bluebox® cloud

BATTLE TESTED OPENSTACK

Our Hardware; Your Deployment

Trust in the expertise of our infrastructure experts with **Blue Box Cloud**. With 24/7 Support from our award-winning team, your experience will be top-notch.

800.613.4305 | +1.206.607.0660 | sales@blueboxcloud.com

blueboxcloud.com

CONTENTS

Getting Started with OpenStack

The Most Popular Open-Source IaaS

By Sriram Subramanian

INTRODUCTION

HISTORY AND VISION

OpenStack™ is an open-source Infrastructure-as-a-Service (IaaS) platform. OpenStack was founded by joint efforts from Rackspace and NASA in 2010. It is used to manage large pools of compute, storage, and networking resources in a data center, all managed through a centralized dashboard.

OpenStack is one of the fastest growing open-source communities in the world, with more than 18,000 individual contributors and 430 participating companies. It is governed by the OpenStack Foundation, which promotes the development, distribution, and adoption of OpenStack. Individual membership to the Foundation is free, and corporate memberships are paid for at two levels: Platinum and Gold. The Foundation is guided by a board of directors, which is composed of individuals and representatives of participating corporations.

According to the OpenStack Pulse 2014 report from 451 Research, OpenStack-based market revenue is expected to reach $1.3B by 2016 and exceed $3B by 2018. OpenStack is quickly becoming the platform of choice for private cloud deployments.

OPENSTACK SOFTWARE

The OpenStack platform is a collection of several projects that work together to provide capabilities to manage compute, storage, and networking resources in your data centers. It also provides an extensive set of APIs and a centralized dashboard to manage cloud resources. OpenStack is designed to run on commodity hardware. For customers who want to use specific enterprise-grade hardware, special drivers are available.

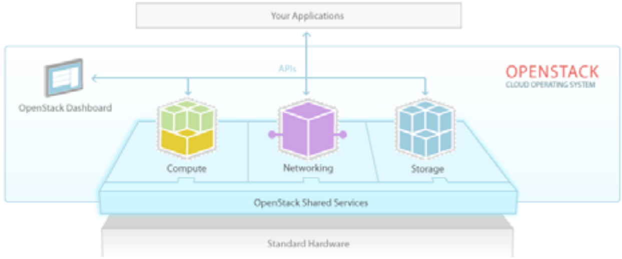


Figure 1: *OpenStack Software*

OpenStack follows a biannual development and release cycle, offering one release in late spring and a second in late fall. Each release is codenamed alphabetically, with the most recent release named Juno (begins with "J" indicating the 10th release). Currently, the Kilo release is in active development and is scheduled for release in April 2015. It will be followed by the Liberty release in November 2015.

Immediately following each release, the OpenStack community convenes to review the newly released cloud software and to begin planning the next release (Design Summit). The next two OpenStack Summits will be held in Vancouver, BC, Canada and Tokyo, Japan.

ADOPTION TRENDS

OpenStack adopters repeatedly identify ability to innovate, to automate, and to utilize open technologies as their top three driving factors for choosing OpenStack. The IT industry shows the widest uptake, and adoption is increasing in Academic/Research, Telecommunications, Finance, Media, and more. OpenStack is being widely adopted across many global regions, with an increasing number of OpenStack deployments moving from test/staging environments into production.

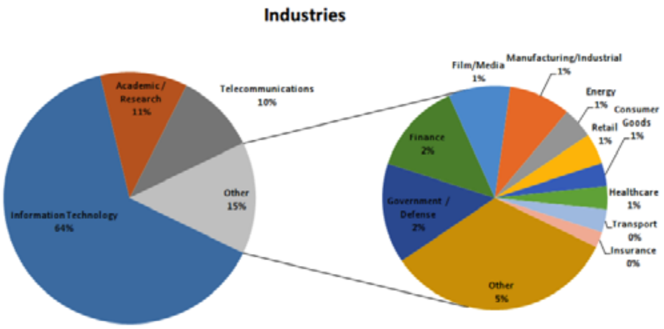


Figure 2: *OpenStack Adoption by Industry Verticals (OpenStack User Survey, Nov 2014)*

Organizations run different types of workloads on OpenStack clouds, led by web services and QA/Testing workloads. OpenStack users report that they are deploying an increasing number of Enterprise Applications.

