
OTUS
ОНЛАЙН-ОБРАЗОВАНИЕ
Онлайн-образование

# Меня хорошо видно && слышно?

Ставьте  , если все хорошо

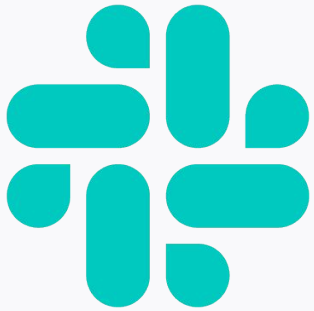
Напишите в чат, если есть проблемы

# Проверить, идет ли запись!

# Openstack

Шаповалов Евгений

# Комьюнити



Все ли есть в закрытой группе в Slack?

Если нет, то напишите комьюнити-менеджеру Евгении

e.skitkovskaya@otus.ru

@Evgeniya OTUS

# Правила вебинара

Активно участвуем

Задаем вопросы в чат или голосом

Off-topic обсуждаем в Slack #канал группы или #general

Вопросы вижу в чате, могу ответить не сразу

# Преподаватель


С 2011 года занимался QA в проектах Parallels и Ingram Micro. Тестировал серверную виртуализацию и автоматизацию у хостинг провайдеров.

В 2019 году занялся напрямую разработкой и DevOps практиками. Работал над билд системой и инфраструктурой разработки в проектах Juniper Networks.

Сейчас работаю инженером-разработчиком над проектами с виртуализацией с  Openstack и k8s.

# Карта курса

1 Кластеризация

2 HighLoad Web

3 СУБД

4 Виртуализация и контейнеризация

5 Оркестрация - Openstack

★ Опрос по программе - каждый месяц в ЛК

# Маршрут вебинара

Cloud computing

Обзор возможностей

Компоненты Openstack

Рефлексия

# **Цели и смысл вебинара** | На занятии вы сможете

1. Познакомиться с разработкой Openstack
2. Познакомиться с основными компонентами Openstack
3. Познакомиться с dashboard

# История Openstack

# История Cloud Computing

- Сначала компании использовали дорогие датацентры
- На разворачивание сервера требовалось большего времени
- Нельзя было быстро реагировать

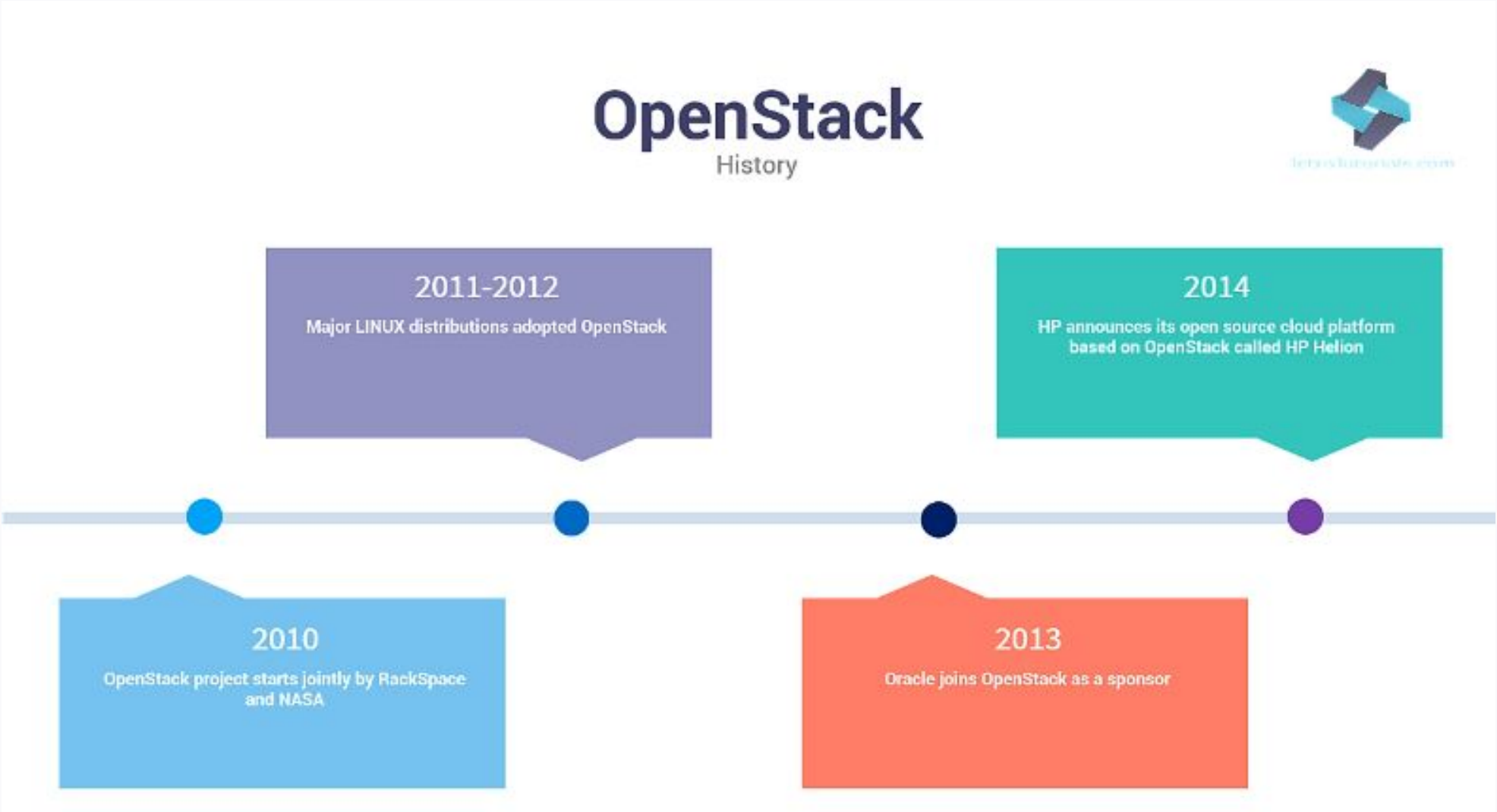
IaaS = Infrastructure as a Service( Инфраструктура как сервис)

