

# OpenSwitch (OPX) Project Data

## Project Data

Data that describes a Linux Foundation Networking Project. Provide links to authoritative public content or steps to reproduce results when possible.

## Project Vitals

**Project name: OpenSwitch (OPX)**

**Project creation date: June 2016**

**Project license: Apache License, Version 2.0**

### Project release schedule

- History of at least two years or age of project
  - OPX 2.1.0 – 2017/07/31
  - OPX 2.2.0 – 2018/02/02
  - OPX 2.3.0 – 2018/03/28
  - OPX 2.3.1 – 2018/05/14
  - OPX 3.0.0 – 2018/09/07
  - OPX 3.1.0 – 2018/12/20
- Planned future release schedule
  - OPX 3.2.0 – TBD

### Statement of alignment with LFN Charter/Mission

OpenSwitch provides the LFN ecosystem with an Network Operating System (NOS) solution for white box switches, that supplements networking, instrumentation and management features with enhanced automation capabilities. Architected as a scalable, cloud-ready, agile solution, the open source OpenSwitch software implements flexible infrastructure to enable both network operators and vendors to rapidly on-board NOS applications. Build on top of unmodified Linux distribution, OpenSwitch allows other vendors to rapidly onboard new platforms.

Backed by industry leaders Dell EMC, Cavium, and Inocybe, OPX supports a rich set of L2/L3 networking features that are compatible with a wide variety of 10G, 25G, 40G and 100G hardware platforms from multiple vendors.

## Community Historical Trends

In January 2017, Dell EMC took on leadership of the OpenSwitch project and open-sourced the code as OPX. Prior to that, the project was led by HPE and the OS was known as OPS. The OPS repositories have been removed from the OpenSwitch project to avoid conflating past with present.

Over the past two years, there have been five major releases of OPX, each consisting mainly of significantly-sized code syncs from 20 repositories internal to Dell EMC out to corresponding open-source OPX repositories, as well as development that was done exclusively on OPX repositories without internal code syncs. For the synced repos, Dell EMC is actively working on making the code delivery more frequent and smaller-sized.

In May 2018, the repos that make up the OPX code transitioned from being hosted on Gerrit at [git.openswitch.net](http://git.openswitch.net) to being hosted on Github at [github.com/open-switch](https://github.com/open-switch).

The below figures for historical trends are close estimates:

- Release 2.1.0 (July 2017)
  - Contribution statistics
    - Number of commits: 1461 (initial contribution)
    - Number of commits per-organization: Various across Dell, Cavium, Mellanox
    - Number of non-trivial commits: 20 (Assuming all synced repos received non-trivial commits due to large sync from Dell internal repos)
      - Merged by uploader: 0
      - Merged by committer from same organization as uploader: 20
      - Merged without substantial code review: 0
  - Contributor Statistics
    - Number of contributors: 11
    - Per-organization
      - Dell EMC: 9
      - Mellanox: 1
      - Cavium: 1
- Release 2.2.0 (February 2018)
  - Contribution statistics
    - Number of commits: 244
    - Number of commits per-organization:
      - 240 Dell
      - 4 Cavium
      - 1 Mellanox
    - Number of non-trivial commits: 9 (20 synced repos, but many without significant changes)
      - Merged by uploader: 0

- Merged by committer from same organization as uploader: 9
  - Merged without substantial code review: 0
- Contributor Statistics
  - Number of contributors: 11
  - Per-organization
    - Dell EMC: 9
    - Cavium: 1
    - Mellanox: 1
- Release 2.3.0 (March 2018)
  - Contribution statistics
    - Number of commits: 46
    - Number of commits per-organization: 46 Dell
    - Number of non-trivial commits: 1
      - Merged by uploader: 0
      - Merged by committer from same organization as uploader: 1
      - Merged without substantial code review: 0
  - Contributor Statistics
    - Number of contributors: 9
    - Per-organization
      - Dell EMC: 8
      - Cavium: 1
- Release 3.0.0 (September 2018)
  - Number of commits: 321
  - Number of commits per-organization:
    - 9 Cavium
    - 312 Dell
  - Contribution statistics
    - Number of non-trivial commits: 5
      - Merged by uploader: 5
      - Merged by committer from same organization as uploader: 0
      - Merged without substantial code review: 0
  - Contributor Statistics
    - Number of contributors: 10
    - Per-organization
      - Dell EMC: 8
      - Cavium: 1
      - Inocybe: 1
- Release 3.1.0 (December 2018)
  - Number of commits: 125
  - Number of commits per-organization:
    - 1 Inocybe
    - 124 Dell
  - Contribution statistics
    - Number of non-trivial commits: 4
      - Merged by uploader: 4
      - Merged by committer from same organization as uploader: 0
      - Merged without substantial code review: 0
  - Contributor Statistics
    - Number of contributors: 9
    - Per-organization
      - Dell EMC: 8
      - Inocybe: 1

## Community Current Status

The project is currently in process of finalizing a draft plan for the increased diversity. The draft plan has been reviewed by the technical committee. The next step is the review and approval by the OPX board. Here is a link to the draft plan:

<https://docs.google.com/document/d/1TEsz8ZK7osun2kSF8F1Zoun8rw0fID0vIKZ1JiCqE0g/edit#heading=h.zfmhk6hhcejr>

Snapshot of the candidate project's community.