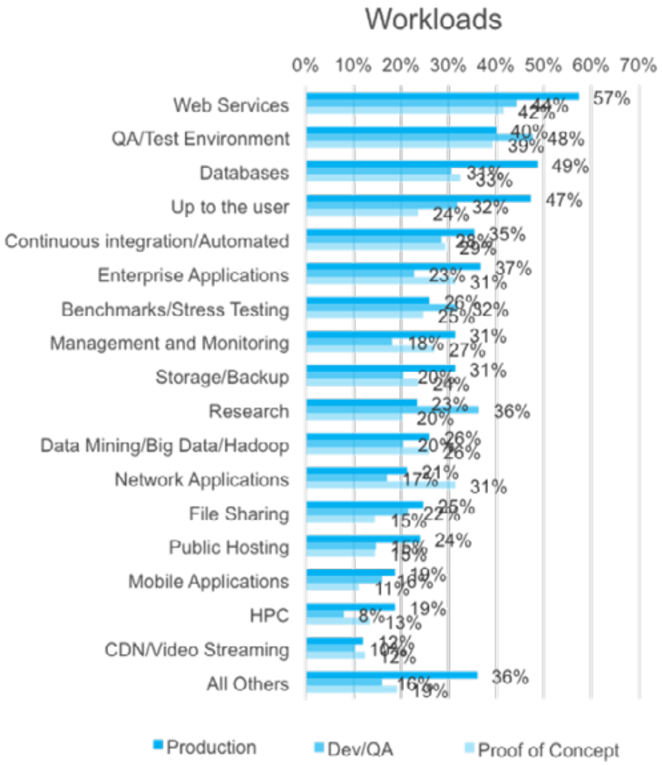


Figure 3: *Common Workloads deployed on OpenStack Clouds (OpenStack User Survey, Nov 2014)*

COMPONENT OVERVIEW

INTEGRATED PROJECTS

Among the OpenStack repertoire are sets of services that are integrated together as part of the regular, biannual release. All OpenStack software described here is available under the Apache License.

COMPUTE

OpenStack Compute service (*Nova*) provides on-demand provisioning and management of virtual machines. It supports multiple hypervisors including KVM, XenServer, VMware ESXi, and Microsoft Hyper-V. It also supports Linux containers such as LXC.

STORAGE

OpenStack Storage service provides support for both object storage and block storage. Object Storage supports scale-out, distributed non-structured data; block storage provides virtual storage for virtual instances.

Object: OpenStack Object Storage (*Swift)* provides for cost-effective, API-accessible, distributed, redundant, scale-out storage ideal for backup, archiving, and data retention. It is an excellent platform for storing images, videos, virtual machine images, and archives.

Block: OpenStack Block Storage (*Cinder)* provides for creating, attaching, and detaching block devices to virtual serves. It is fully integrated with OpenStack Compute, can be managed from the Dashboard, and it supports integration with enterprise storage platforms such as Nimble, Solidfire, or EMC through specific drivers. It also provides support for volume snapshots.

NETWORKING

OpenStack Networking service (*Neutron*) provides a pluggable, API-driven platform for managing networks and IP addresses. It supports multiple network models (Flat, VLAN, VXLAN), static IPs, and DHCP. It can also leverage advanced networking capabilities by taking advantage of SDN platforms such as OpenFlow.

SHARED SERVICES

Apart from the three basic building blocks of compute, storage, and networking, OpenStack has several services that integrate these components with each other as well as with external systems to provide a unified experience for the users.

IDENTITY

OpenStack Identity service (*Keystone*) provides authentication and authorization capabilities. It provides a central directory of users, mapping them to the services they can access. It can also integrate with existing backend directory services such as LDAP.

