Deutsche Telekom AG

TELEKOM SMALL BUSINESS CLOUD

Business Applications & Cloud Services

created by Adriano Perri

Company, Unit
Group **Innovation+**
DTAG, P&I, Business Applications & Cloud Services
Country
Germany
Place of development
P&I, Dernburgstraße 50, 14057 Berlin

10.02.2016

TELEKOM SMALL BUSINESS CLOUD

Deutsche Telekom AG – Business Applications & Cloud Services

Prolog

This documents gives a overview about the new OpenStack environment and describes the specific deployment of this platform as used in Biere.

Index

1. Introduction

OPS designates the Cloud Operations team, a dedicated group within the Product & Innovation, Cloud Business Marketplace. This group has direct access to cloud management and administration and ensure functionality of the architecture to Application Integrators (AppIntegrators) and the general operational health of the platform to end customers.

This document describes a cloud-based infrastructure service ([1]IaaS). Its goal is to provide a dynamic, scalable and elastic base system for varying applications running in this environment.
It is part of DT's effort to create a SaaS-platform for [2]SME clients. In this context, this project provides the base computing capabilities, which are then utilized by services on the SaaS-layer.

It features OpenStack as its base technology and focuses on advanced virtualization techniques to ensure basic aspects of modern cloud platforms. The primary clients of this platform are AppIntegrators, a group of DT internal employees responsible for setting up and running applications, which are part of the SME Marketplace. Due to clearcut division of responsibilities, there is not intermingling of sensitive data.

Primary goal

- **Provide a secure platform of virtual hosts to Independent Software Vendors (ISVs)**
- **Offer a defined set of functions to Independent Software Vendor to managed and utilize this platform**
- **Ensure basic service levels for platform operations**

[1] Infrastructure as a Service
[2] Storage Made Easy

2. Why OpenStack ?

A couple 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.

Challenges of Today's Cloud Infrastructure Software

- **Traditional hosting is really expensive for a large number of Apps & Customers**
- **Operational costs very high , degree of automation to small**
- **Proprietary cloud platforms no option, license costs**
- **Enhanced hosting environments based on open source technology as interim platform**
  - **requires technical experience, still not very efficient, scalability limited**

OpenStack – Features

- **Open source, open design, open development, open community**
- **Lower-cost production platform**
- **Standard APIs for developers and ISVs result in faster partner on-boarding**
- **Innovative**

2.1 Advantages – App Developers & Global Communications Companies

Why app developers are embracing OpenStack

✓ Global Standard

Open source APIs provide a common platform across public, private and hybrid clouds

✓ Strong Community

Join a vibrant, global community of developers who have your back

✓ Tools & Resources

Get moving faster with a growing ecosystem of development tools, applications and public clouds

Why global communications companies are choosing OpenStack

✓ Network Agility

offers greater flexibility, scalability and resiliency

✓ Cost Savings

both on installation and management

✓ Proven Implementation

by the world's leading telecommunications companies

✓ Open Standards

through ETSI and OPNFV, including reference architecture

✓ Vendor Support

industry-leading solutions are based on OpenStack

2.2 Successful Companies operating with OpenStack

Hong Kong Cyberport Management Company Limited

With a growing base of tech start-ups and entrepreneurs, Hong Kong-based information and communications technology incubator Cyberport is already well on its way to becoming the leading technology hub within the Asia-Pacific region. Operated by the Hong Kong Cyberport Management Company, Cyberport is nurturing these start-ups though cost-saving resource pooling that help accelerate technology adoption through strategic partnerships and initiatives.

> "A lot of cloud platforms are good at talking about cloud, but their actual implementation falls short when it comes to overall capabilities and security," says Chung. "By contrast, OpenStack is a global movement that is constantly evolving and becoming the de facto standard for cloud computing. OpenStack has proven to be the fastest-to-deploy, most cost-effective cloud option."

(source: http://www.openstack.org/user-stories/hong-kong-cyberport-management-company-limited/)

CERN

When one of the world's most prestigious research laboratories decided to embrace cloud computing, it chose OpenStack. Using the world's largest, most complex scientific instruments, CERN, the European Organization for Nuclear Research, continues its study of the most basic constituents of matter – fundamental particles – to help scientists gain a better understanding of the very structure of the universe.