### Committer statistics

- Number of committers: 10
- Number of committers per-organization
  - Dell EMC: 8 (8 people actively commit code publicly on Github, but the code is developed internally by about 40 Dell EMC employees)
  - Inocybe: 1
  - Cavium: 1
- Number of active committers: 9
- Number of active committers per-organization
  - Dell EMC: 8
  - Inocybe: 1

### Contributor approval process

- Contributor eligibility

- None; Based on discussions on common communication channels and TSC call discussions, contributors are encouraged to open pull-requests.
- Process to become a contributor
  - None, but contribution guidelines are here: <https://github.com/open-switch/opx-docs/wiki/Contribute-to-OpenSwitch-OPX>
- Process to remove a contributor \* None

## Committer approval process

- Committer eligibility
  - Determined by organization admins. Organization admins are a group of engineers who are aware of overall structure and architecture of OPX. The admins can make determination whether committers have enough project-level knowledge, experience and awareness to contribute. The admin group is comprised of project tools and build admin, one or more architects, and one or more experienced committers.
- Process to become a committer
  - No formal process. Committers currently consist of mostly Dell EMC employees, plus an Inocybe employee for a couple of repos in the project.

## Process to remove a committer

- Removal requests are considered on individual basis, with written request by developer or maintainers to TSC.

## Project governance structure

- Governance charter is found here: <https://www.openswitch.net/about/project-charter/>
- OpenSwitch's mission is to facilitate a community-led, industry-supported open source Network Operating System (NOS) framework, including code and architecture, to accelerate, promote and advance a common multi-vendor supported diverse and robust Open Networking platform, enabling the transition to disaggregated switching modern networks.
- Specifically, the mission includes but is not limited to the following:
  1. Grow a community of partners who share OpenSwitch vision and mission
  2. Create an ecosystem of contributors and users around the OpenSwitch NOS
  3. Produce an open-sourced innovative network operating system software at deployable grade, striving for quality and consistency
  4. Create an open source, open participation technical community to benefit the ecosystem of OpenSwitch solution providers and users
  5. Promote participation of leading members of the ecosystem, including developers, service and solution providers and end users
  6. Establish a neutral home for community infrastructure, meetings, events and collaborative discussions and providing structure around the business and technical governance of the OpenSwitch project
  7. In pursuit of its mission and goals, the OpenSwitch is committed to value and enable innovation, operate with transparency and respect all the community members in all interactions

## User community

- Communities contributing currently and in the past:
  - HPE: Original code for OpenSwitch (aka OPS)
  - Dell EMC: The base code to replace HPE
  - SnapRoute: The routing stack to run on top of the Dell contributed base software.
  - MetaSwitch: The routing stack to work with OPX.
  - Cavium: Platform and ASIC code contributions
  - Mellanox: ASIC code contributions
  - Inocybe: DHCP Relay, monitoring/telemetry software, and other small contributions.
  - Broadcom: Monitoring and telemetry software/drivers.

## Project Functionality

### Summary of candidate project functionality

OPX is an open source networking operating system. It provides an API for applications to work with platform and networking elements of a network switch, and it leverages the OCP's Switch Abstraction Interface (SAI) to provide portability between various ASICs. Lastly, OPX runs on an unmodified Linux kernel in order to provide a familiar environment to users and an easy path to onboard new platforms and applications.

### Summary of candidate project technology components and purposes –

The [OPX architecture](#) consists of three major components:

- [Platform Adaptation Services](#) (PAS) – Provides system integrators with a common way to define platform elements like buses, fans, power supplies, LEDs, supported media, etc.
- [Network Adaptation Services](#) (NAS) – Working alongside SAI (and a vendor's implementation of SAI), provides and manages the high-level Network Processing Unit (NPU) abstraction and adaptation. NAS abstracts and aggregates the core functionality required for networking access at Layer-1, Layer-2, and Layer-3, ACL, QoS, and network monitoring. NAS is also responsible for providing packet I/O services using the Linux kernel IP stack and Netlink interface.
- [Control Plane Services](#) (CPS) – Provides an object-centric framework that mediates interactions between applications and the NAS and PAS services. Client applications execute create/set/get/delete operations on individual objects or lists of objects, and services handle those operations. CPS also supports a publisher/subscriber model, where clients can register for events generated when objects are created, modified, or deleted. The publisher/subscriber approach and object-centric operations allow for completely independent operation of client and server applications.