IMAGE

OpenStack Image service (*Glance*) provides API-accessible discovery, registration, and delivery services for disk and server images. The image service can store images in a variety of backends, including OpenStack's object store Swift. The OpenStack Image Service accommodates multiple image formats, including Raw, VHD, VMDK, and VDI.

TELEMETRY

OpenStack Telemetry service (*Ceilometer*) aggregates usage and performance data and enables alarm capabilities across OpenStack services.

ORCHESTRATION

OpenStack Orchestration service (*Heat*) provides automated, template-based infrastructure deployment capabilities. Along with the Telemetry service, it can provide auto-scaling capabilities to select features.

DATABASE

OpenStack Database service (*Trove*) provides Database-as-a-Service (DBaaS) capability on OpenStack cloud infrastructure.

DASHBOARD

OpenStack Dashboard (*Horizon*) provides centralized graphical user interface to access, provision, and manage cloud resources. Third-party services such as billing and monitoring can be easily integrated with OpenStack Dashboard. OpenStack services can also be accessed through APIs.

DATA PROCESSING

OpenStack Data Processing service (*Sahara*) provides a scalable data processing stack and associated management interfaces.

LATEST PROJECTS

OpenStack has more services that can optionally integrate with other OpenStack services. Some of these projects include Application Catalog (*Murano*), Bare Metal Service (*Ironic*), Containers Service (*Magnum*), Deployment service (*TripleO*), DNS-as-a-Service (*Designate*), Key Management (*Barbican*), Message Service (*Zaqar*), Shared File System (*Manila*), and more.

LOGICAL OVERVIEW

For a logical view of OpenStack architecture, please check http://docs.openstack.org/admin-guide-cloud/content/logical-architecture.html.

DELIVERY MECHANISMS

OpenStack software is currently delivered to customers in two ways:

1. **As a collection of packages**: This is commonly referred to as a 'distribution.' A special category of distributions is bundled together with specific hardware and is called 'appliances.'
2. **As a service**: Currently, delivering via a distribution is the default delivery mechanism for OpenStack. Apart from non-commercial distributions such as Debian and Fedora, many vendors offer proprietary distributions, which typically add value-added services.

DISTRIBUTIONS

An OpenStack distribution bundles OpenStack services as packages to download; these packages are available for different platforms (Ubuntu, CentOS, Debian, etc.). OpenStack software is updated on a continuous basis and is available for download from what is commonly referred to as a trunk. These continuous versions are not stable; only the biannual releases are stable. One can always download the most recent stable version instead of building packages continuously.

THIRD-PARTY DISTRIBUTIONS

Participating vendors also offer their custom distributions of OpenStack. The advantages provided by the vendor could include value-added services, proprietary services, or integrations with other vendor offerings. Here are some of the popular third-party distributions:

1. **Mirantis OpenStack**: Mirantis OpenStack is an OpenStack distribution from Mirantis, one of the largest OpenStack System Integrators (SI). They are commonly referred to as a pure-play OpenStack vendor, which exemplifies their philosophy of being vendor-agnostic and featuring only open-source code.
2. **Red Hat Enterprise Linux OpenStack Platform**: Red Hat Enterprise Linux OpenStack Platform is a security-hardened, enterprise-class cloud platform from Red Hat that leverages the advantages of Red Hat Enterprise Linux and Red Hat Enterprise Virtualization platforms.
3. **HP Helion OpenStack**: HP Helion OpenStack is a scalable, secure OpenStack distribution from HP, with more value-added services that integrate well with other HP offerings. HP also gives the option of a free-to-download HP Helion OpenStack Community Edition that comes with support for small-scale private clouds.
4. **IBM Cloud Manager**: IBM Cloud Manager with OpenStack is a cloud management software from IBM, based on OpenStack.
5. **Others**: You can find about more OpenStack distributions here, including Nebula One Cloud Controller, Piston OpenStack, Rackspace Private Cloud Software, Ubuntu OpenStack, and more.

AS A SERVICE

A handful of vendors offer OpenStack cloud through an 'as-a-service' delivery mechanism, in which OpenStack cloud is available on-demand and in a hosted environment. Customers get single-tenant OpenStack cloud with all its advantages, including API access, without the overhead of deployment and operations.