> "OpenStack's technical architecture clearly addressed our needs to run at scale," says Bell. "Also, the technology and developer ecosystem around OpenStack are very vibrant and would enable us to build the services we needed within the cloud. With an open community, we can benefit from the work of the active contributors but also use our engineering skills to enhance the product for others."

(source: http://www.openstack.org/user-stories/cern/)

Visa Inc.

There are two main goals for OpenStack at Visa, he adds. The first is giving developers tools to build products and services that don't require them to worry about underlying management of those applications once they go into a specific environment. The second is to provide a platform that is invisible to developers so they can focus on building value for the business.

> "What we're trying to do with OpenStack is enable developer productivity," says Stan Chan."We needed to partner with someone in engaged with the community and who has the expertise to build OpenStack at scale.

(source: http://superuser.openstack.org/articles/visa-inc-turns-to-openstack-to-boost-developer-productivity)

3. OpenStack Software Overview

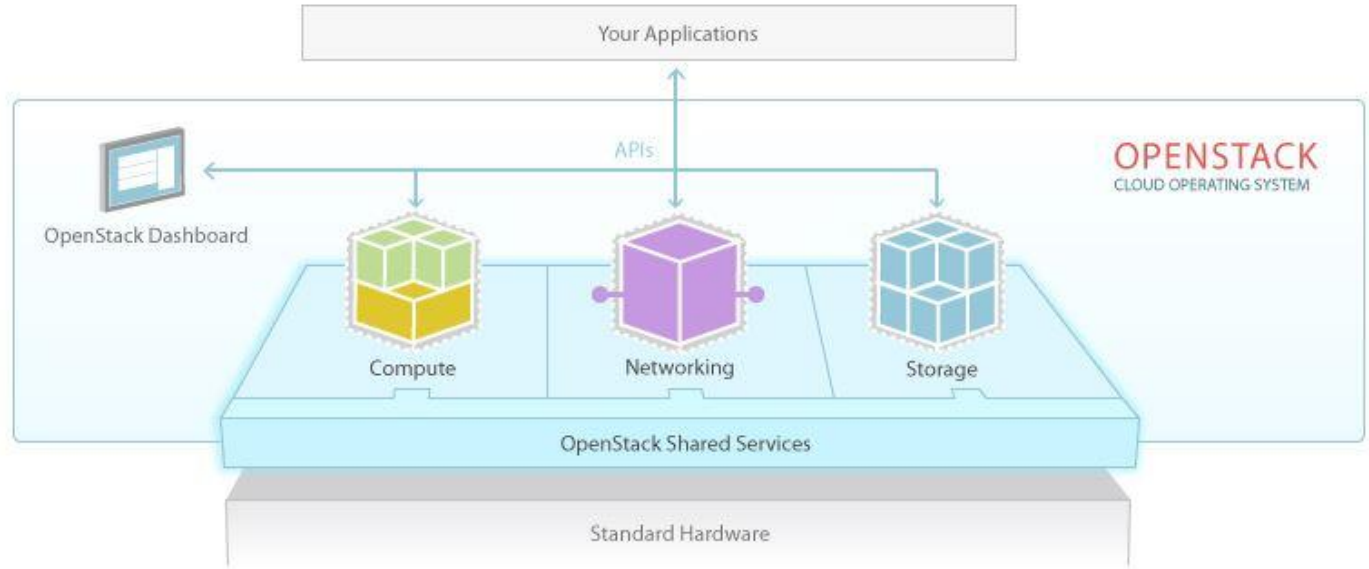
Explanation and Capabilities

OpenStack believes in open source, open design, and open development, all in an open community that encourages participation by anyone. The longterm vision for OpenStack is to produce a ubiquitous open source cloud computing platform that meets the needs of public and private cloud providers regardless of size. By focusing on ease of implementation, massive scalability, a variety of rich features, and tremendous extensibility, the project aims to deliver a practical and reliable cloud solution for all types of organizations. OpenStack is designed to run on commodity hardware. For customers who want to use specific enterprise grade hardware, special drivers are available.

The OpenStack platform is a collection of several interrelated projects to provide capabilities to manage compute, storage, and networking resources in your data centers. It also provides an open API so that all of these resources can be managed through a dashboard that gives administrators control while empowering users to provision resources through a web interface, a command-line client, or software development kits that support the API.

