedge-100) that can manage stacks based on ONL and Snaproute as well as other network operating systems, available in early 2017.

We're thrilled to have such a complete and diverse set of technologies in the open networking hardware space, and we'll continue to work with other companies on Wedge 100. Early next year, we plan on showcasing these networking software and hardware solutions from the OCP community, including those built on Wedge 100, and we'll continue to share our experiences with disaggregated networking. Stay tuned!

*Thank you to all the teams and people who contributed to this project.*