For customers who want OpenStack in a dedicated, isolated environment, but without the overhead of operations and additional infrastructure, OpenStack cloud-as-a-service is an attractive option.

1. **Blue Box**: Blue Box, a venture-funded (Series A, Series B) startup based in Seattle, is one of the earliest providers to offer hosted OpenStack Private-cloud-as-a-Service—PcaaS. Blue Box is rated as a "Strong Performer" in The Forrester Wave: Hosted Private Cloud Solutions, Q4 2014. Blue Box leads the OpenStack Seattle User Group meetups and actively contributes to the OpenStack community, particularly around the OpenStack Operators work group. They recently started offering services to manage your OpenStack private cloud in your own data center, fully managed by Blue Box experts.

2. **Metacloud (acquired by Cisco):** Metacloud, now part of Cisco, is another vendor offering OpenStack cloud-as-a-service. They started with Cloud-in-a-box or Private Cloud Appliance based in OpenStack, and started offering hosted Private Cloud. Their custom OpenStack distribution is also known for its superior User Experience (UX).

ENVIRONMENT

PUBLIC CLOUD

The easiest way to learn more about how OpenStack cloud operates is to try out one of the OpenStack-based public cloud service providers. Here are some of the public cloud service providers. You can learn more about them here.

RACKSPACE PUBLIC CLOUD

Rackspace Public Cloud is the largest OpenStack-based public cloud service provider. Rackspace, one of the co-founders of OpenStack, is also a strong contributor to OpenStack projects and documentation.

HP HELION PUBLIC CLOUD

HP Helion Public Cloud, HP's public cloud service offering based on OpenStack, offers compute and storage resources.

LOCAL

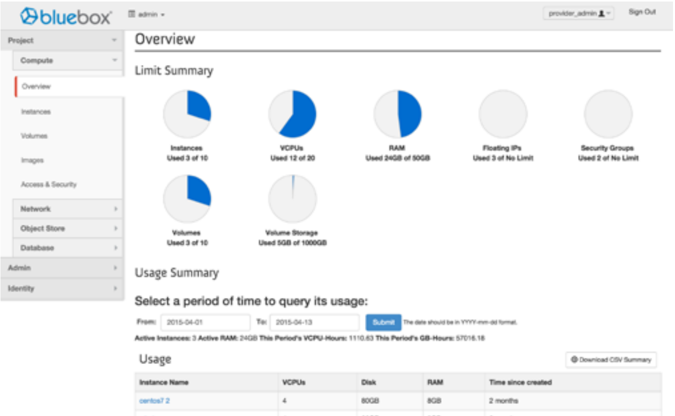
If you want to get started on a local environment, you can get started easily with DevStack—an easy to setup, all-in-one configuration.

DEVSTACK

DevStack is an easy way to install OpenStack test cloud with minimal configurations. It is currently available on Ubuntu, Fedora, and CentOS/RHEL platforms. It is intended to be used for prototyping environments. DevStack offers All-in-One-Single VM, All-in-One-Single Machine, and Multi-Node configurations.

HOSTED

Hosted OpenStack clouds are an easy option for getting started with OpenStack. For instance, when you sign up with a provider like Blue Box, you can have your private cloud started in a few hours. Here is a snapshot of the Blue Box account dashboard showing one active resource:



Select a period of time to query its usage:

From: 2015-04-01 To: 2015-04-13 The date should be in YYYY-mm-dd format.

Active Instances: 3 **Active RAM:** 24GB **This Period's VCPU-Hours:** 1110.63 **This Period's GB-Hours:** 57016.18

Instance Name	VCPUs	Disk	RAM	Time since created
centos7 2	4	80GB	8GB	2 months

Figure 4: *Blue Box Account Dashboard*

KEY TERMS

ACCESS KEY/SECRET KEY

Used in combination to access and communicate with a compute instance. Secret key is used to digitally sign each request.

AVAILABILITY ZONE

An isolated set of hypervisors within a cloud used to provide fault tolerance to virtual machines.