Many OpenStack APIs are extensible, meaning you can keep compatibility with a core set of calls while providing access to more resources and innovating through API extensions. The OpenStack project is a global collaboration of developers and cloud computing technologists.

3.1 Core Services



(source: http://www.openstack.org/software/)

3.1.1 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 and Docker.

The OpenStack cloud operating system enables enterprises and service providers to offer on-demand computing resources, by provisioning and managing large networks of virtual machines. Compute resources are accessible via APIs for developers building cloud applications and via web interfaces for administrators and users. The compute architecture is designed to scale horizontally on standard hardware, enabling the cloud economics companies have come to expect.

Flexible Architecture

OpenStack is architected to provide flexibility as you design your cloud, with no proprietary hardware or software requirements and the ability to integrate with legacy systems and third party technologies.

It is designed to manage and automate pools of compute resources and can work with widely available virtualization technologies, as well as bare metal and highperformance computing (HPC) configurations. Administrators often deploy OpenStack Compute using one of multiple supported hypervisors in a virtualized environment. KVM and XenServer are popular choices for hypervisor technology and recommended for most use cases. Linux container technology such as LXC is also supported for scenarios where users wish to minimize virtualization overhead and achieve greater efficiency and performance. In addition to different hypervisors, OpenStack supports ARM and alternative hardware architectures.

3.1.2 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.

Like other aspects of the cloud operating system, it can be used by administrators and users to increase the value of existing datacenter assets. OpenStack Networking ensures the network will not be the bottleneck or limiting factor in a cloud deployment and gives users real self service, even over their network configurations.

Networking Capabilities

- OpenStack provides flexible networking models to suit the needs of different applications or user groups. Standard models include flat networks or VLANs for separation of servers and traffic.

- OpenStack Networking manages IP addresses, allowing for dedicated static IPs or DHCP. Floating IPs allow traffic to be dynamically rerouted to any of your compute resources, which allows you to redirect traffic during maintenance or in the case of failure.

- Users can create their own networks, control traffic and connect servers and devices to one or more networks.

- The pluggable backend architecture lets users take advantage of commodity gear or advanced networking services from supported vendors.

- Administrators can take advantage of software-defined networking (SDN) technology like OpenFlow to allow for high levels of multi-tenancy and massive scale.

- OpenStack Networking has an extension framework allowing additional network services, such as intrusion detection systems (IDS), load balancing, firewalls and virtual private networks (VPN) to be deployed and managed.

3.1.3 Storage

OpenStack Storage service provides support for both object storage (Swift, we use Ceph S3) and block storage (Cinder). Object Storage supports scale-out, distributed non-structured data and block storage provides virtual storage for virtual instances.

Object Storage is ideal for cost effective, scale-out storage. It provides a fully distributed, API-accessible storage platform that can be integrated directly into applications or used for backup, archiving and data retention. Block Storage allows block devices to be exposed and connected to compute instances for expanded storage, better performance and integration with enterprise storage platforms, such as NetApp, Nexenta and SolidFire.

Object Storage Capabilities

- OpenStack provides redundant, scalable object storage using clusters of standardized servers capable of storing petabytes of data

- 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.

- Objects and files are written to multiple disk drives spread throughout servers in the data center, with the OpenStack software responsible for ensuring data replication and integrity across the cluster.

- Storage clusters scale horizontally simply by adding new servers. Should a server or hard drive fail, OpenStack replicates its content from other active nodes to new locations in the cluster. Because OpenStack uses software logic to ensure data replication and distribution across different devices, inexpensive commodity hard drives and servers can be used in lieu of more expensive equipment.

Block Storage Capabilities

- OpenStack provides persistent block level storage devices for use with OpenStack compute instances.

- The block storage system manages the creation, attaching and detaching of the block devices to servers. Block storage volumes are fully integrated into OpenStack Compute and the Dashboard allowing for cloud users to manage their own storage needs.

