**OSS Infrastructure Projects**

Open source software has been a key behind enterprise IT for years, so it’s no surprise that it’s helping to drive the infrastructure part of the equation forward just as much as application development.

Some projects are much more influential than others, and here are five that are doing the most to help enterprise infrastructure keep pace with the demands of an ever-more sophisticated operating environment.

**OpenStack**

OpenStack is notable in part for being an open-source competitor to the most important proprietary virtualization software on the market—VMware’s VSphere. For the basic task of virtualizing servers into a flexible pool of computing resources, the difference appears to be ease of use—it’s simpler to use VMware when there isn’t a lot of in-house virtualization or private-cloud expertise.

OpenStack is important in networking for one main reason: the telecom sector and [network function virtualization (NFV](https://www.networkworld.com/article/3253118/what-is-nfv-and-what-are-its-benefits.html)), which uses enterprise virtualization technology to perform networking tasks previously allocated to dedicated hardware tied to proprietary software. Telecom providers love this idea because it lets them replace expensive, proprietary products with general-purpose switches and servers. Also, software used for NFV, like OpenStack, lets them dynamically provision workloads and deploy new capabilities more flexibly.

**Ansible**

Originally developed by Red Hat, Ansible is an open-source IT-automation and configuration-management tool that offers an alternative to configuring hardware manually. The idea is that the IT team writes a script that describes the network and what it’s supposed to do and then Ansible automatically configures the relevant devices. It doesn’t use agent software, instead pushing “Ansible modules” directly to the devices via Secure Shell (SSH) for easy deployment.

Red Hat offers an array of paid add-ons for Ansible, as well, including improved security, role-based access control, and job scheduling. Ansible offers a method of network configuration that lets IT workers set configurations once on a single controller and automatically pushes them to devices on their networks. Software can also be pushed to all devices on the network or just a relevant subset by editing to the main playbook. Changes can be tracked and identified in Git or some other versioning system.

**Akraino**

Akraino launched in 2019 and is the product of the Linux Foundation’s LF Edge program, which is aimed at creating open frameworks for [edge-computing](https://www.networkworld.com/article/3224893/what-is-edge-computing-and-how-it-s-changing-the-network.html) deployments. Akraino is a collection of configuration blueprints designed to offer a freely available, off-the-shelf recipe for network and hardware configurations for specific use cases.

Akraino currently includes 11 blueprint families grouped by general use area and 27 specific blueprints. One example is StarlingX Far Edge Distributed Cloud, which specifies a hardware setup, containerization providers, and an orchestration framework to enable applications to run in high-density locations like airports, sports stadiums, and malls. Other blueprints focus on AR/VR infrastructure, telecom radio deployment, and various types of [IoT](https://www.networkworld.com/article/3207535/what-is-iot-the-internet-of-things-explained.html).

The idea behind StarlingX is to offer vendors and sophisticated end-users a way to streamline the configuration of the common elements of edge deployments. A company with a new application for a particular vertical market—say, providing real-time monitoring to connected factories—can focus on that without having to design the underlying computing infrastructure.

**Kubernetes**

Kubernetes is a containerization platform for all kinds of enterprise workloads, originally the product of Google engineers, but released as open source in 2014. It’s since become an industry standard, accounting for 71% of containerization use in the enterprise, according to a study from 451 Research.

Enterprises like containerization in general, and Kubernetes in particular, because it’s an effective simplification vs. monolithic models of service deployment. Instead of a single application offering a range of services and requiring a specialized infrastructure, Kubernetes breaks each process used by the application into its own container and virtualizes it.

What that means is that the containerized workloads can run wherever—on premises, public cloud, private cloud, or various combinations thereof  simultaneously—and work just as they would if they were bundled in a single application running on dedicated hardware. Consequently, developers can create a file that outlines how the services are supposed to work, and Kubernetes automates everything from provisioning to failover to updates.

**Linux itself**

Any listing of open-source projects that are important to the enterprise networking must include the Linux kernel. Linux fundamentally underpins huge amounts of modern enterprise networks, including all of the other projects listed here. By extension that means it’s also the basic operating system behind 90% of the public cloud, according to a 2019 survey by Red Hat.

Even by itself, the operating system includes robust networking features that make it easy to deploy on white-box hardware. As the tasks of deploying and managing networks become more and more software-based, Linux skills are increasingly critical to just about every network IT professional out there.

**OpenDev Project**

OpenDev is an evolution of the OpenStack Infrastructure project. The goal is to make OpenStack’s proven software development tools available for projects outside of OpenStack. It is believed that Free Software needs Free tools and OpenDev provides one such set that has been proven to work at large and small scales of development. The OpenDev team is an open meritocracy that welcomes new members.

**Scope**

OpenDev now covers many of the original OpenStack Infrastructure systems, but not all. The goal is to run any service that has generic applicability for open and collaborative software development in OpenDev. OpenStack and other project specific tooling would be managed by those projects outside of OpenDev.

In particular OpenDev covers the operations and development of code management systems and collaboration tools. Git repository management, code review, continuous integration, mailing lists, and more are all within the scope of OpenDev.

**Priority Efforts**

The OpenDev project designates a small number of efforts underway at any time as priority efforts. These are areas where the project has decided to focus resources to achieve major initiatives. These help reviewers prioritize their review workload and help to ensure the project accomplishes important tasks. Priority efforts are a great way to get involved in the project as they will generally provide the most interaction with experienced contributors.