BLOCK STORAGE

Type of storage (intended to offer persistent storage mounts for virtual machines) that supports volumes, volume snapshots, and volume types.

CONTAINER (OBJECT)

Organizes and stores objects in object storage.

CONTAINER (LXC)

Linux Container is a virtualization environment at the operating system level for running multiple isolated Linux systems on a single Linux host. OpenStack Container Service (Magnum) provides for management of application containers.

FIXED/STATIC IP

An IP address associated with the same VM instance each time it reboots. This is used for VM management, and is generally not accessible to users.

FLAVOR

A set of parameters of the virtual machine images; these parameters include CPU, memory size, storage size, etc.

FLOATING IP

An IP address associated with a VM instance so that the instance has the same public IP address every time it reboots.

IMAGE

A collection of files for a specific operating system that one needs to create or rebuild a server. OpenStack supports multiple image formats (AMI, VMDK, VHD, QEMU).

NETWORKS

A virtual network is a layer-2 network that provides connection between entities.

OBJECT

An object is any kind of data that is stored in object storage, and can be of any format — file, music, video, or binary.

OBJECT STORAGE

A type of storage that supports eventually consistent, redundant, non-structured data.

PROJECT/TENANT

A logical grouping of users (commonly referred to as tenant).

QUOTAS

Limits on compute and storage resources, set on a per-project basis.

ROLE

Includes a set of rights and privileges. A user assuming that role inherits those rights and privileges.

SECURITY GROUP

A set of filtering rules applied to a compute instance.

SERVER

A virtual machine instance.

SERVICE

An OpenStack service that provides endpoints through which users can access and control resources.

SNAPSHOT

A point-in-time copy of a storage volume or an image.

USER

A part of a *project/tenant* who consumes cloud resources.

VOLUME

Disk-based data storage that is generally represented as an iSCSI target with a file-system that supports extended attributes. This could be persistent or ephemeral (lost during reboots).

QUICK START COMMANDS

OpenStack also provides a command-line tool (OpenStack Client) to access, provision, and manage cloud resources. This section provides tutorials on how to use the OpenStack Client tool for some common scenarios. Please refer to the instructions on installing the client. Also, ensure that the client is authenticated with an openrc file containing required credentials.

KEY MANAGEMENT

CREATE AND UPLOAD SSH KEY

SSH keys are used to SSH into instances launched with OpenStack. OpenStack has a built-in method for inserting an SSH keypair's public half into the authorized keys file for a newly booted instance. Compute can create the private part of the key for you, or you can upload the public part of an existing key pair.

UPLOAD PUBLIC HALF OF EXISTING PAIR

```
$ nova keypair-add --public-key ~/.ssh/id_rsa.pub
<keypair_name>
```

LAUNCH INSTANCE USING CREATED KEY

In order to launch an instance, a minimal set of info is required: image, flavor, and name. Most OpenStack environments also require a provided network device.

LAUNCH INSTANCE

```
$ nova boot --image <image_name> --flavor <flavor_name>
--nic net-id=<network_id> --key-name <keypair_name>
<server_name>
```

You can use the following commands to find required info, such as image, flavor, and network.

DISCOVER AVAILABLE IMAGES

```
$ glance image-list
```

DISCOVER AVAILABLE FLAVORS

```
$ nova flavor-list
```

DISCOVER AVAILABLE NETWORKS

```
$ neutron network-list
```

SERVER MANAGEMENT

LIST INSTANCES

```
$ nova list
```

LIST INSTANCES BY STATUS

```
$ nova list --status build
$ nova list --status active
$ nova list --status error
```

SET INSTANCE METADATA

Instance metadata can be useful for sorting or cataloging instances. Metadata is a free-form key=value store per-instance.

```
$ nova meta <server> set key=value [key=value]
```

REBUILD A SERVER

Rebuilding a server takes fewer arguments than an initial build. It can be an easy way to reset state and start over.

```
$ nova rebuild <server> <image>
```

DISPLAY LOGS FROM A SERVER

OpenStack has the ability to show logs.

```
$ nova console-log <server>
```

ALLOCATE FLOATING IP AND ATTACH TO A VM

Floating IPs create a public reachable IP and direct traffic from it to a configured instance. The IP won't actually exist on the instance; instead, it provides a NAT-like relationship. First, one must be allocated from a provided pool, and then it can be attached to an instance.