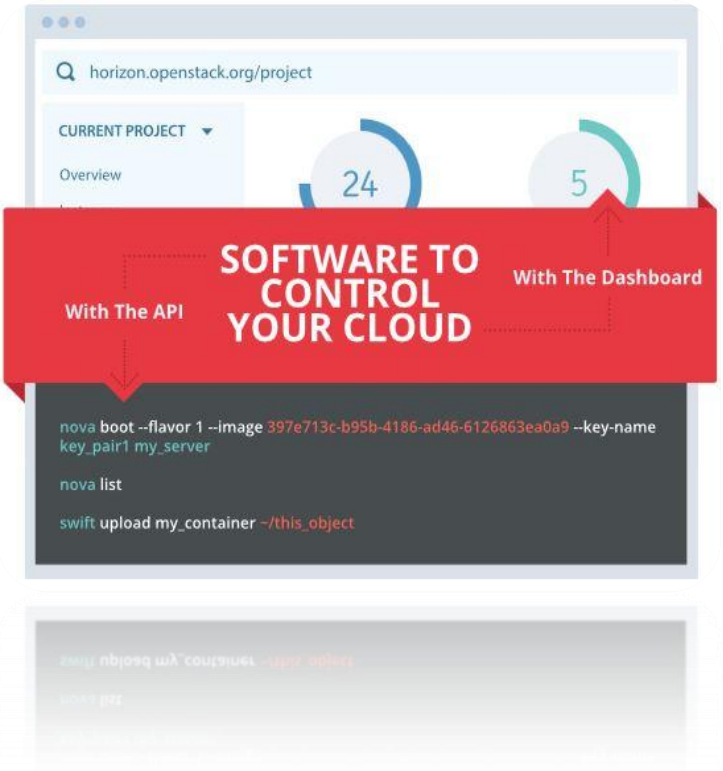
- In addition to using simple Linux server storage, it has unified storage support for numerous storage platforms including Ceph, NetApp, Nexenta, SolidFire, and Zadara.

- Block storage is appropriate for performance sensitive scenarios such as database storage, expandable file systems, or providing a server with access to raw block level storage.

- Snapshot management provides powerful functionality for backing up data stored on block storage volumes. Snapshots can be restored or used to create a new block storage volume.

3.2 Dashboard

The OpenStack dashboard provides administrators and users a graphical interface to access, provision and automate cloud-based resources. The extensible design makes it easy to plug in and expose third party products and services, such as billing, monitoring and additional management tools. The dashboard is also brandable for service providers and other commercial vendors who want to make use of it. The dashboard is just one way to interact with OpenStack resources. Developers can automate access or build tools to manage their resources using the native OpenStack API or the EC2 compatibility API.



(source: http://www.openstack.org/)

Dashboard Capabilities

- The dashboard is an extensible web app that allows cloud administrators and users to control their compute, storage and networking resources.

- As a cloud administrator, the dashboard provides an overall view of the size and state of the cloud. You can create users and projects, assign users to projects and set limits on the resources for those projects.

- The dashboard provides users a self-service portal to provision their own resources within the limits set by administrators.

3.3 OpenStack Services

- we use
- we don't use (currently)

Service	Project name	Description
Dashboard	**"Horizon"**	Provides a web-based self-service portal to interact with underlying OpenStack services, such as launching an instance, assigning IP addresses and configuring access controls.
Compute	**"Nova"**	Manages the lifecycle of compute instances in an OpenStack environment. Responsibilities include spawning, scheduling and decommissioning of virtual machines on demand.
Networking	**"Neutron"**	Enables Network-Connectivity-as-a-Service for other OpenStack services, such as OpenStack Compute. Provides an API for users to define networks and the attachments into them. Has a pluggable architecture that supports many popular networking vendors and technologies.
Object Storage	**"Swift"** **We use "Ceph S3"**	Stores and retrieves arbitrary unstructured data objects via a RESTful, HTTP based API. It is highly fault tolerant with its data replication and scale-out architecture. Its implementation is not like a file server with mountable directories. In this case, it writes objects and files to multiple drives, ensuring the data is replicated across a server cluster.
Block Storage	**"Cinder"**	Provides persistent block storage to running instances. Its pluggable driver architecture facilitates the creation and management of block storage devices.
Identity service	**"Keystone"**	Provides an authentication and authorization service for other OpenStack services. Provides a catalog of endpoints for all OpenStack services.
Image service	**"Glance"**	Stores and retrieves virtual machine disk images. OpenStack Compute makes use of this during instance provisioning.