- Самостоятельный деплой виртуальных машин в облако
- Замена традиционным датацентрам
- Затраты только на использование компьют времени
- Очень быстрый и гибкий способ реагировать на бизнес требования

# Успех Openstack

- Открытый код
- Основа на успехе Linux
- Openstack foundation агрегирует в себе успех многих компаний
- Возможность составлять продукты с премиальными фичами
- Технологии Openstack используются и вне его

# История Openstack



# Кто использует Openstack

## Companies involved in OpenStack

OpenStack was originally launched in July 2010 by Rackspace and NASA as an open source initiative that combined NASA's Nebula platform and Rackspace's Cloud Files platform.

Today, hundreds of companies contribute to OpenStack code, with many more playing a part in the broader community.

INFLUENTIAL CONTRIBUTORS TO OPENSTACK

CANONICAL IBM redhat HUAWEI openSUSE rackspace

# Кто использует Openstack

# Кто использует Openstack

oud Architecture

## OpenStack at CERN by numbers

Смотреть ... Поделиться

~ 5000 Compute Nodes (~130k cores)
- ~ 4800 KVM
- ~ 200 Hyper-V

~ 16000 VMs running

~ 2400 Images ( ~ 30 TB in use)

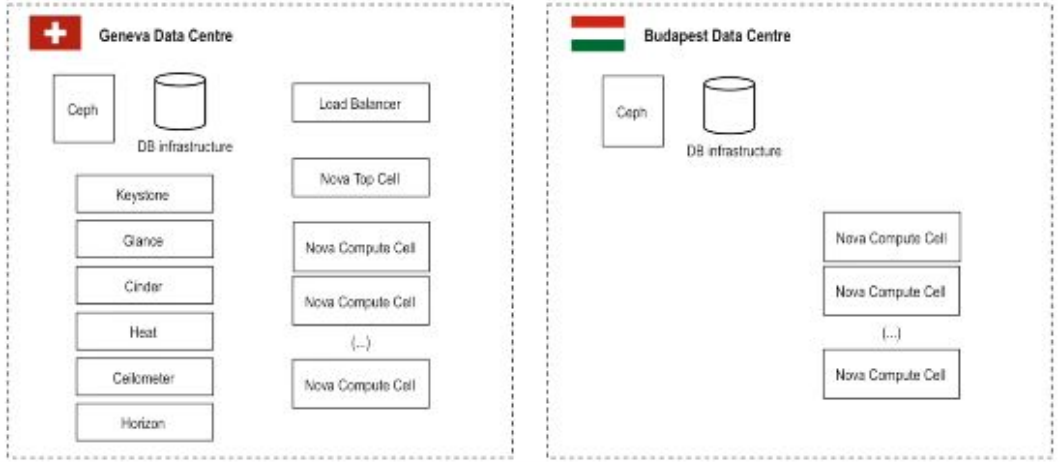
~ 1800 Volumes ( ~ 800 TB allocated)