DISCOVER AVAILABLE FLOATING IP POOLS

```
$ neutron floatingip-list
```

ALLOCATE FLOATING IP

```
$ nova floating-ip-create <floating_ip_pool>
```

ASSOCIATE FLOATING IP TO SERVER INSTANCE

```
$ nova add-floating-ip <server> <floating_ip>
```

SECURITY GROUPS

UPDATE SECURITY GROUP

A security group is like a firewall for a group of systems. You can provide access rules that reference other groups or CIDR blocks. These groups can be attached to an instance to provide access to that instance.

CREATE A SECURITY GROUP FOR SSH

```
$ nova secgroup-create <name> <description>
```

CREATE AN INBOUND SSH RULE

```
$ nova secgroup-add-rule --proto tcp --dst-port 22
<group_name>
```

ADD SECURITY GROUP TO SERVER INSTANCE

```
$ nova add-secgroup <server> <group_name>
```

IMAGE MANAGEMENT

CREATE IMAGE OF RUNNING SERVER

Server images allow creating preconfigured images for reuse at a later time. They can be used to launch new server instances, and will show up when listing images.

```
$ nova image-create <server> <image>
```

CREATE IMAGE

Creates a new image from an existing image or URL. Optional parameters include image ID, disk format, project, volume to create from, etc.

```
$ glance image-create -name <image>
```

LIST IMAGES

```
$ glance image-list
```

DELETE IMAGE

```
$ glance image-delete <image>
```

BLOCK STORAGE

CREATE A BLOCK STORAGE VOLUME

A block storage volume is a block device that is not directly connected to a running instance. It can be attached to one instance at a time, but will survive if the instance itself fails or is shut down. OpenStack volumes are often an interface to an existing SAN, allowing the use of an easy-to-use API to interact with them.

```
$ cinder create --display-name <name> <size_in_GB>
```

ATTACH A BLOCK STORAGE VOLUME TO A SERVER INSTANCE

When attaching a volume to a server, it is possible to rely on the system to automatically assign a device ID within the instance to the new volume. You can choose to specify if desired. Once the volume is attached, if it is a new volume, it is necessary to format the volume for use. Subsequent usage of the volume will not require formatting.

```
$ nova volume-attach <server> <volume> <device>
```

OBJECT STORAGE

CREATE AN OBJECT STORAGE CONTAINER

OpenStack object storage is not a traditional file system, but rather a distributed storage system for static data such as virtual machine images, photo storage, email storage, backups, and archives. Having no central “brain” or master point of control provides greater scalability, redundancy, and durability. Data is represented by objects, and objects are stored in containers.

```
$ swift post <container_name>
$ swift post <container_name> <file_path>
```

DOWNLOAD AN OBJECT FROM A CONTAINER

```
$ swift download <container_name> <object_name>
```

IDENTITY AND ACCESS CONTROL

CREATE USERS

Cloud Admin can create new users with just a user name. Optional parameters include project ID, email, and password.

```
$ keystone user-create <name>
```

LIST USERS

One can list all the users, optionally filtered by specific project name or project ID.

```
$ keystone user-list [--tenant <tenant>]
```

DELETE USERS

```
$ keystone user-delete <name>
```

CREATE ROLE

```
$ keystone role-create <name>
```

DELETE ROLE

```
$ keystone role-delete <role>
```

LIST ROLES

```
$ keystone role-list
```

ADD ROLE TO TENANT::USER

```
$ keystone user-role-add --user <user> --role <role>
[--tenant <tenant>]
```

CREATE TENANT

```
$ keystone tenant-create <name>
```

LIST TENANTS

```
$ keystone tenant-list
```

SET TENANT PROPERTIES

```
$ keystone tenant-update --property <update> <tenant>
```

DELETE TENANT

```
$ keystone tenant-delete <tenant>
```

SET QUOTAS

Quotas can be set at a project or class basis to limit consumption of resources, such as size of memory (RAM), IP addresses, number of cores, size of storage volumes, etc.

```
$ nova quota-update --properties <update> <tenant>
```

REVIEW QUOTAS

OpenStack can enforce quotas on resource consumption. This command expects a project name; projects are the new name for tenants.

```
$ nova quota-show <tenant>
```

SHOW A LIST OF AVAILABILITY ZONES

OpenStack supports availability zones that may be provided to allow either logical separation or clustering of services.