Message oriented middleware	**"RabbitMQ"**	communication between distributed software components of the OpenStack cluster.
Bare Metal Provisioning	**"Ironic"** **We use "Razor"**	Provides several reference drivers which leverage common technologies like PXE and IPMI, to cover a wide range of hardware. It's pluggable driver architecture also allows vendor-specific drivers to be added for improved performance or functionality not provided by reference drivers.
Telemetry	**"Ceilometer"**	Monitors and meters the OpenStack cloud for billing, benchmarking, scalability, and statistical purposes.
Orchestration	**"Heat"**	Orchestrates multiple composite cloud applications by using either the native HOT template format or the AWS CloudFormation template format, through both an OpenStack-native REST API and a CloudFormation-compatible Query API.
Database service	**"Trove"**	Provides scalable and reliable Cloud Database-as-a-Service functionality for both relational and non-relational database engines.
Data processing service	**"Sahara"**	Provides capabilities to provision and scale Hadoop clusters in OpenStack by specifying parameters like Hadoop version, cluster topology and nodes hardware details.
Multiple Tenant Cloud Messaging	**"Zaqar"**	Multi-tenant cloud messaging service for web and mobile developers. features a fully RESTful API, which developers can use to send messages between various components of their SaaS and mobile applications, by using a variety of communication patterns.
Shared File System Service	**"Manila"**	Provides an open API to manage shares in a vendor agnostic framework. Standard primitives include ability to create, delete, and give/deny access to a share and can be used standalone or in a variety of different network environments.
Security API	**"Barbican"**	Barbican is a REST API designed for the secure storage, provisioning and management of secrets. It is aimed at being useful for all environments, including large ephemeral Clouds.

4. 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 a 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; its 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, which could be of any format — file, music, video, or binary.

Object Storage

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

Project/Tenant

A logical grouping of users; also commonly referred to as tenant.

Quotas

Resource limits on compute and storage resource, set on a per-project basis.

Role

A 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).