Summary of where candidate project complements functionality already provided by project(s) within LFN

Since OPX runs on an unmodified Linux kernel and enables a lot of networking functionality with the Netlink API, many projects that work with Linux can similarly work with OPX. One such example is FRRouting, which provides a Layer-3 networking control plane – as FRRouting installs routes into the Linux kernel, those routes are also populated in the switch's networking ASIC (via the NAS Linux adapter) so that the switch can forward traffic accordingly.

Summary of where candidate project overlaps functionality already provided by project(s) within LFN

OPX is a network operating system, so its functionality would potentially overlap with other projects within LFN that are also network operating systems that run directly on an embedded system, such as Stratum.

## Project Tooling

Details about the tooling used by the candidate project.

### Bug tracker:

- <https://github.com/open-switch/opx-docs/wiki/Report-bugs>
- Issues list: (Login to github required) <https://github.com/issues?q=is%3Aopen+is%3Aissue+user%3Aopen-switch+sort%3Aupdated-desc>
- OPX uses the Github issue tracker for all of its bugs.

### Chat tooling

- Project RocketChat
  - <https://chat.openswitch.net>
  - OPX community quite active in chat channels.
  - Links to chat tooling used by the project.
  - Overview of chat tooling used by the candidate project.
  - Users have the option of creating private channels with each other if necessary.

### Code repositories

- <https://github.com/open-switch>
  - 59 public repositories
  - Provides a list of all OPX repositories which are mapped to architecture components
    - <https://github.com/open-switch/opx-docs/wiki/Repositories>

### Code review

- \* OPX uses the Github Pull Request interface for code review
- \* Overview of code review norms, practices, conventions, rules.
- \* To what extent are external/private code review systems used? None

### Continuous Integration tooling

- CodeFactor
- BuildKite, whose pipeline consists of:
  - Setup
    - Docker Hub
  - Build
    - dbp
  - Smoketest
    - Docker Hub
    - AWS
  - Github DCO

### Documentation

- <https://github.com/open-switch/opx-docs/wiki>

### Mailing lists

- <https://lists.openswitch.net/>
- dev@ primary mailing list 556 subscribers.
- tsc@ Public TSC list
- Private Governing board list.
- Active chat forum <https://chat.openswitch.net> with 285 subscribers
- To what extent are external/private mailing lists used? None

### Meeting calendars

- Weekly public TSC meetings.
  - <https://github.com/open-switch/opx-docs/wiki/Technical-Steering-Committee-Meeting-Details> \* Weekly meeting minutes (PPT files) archived in tsc mailing list. <https://lists.openswitch.net/g/tsc/topics>
  - To what extent are meetings public, and clearly publicly documented?
- Meeting minutes
  - Weekly meeting presentations are archived in tsc mailing list. <https://lists.openswitch.net/g/tsc/topics>
  - TSC call recordings are available if requested.

- To what extent are public minutes for meetings taken and shared? Generally the meeting topics are documented in the TSC powerpoint files. Open action items are tracked in the last slide(s) of the PPT files and are followed up at the next week's meeting.

## Integrations

- Inocybe, who are contributing a DHCP relay agent and OpenDaylight REST interface
- InMon, who adapted their hsflowd SFlow agent to work with our OPX CPS interface.
- Broadcom and Cavium to integrate their telemetry applications, although unlike the above two projects these were closed-source.
- Integration with FRR Routing protocols (we have not contributed to each others' code); we rely on our mutual integration with the Linux kernel in order to work.

## Vocabulary Reference

Explanations of domain-specific vocabulary.

- Active
  - In this context, typically related to the activity level of a project or person.
  - As a person: "Foo Committer on Bar Project has not sent any patches or done any code review for Bar in the last 12 months. Bar's Project Lead reached out to Foo Committer to discuss transitioning to an Emeritus Committer."
  - As a project: "Bar Project has not had any non-trivial code changes merged in the last 12 months. The LFN TAC reached out to Bar Project to discuss transitioning to the LFN Archived lifecycle state."
  - The LFN norm for "active" is about 12 months.
- Committer
  - Person with permission to cause commits to be merged into a project's source control repositories.
- Contributor
  - Person who has contributed to a project. "Contributions" are broadly defined. Examples include things like code, documentation, and bug tracker changes.
- Diverse
  - In this context, typically related to the number of different organizations involved in a project.
- Downstream
  - In this context, typically means the products based on a project. Community collaborates on upstream project, which is downstreamed by a company into a product.
  - Alternatively, could relate to a relationship between two "upstream" open source projects (not by-company products) where one consumes (is downstream of) the other.
  - As a verb: "to copy something from the open source project to a product based on it".
  - As a dependency relationship: "Linux is a downstream of C".
- Upstream
  - In this context, typically means the main open source project a community collaborates on. The code, tooling and people that comprise a project.
  - As a verb: "to add something to the main open source project".
  - As a dependency relationship: "C is an upstream of Linux".