Typically, these are for power-accessibility or geographical reasons. A zone can be selected when launching new instances.

```
$ nova availability-zone-list
```

TOOLS & ECOSYSTEM

MARKETPLACE

OpenStack customers have a lot of choices to pick from—service providers, vendors, system integrators, distributions, trainers, consultants, and more. In order to help them make informed decisions, the OpenStack Marketplace provides a central repository of information and reviews. It also enables searching based on the region/location of your choice.

SDKS

OpenStack provides two ways to manage your compute, storage, and networking resources in your datacenters: Dashboard and Service Endpoints. OpenStack Service Endpoints are exposed through programmable interfaces (REST APIs). One can access these APIs through any REST client, such as the Client Tools available as part of the OpenStack project. There are also SDKs available for various programming languages to work with these APIs.

OpenStack Client Tools: OpenStack includes official clients for various projects including Compute, Object Storage, Identity, Networking, Database, and Image Services, while additional clients for other projects are in development.

OpenStack SDKs: The official OpenStack SDK for Python is available, and unofficial SDKs are available for various programming languages including Node.js, Go, Java, Ruby, PHP, .NET, and more.

OTHER TOOLS

In realistic cloud deployments, it is inevitable to use a lot of other open-source tools. Here are some of the tools that are widely used along with OpenStack. Note that some of the proprietary OpenStack distributions may not need these tools, as they tend to include custom equivalents.

ORCHESTRATION

Though OpenStack includes the Orchestration service (Heat), many deployments employ popular orchestration tools such as Chef, Puppet, and Ansible. Some deployments also employ homegrown tools.

1. **Chef:** Chef is a popular configuration management tool from Chef (formerly Opscode) that is widely used for OpenStack deployments. There are official cookbooks available to get you started deploying OpenStack using Chef. Chef employs a client-server architecture.

2. **Puppet**: Puppet is another popular configuration management tool from PuppetLabs. One can use the official Puppet modules available to deploy large-scale OpenStack clouds. Puppet also employs client-server architecture.

3. **Ansible**: Ansible software is an agentless configuration management tool from Ansible. Though official Ansible playbooks are not available for OpenStack, one can get started with these playbooks.

MONITORING

Operating large-scale OpenStack clouds needs constant monitoring of different components, whether compute resources, virtual machines, or virtual networks, in order to monitor their health. Such monitoring also enables raising alerts proactively, such as when a compute node approaches specific CPU utilization. OpenStack provides the Telemetry service, which provides event-based infrastructure to enable monitoring; however, it doesn't provide complete monitoring capabilities. Hence, OpenStack cloud deployments typically utilize open-source monitoring tools such as Nagios.

LOGGING

To provide advanced log management capabilities, open-source tools such as LogStash are employed. LogStash provides a centralized capability for logging, parsing, and storing the logs. When used in conjunction with search tools such as ElasticSearch, it provides a powerful and scalable log management solution for your OpenStack cloud.

PROVISIONING

Deploying OpenStack involves provisioning bare metal servers as controller nodes or resource nodes. Popular tools used for provisioning include Chef, Puppet, Ubuntu MaaS, and Crowbar. OpenStack's bare metal service is also an option.

COMMUNITIES

The biggest strength of OpenStack is its vibrant community. More than 18,000 individual contributors from more than 140 countries are part of the community, with more than 70 user groups worldwide. These user groups meet regularly and act as venues for spreading awareness, discussing technical details, and showing camaraderie. If you are new to OpenStack, learning from other users is the easiest way to get started.