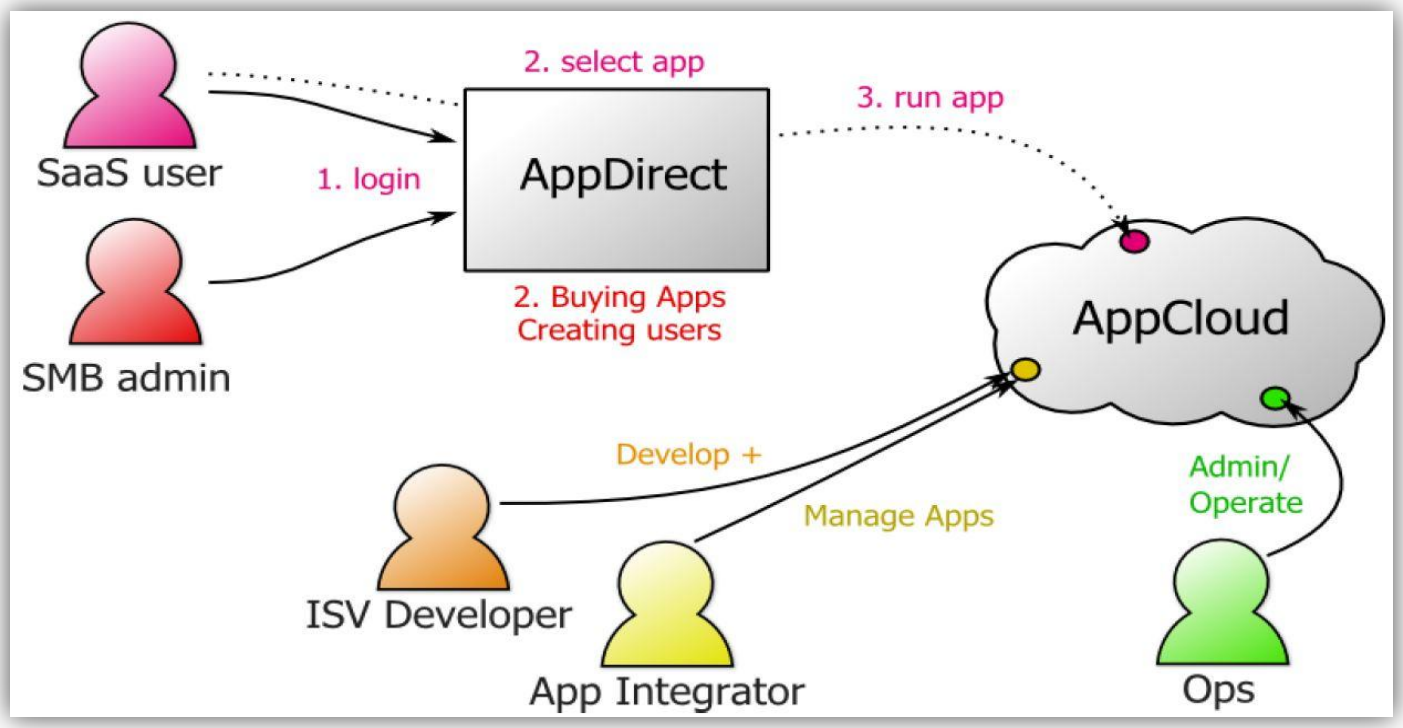
5. TelekomCLOUD – Deployment

(Formerly known as Cloud Business Marketplace)

- We give the resources you need

  (Image Store, virtual systems, API)
- A secure cloud platform (standardized privacy and security concept)
- High data security (Data don't get lost)

5.1 Partner & User Integration Into TelekomCLOUD

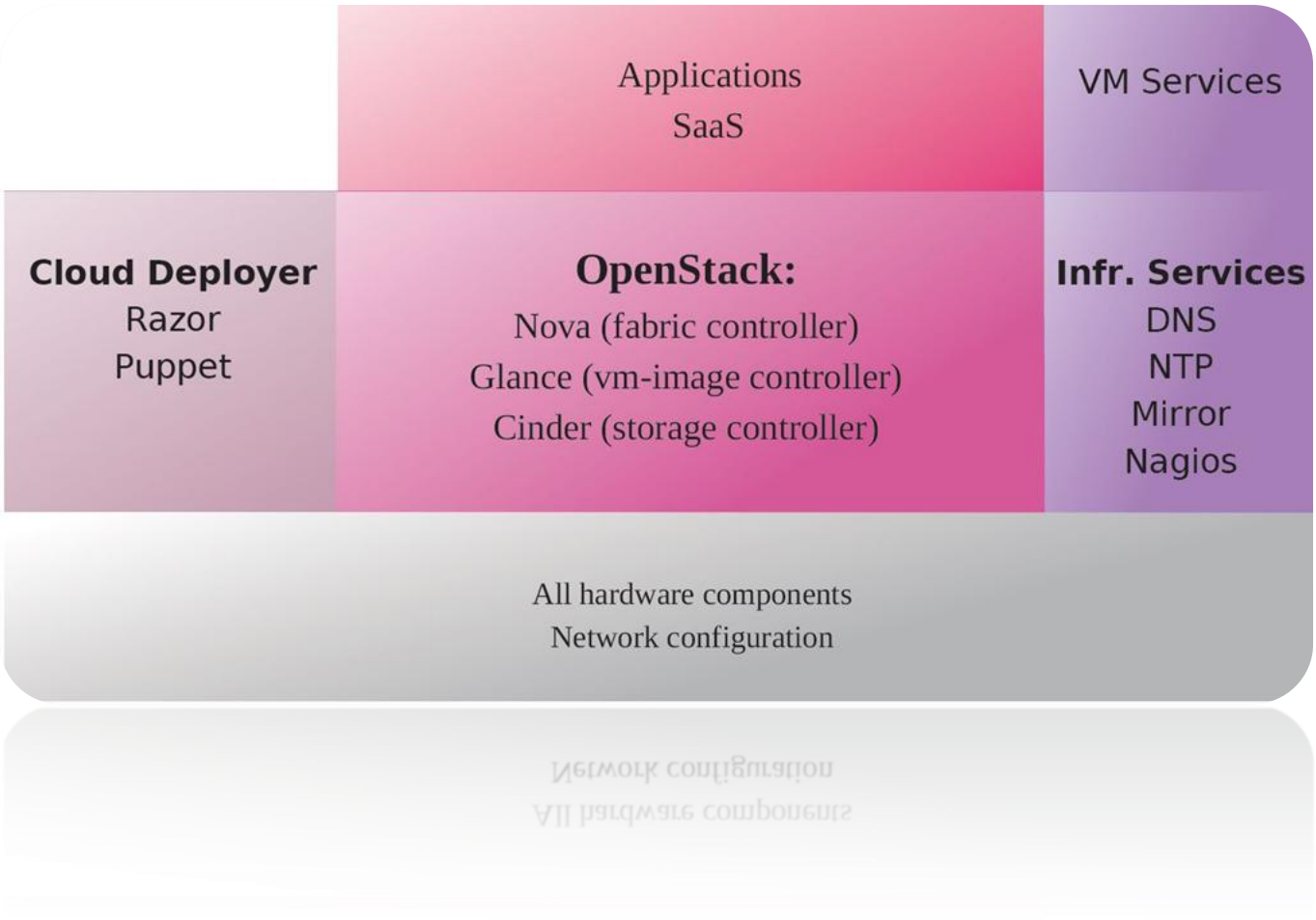


What are the advantages for ISV Developer?