Priority efforts are documented in the infra-specs repo. Each priority effort has one entry in infra-specs, though that may link to multiple smaller specifications for individual units of work if the effort is sufficiently large. Each priority effort also has a single person designated as the driver of that effort. That person is responsible for ensuring that anything blocking progress of the effort is discussed at team meetings and may be a good point of contact for someone who wants to get involved.

Changes not related to priority efforts will be reviewed by the core review team as time permits. This may mean that they spend considerable time in review, but they will be reviewed eventually.

**Governance**

The OpenDev project is governed by two entities. The first is the Service Coordinator. They coordinate work of contributors and act as a tie breaker when clear consensus isn’t found. The Service Coordinator is responsible for managing spec reviews and core team membership.

The Service Coordinator is elected every 6 months. The nominee pool and electorate are individuals that have contributed changes to OpenDev git repositories in the last 12 months.

The second is an Advisory Board made up of representatives from OpenDev’s user base of projects and organizations that contribute compute resources. This Advisory Board provides a formal location for those key stakeholders to express their needs to the OpenDev project. This creates a clear contact point for feedback on priorities and direction. Their input will help ensure that the OpenDev project is a good steward of the resources provided to it and that user needs are being addressed.

**Teams**

The OpenDev project is open, meaning anyone may join and begin contributing with no formal process. As an individual’s contributions and involvement grow, there are more formal roles. These roles are designed to empower groups of people to get work done in their area of expertise and interest, as well as supply a strong sense of direction for the OpenDev project as a whole. Everyone participating in the project is encouraged to expand their own knowledge while helping to support and mentor others as they progress.

Core Teams

The OpenDev project is composed of a large number of subprojects. Every source code repository has its own core team which is responsible for maintenance of that subproject, with some groups of repositories sharing a core team. These core teams are empowered to approve changes that reflect the currently understood project direction. Changes in project direction or major new initiatives must be approved by the OpenDev council.

Any existing core team member may nominate someone for addition to that core team by private communication with the OpenDev Service Coordinator. The Service Coordinator will consider the opinions of the existing core team members and the review history of the person in question, but final determination of core team membership (additions and removals) rests with the Service Coordinator. This process is private to enable honest evaluations in a safe environment.

OpenDev Core Team

Individuals who show an interest in a wide range of areas of the OpenDev project may be asked to join the infra-core team. To provide a baseline level of support to all of our subprojects and to ensure that important efforts may move forward, this team has approval rights in all OpenDev repositories. Members of this team may not be experts in all areas, but know their limits, and will not exceed those limits when reviewing changes outside of their area of expertise.

They are expected to have a wide general knowledge of what is going on in the OpenDev project and to help guide overall project direction. To that end, they are able to veto specs proposed to the OpenDev council.

OpenDev Council

The OpenDev council is the technical design body for the Opendev project. While individuals and groups are empowered to execute the designs from the council, major technical designs are agreed upon as a group to ensure that their large set of projects are all working together to the same end. The council need not delve too deeply into technical detail – just enough so that development efforts may happen in parallel and work toward a common goal.

All members of any OpenDev project core team have a seat on the Council. The Council is responsible for approving changes in project direction, major new initiatives, setting priority efforts, and addition or removal of projects.

OpenDev Root Team

Core membership enables one to approve changes within our code repositories. Because the OpenDev team operates production servers there is another sub-group of the OpenDev team that has root access to all servers. Root membership is handled in the same way as core membership. Root members must also be infra-core members, but infra-core members may not necessarily be root members. This is because primary system administration is performed through code review, so anyone able to log into a machine to execute commands must be able to approve those same commands in configuration management; otherwise it would be easier for a person to bypass configuration management than use it in the intended fashion.

Root access is generally only necessary to launch new servers, perform low-level maintenance, manage DNS, or fix problems. In general it is not needed for day-to-day system administration and configuration which is done through automated config management tools (where anyone may propose changes). Therefore it is generally reserved for people who are well versed in OpenDev operations and can commit to spending a significant amount of time troubleshooting on servers.

Some individuals may need root access to individual servers; in these cases the infra-core group may grant root access on a limited basis.

**Review Criteria**

The primary purpose of change review is to catch substantial errors before they are merged. In order to keep this process useful and avoid frustration for both authors and reviewers, please do not leave negative reviews for insubstantial faults or potential improvements. The purpose is not to make someone else’s code match your vision of perfection, but to enable all of us to work together on a project.

Please use discretion when deciding what is important enough for someone to spend the time to rework and for you to spend the time re-reviewing. Sometimes minor things are important, such as consistent use of hyphens versus underscores in a configuration language. Sometimes they are not, such as whitespace in documentation.

**Open Infrastructure Technical Overview**

The OpenDev system administration team strives to run the services behind the OpenDev Collaboratory as an open source project; termed as open infrastructure.

Our infrastructure is code and contributions to it are handled just like the rest of OpenDev. This means that anyone can contribute to the installation and long-running maintenance of systems without shell access, and anyone who is interested can provide feedback and collaborate on code reviews. There are no permissions or special privileges required to contribute to the OpenDev infrastructure project.

**Operating environment**

The OpenDev production systems run in resources (compute, network, storage) provided by donations from companies who support the project. Our standard production system is based on the latest Ubuntu LTS release.

Production systems are deployed by Ansible. Most production applications run from containers; some are custom built and others we use unmodified from upstream sources.

Zuul handles the testing and deployment of all changes. Current trends would refer to this as a gitops model – all production changes are ultimately driven by a change proposed to the code-review system.