USER GROUPS

Some of the largest user groups are the SFBay OpenStack user group, the Indian OpenStack user group, and the China OpenStack user group. To learn from other's experiences, check out your nearest group or start one to represent your region.

DOCUMENTATION

ONLINE DOCUMENTATION

The OpenStack project provides great documentation targeted at System Administrators, Cloud Administrators, End Users, Architects, Operators, and Cloud Developers. The installation guides are available for different platforms (Ubuntu, RHEL, Debian, and SUSE), include references to common Configurations, and are updated with every release. There is also documentation available on advanced topics such as Security and High Availability, which are updated regularly, but not on the same schedule as OpenStack releases.

ASK OPENSTACK

If you have specific questions on OpenStack, learn from experts at Ask OpenStack—a crowd-sourced Q&A site modeled off of the popular site StackOverflow.

OPENSTACK WIKI

OpenStack developers use an internal wiki for project-specific discussions. This is not recommended for end-user consumption.

USE CASES

OpenStack is used across many industry verticals for multiple use cases. Here are some of the most popular, successful use cases:

CLOUD SERVICE PROVIDER

In this use case, OpenStack platform is used to provide virtual infrastructure (compute and storage) as-a-service at larger scale. A typical stack includes open-source tools such as KVM (hypervisor), Ubuntu (OS), OpenStack (cloud management layer), Nagios (monitoring), LogStash (Logging), Kibana (Analytics), Chef (Orchestration), HAProxy (Load Balancers), along with homegrown components. Though commodity hardware is typically used to provide compute and certain kinds of storage, multiple hardware configurations could be deployed depending on the level of service to be provided.

Rackspace Public Cloud is the representative of this use case.

TELCO SERVICE PROVIDER

Telecom service providers have been early adopters of OpenStack for their private cloud requirements. Such use cases typically enable IT-as-a-Service either to internal organizations or partner organizations. The types of workloads and applications that are run vary between providers. The OpenStack community is also working on enabling advanced capabilities specific to Telcos (such as NFV).

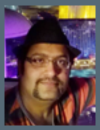
Among many Telcos, Ericsson is reportedly developing such a use case through a major deal with Mirantis.

DEV/TEST

One of the widely adopted use cases of OpenStack is to enable Continuous Integration/Continuous Development (CI/CD) in Dev/Test clouds. Apart from enabling business agility, such Dev/Test cloud workloads increase customer confidence towards running LOB applications on OpenStack clouds. Applications and workloads that run such clouds vary depending on the customer, but typically involve a source code repository (Git for instance), a test harness (Jenkins), a bug database (Atlassian), deployment tools, and more.

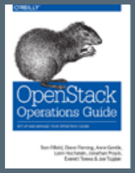
OpenStack's test infrastructure is a great example of a Dev/Test workload. It employs various applications—such as Jenkins, Gerrit, Git, IRC services, etc.—on a highly available CI/CD environment running on OpenStack public clouds (HP Public Cloud, Rackspace Public Cloud).

ABOUT THE AUTHOR



Sriram Subramanian is the Founder and Principal Cloud Specialist at Cloud Don LLC, a cloud services firm offering Research, Analysis and System Integration services. As a popular cloud Influencer and a Helion HP MVP, he offers insights both in his blog and other popular portals. He frequently speaks about challenges and best practices adopting OpenStack at OpenStack Summits and local meetups. His past industry experience includes working for ComputeNext, Microsoft, Intel, and Hitachi, working on a wide spectrum of technologies like Cloud Computing, Virtualization, Compilers, and Low Power Design.

RECOMMENDED BOOK



DZONE, INC.
150 PRESTON EXECUTIVE DR.
CARY, NC 27513

888.678.0399
919.678.0300

REFCARDZ FEEDBACK WELCOME
refcardz@dzone.com

SPONSORSHIP OPPORTUNITIES
sales@dzone.com

ISBN-13: 978-1-936502-77-6
ISBN-10: 1-936502-77-1
50795
9 781936 502776

VERSION 1.0 $7.95