- Hosting in the Telekom environment
- Maximum flexibility on the side of the resource and its application architecture
- Total control for management & maintenance interface
- Low-cost (for example: to test new concepts)

5.2 New OpenStack Environment

The Cloud Hosting Platform 2.0 will be operated in the Biere data center of T-Systems.
The general system composition is seen here:



6. Command List

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.

6.1 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 inject a 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.

To Upload Public Half of Existing Pair

```
1
2  $ nova keypair-add --public-key ~/.ssh/id_rsa.pub
3    <keypairname>
```

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

```
1
2  $ nova boot --image <image_name> --flavor <flavor_name
3    --nic net-id=<network_id> --key-name <keypairname> <server_name>
```

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

Discover Available Images

```
1
2  $ glance image-list
```

Discover Available Flavors

```
1
2  $ nova flavor-list
```

Discover Available Networks

```
1
2  $ neutron network-list
```

6.2 Server Management

List Instances

```
1
2  $ nova list
```

List Instances by Status

```
1
2  $ nova list --status build
3  $ nova list --status active
4  $ nova list --status error
```

Set Instance Metadata

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

```
1
2  $ nova meta <server> set key=value [key=value]
```

Rebuild a Server

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

```
1
2  $ nova rebuild <server> <image>
```

Display Logs From a Server

OpenStack has the ability to show logs.

```
1
2  $ nova console-log <server>
```

Allocate Floating IP and Attach to a VM (Virtual Machine)

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.

Dicover Available Floating IP Pools

```
1
2  $ neutron floatingip-list
```

Allocate Floating IP

```
1
2  $ nova floating-ip-create <floating ip pool>
```

Associate Floating IP to Server Instance

```
1
2  $ nova add-floating-ip <server> <floating-p>
```

6.3 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 (Classless Inter-Domain Routing) blocks. These groups can be attached to an instance to provide access to that instance.

Create a Security Group for SSH

```
1
2  $ nova secgroup-create <name> <description>
```

Create an Inbound SSH Rule

```
1
2  $ nova secgroup-add-rule --proto tcp --dst-port 22 <group_name>
```

Add Security Group to Server Instance

```
1
2  $ nova add-secgroup <server> <group_name>
```

6.4 Image Management

Create Image of Running Server

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

```
1
2  $ 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.

```
1
2   $ glance image-create –name <image>
```

List Images

```
1
2   $ glance image-list
```

Delete Images

```
1
2   $ glance image-delete <image>
```

6.5 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.

```
1
2   $ nova volume-create --dislplay-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.

```
1
2   $ nova volume-attach <server> <volume> <device>
```

6.6 Ceph S3-Object Storage

Ceph delivers object, block, and file storage in one unified system. Ceph is highly reliable, easy to manage, and free. The power of Ceph can transform your company's IT infrastructure and your ability to manage vast amounts of data. Ceph delivers extraordinary scalability–thousands of clients accessing petabytes to exabytes of data. A Ceph Node leverages commodity hardware and intelligent daemons, and a Ceph Storage Cluster accommodates large numbers of nodes, which communicate with each other to replicate and redistribute data dynamically.

Service Ops

List Buckets

GET / returns a list of buckets created by the user making the request. GET / only returns buckets created by an authenticated user. You cannot make an anonymous request.

```
1
2  $ GET / HTTP/1.1
3  $ Host: cname.domain.com
4  $ Authorization: AWS {access-key}:{hash-of-header-and-secret}
```

Bucket Ops

PUT Bucket

Creates a new bucket. To create a bucket, you must have a user ID and a valid AWS Access Key ID to authenticate requests. You may not create buckets as an anonymous user.

```
1
2  $ PUT /{bucket} HTTP/1.1
3  $ Host: cname.domain.com
4  $ x-amz-acl: public-read-write
5  $ Authorization: AWS {access-key}:{hash-of-header-and-secret}
```

Delete Bucket

Deletes a bucket. You can reuse bucket names following a successful bucket removal.

```
1
2  $ DELETE /{bucket} HTTP/1.1
3  $ Host: cname.domain.com
4  $ Authorization: AWS {access-key}:{hash-of-header-and-secret}
```

GET Bucket

Returns a list of bucket objects.

```
1
2  $ GET /{bucket}?max-keys=25 HTTP/1.1
3  $ Host: cname.domain.com
```

Get Bucket Location

Retrieves the bucket's region. The user needs to be the bucket owner to call this. A bucket can be constrained to a region by providing LocationConstraint during a PUT request.

```
1
2  $ GET /{bucket}?location HTTP/1.1
3  $ Host: cname.domain.com
4  $ Authorization: AWS {access-key}:{hash-of-header-and-secret}
```

Get Bucket ACL (Access Control List)

Retrieves the bucket access control list. The user needs to be the bucket owner or to have been granted READ_ACP permission on the bucket.

```
1
2  $ GET /{bucket}?acl HTTP/1.1
3  $ Host: cname.domain.com
4  $ Authorization: AWS {access-key}:{hash-of-header-and-secret}
```

PUT Bucket ACL

Sets an access control to an existing bucket. The user needs to be the bucket owner or to have been granted WRITE_ACP permission on the bucket.

```
1
2  $ PUT /{bucket}?acl HTTP/1.1
```

Object Ops

Put Object

Adds an object to a bucket. You must have write permissions on the bucket to perform this operation.

```
1
2  $ PUT /{bucket}/{object} HTTP/1.1
```

Remove Object

Removes an object. Requires WRITE permission set on the containing bucket.

```
1
2  $ DELETE /{bucket}/{object} HTTP/1.1
```

Get Object

Retrieves an object from a bucket within RADOS.

```
1
2  $ GET /{bucket}/{object} HTTP/1.1
```

Get Object Info

Returns information about object. This request will return the same header information as with the Get Object request, but will include the metadata only, not the object data payload.

```
1
2  $  HEAD /{bucket}/{object} HTTP/1.1
```

6.7 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.

```
1
2  $ keystone user-create <name>
```

List Users

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

```
1
2  $ keystone user-list [--tenant <tenant>]
```

Delete Users

```
1
2  $ keystone user-delete <name>
```

Create Role

```
1
2  $ keystone role-create <name>
```

Delete Role

```
1
2  $ keystone role-delete <role>
```

List Roles

```
1
2  $ keystone role-list
```

Add Role to Tenant::User

```
1
2  $ keystone user-role-add --user <user> --role <role>
3  [--tenant <tenant>]
```

Create Tenant

```
1
2  $ keystone tenant-create <name>
```

List Tenants

```
1
2  $ keystone tenant-list
```

Set Tenant Properties

```
1
2  $ keystone tenant-update -property <update> <tenant>
```

Delete Tenant

```
1
2  $ 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.

```
1
2  $ 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.

```
1
2  $ 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.

```
1
2  $ nova availability-zone-list
```

7.References

OpenStack Official Website
http://www.openstack.org/

Wiki
https://wiki.openstack.org/wiki/Main_Page

OpenStack Documentation
http://docs.openstack.org/

Ceph Documentation
http://docs.ceph.com/docs/master/start/intro/

OpenStack Projects
http://www.openstack.org/software/project-navigator/

OpenStack End User Guide
http://docs.openstack.org/user-guide/

OpenStack Developer and Community Infrastructure Documentation
http://docs.openstack.org/infra/

OpenStack API Complete Reference
http://developer.openstack.org/api-ref.html

OpenStack Command-Line Interface Reference
http://docs.openstack.org/cli-reference/

OpenStack User Stories
http://www.openstack.org/user-stories/

OpenStack Superuser
http://superuser.openstack.org/articles/section/user-stories

Other used Sources
https://dzone.com/refcardz/getting-started-openstack-most
https://en.wikipedia.org/wiki/OpenStack

Company, Unit
Group Innovation+
DTAG, P&I, Business Applications & Cloud Services
Country
Germany
Place of development
P&I, Dernburgstraße 50, 14057 Berlin