~ 2000 Users
~ 2300 Projects


Number of VMs created (green) and VMs deleted (red) every 30 minutes

8

## Architecture Overview




12

# Разработка Openstack

# Openstack проект на Python

- > 12 млн строк кода
- > 1 млн коммитов
- > 15000 контрибьютеров
- > 1500 проектов
- около 1000 проектов на Python

# Основные котрибьюторы в  Openstack

## Reviews by Company

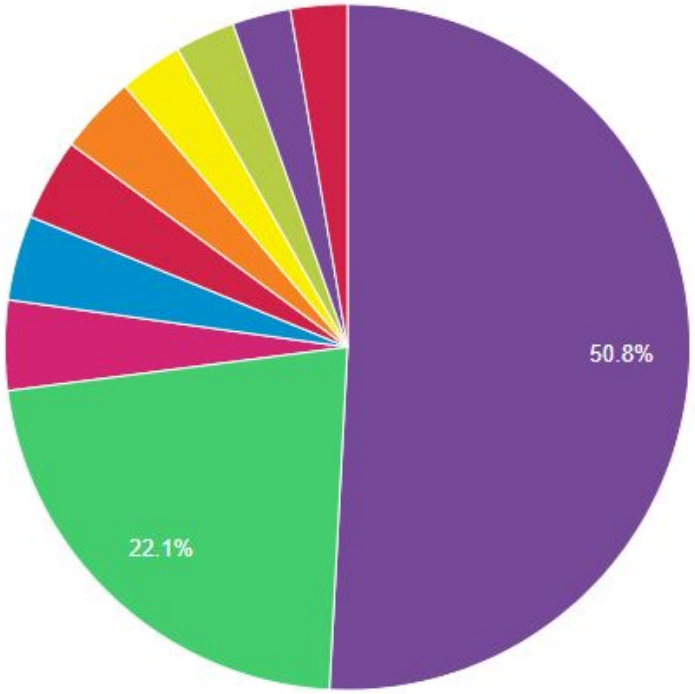
Show 10 entries

Search

| # | Company | Reviews |
|---|---|---|
| 1 | Red Hat | 4242 |
| 2 | Yovole | 354 |
| | *independent | 334 |
| 3 | Canonical | 323 |
| 4 | AT&T | 307 |
| 5 | BBC | 249 |
| 6 | Dell EMC | 236 |
| 7 | NEC | 230 |
| 8 | Ericsson | 223 |
| 9 | StackHPC | 160 |

Showing 1 to 10 of 71 entries

Previous Next



# Схема разработки Openstack

opendev

Navigation



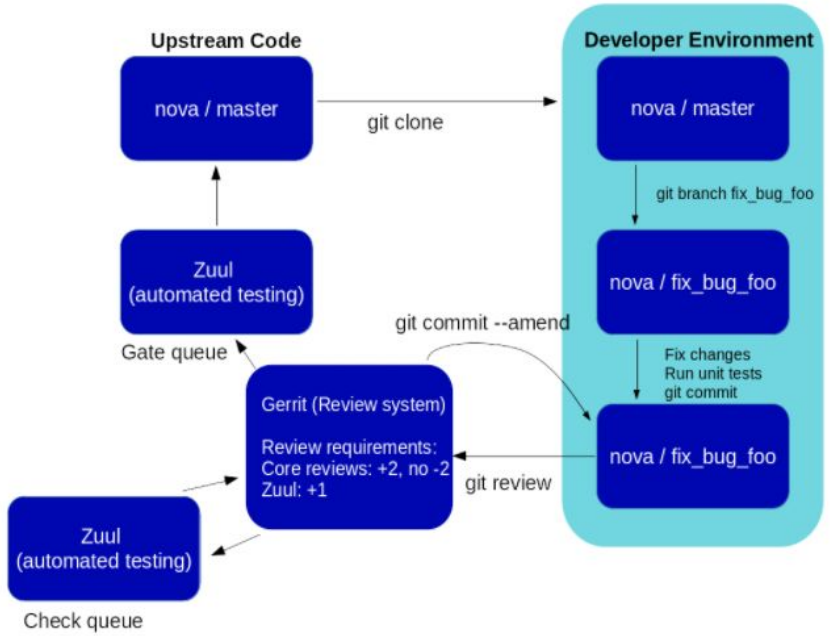
Quick search

Go

## Getting Started

### The OpenDev Workflow

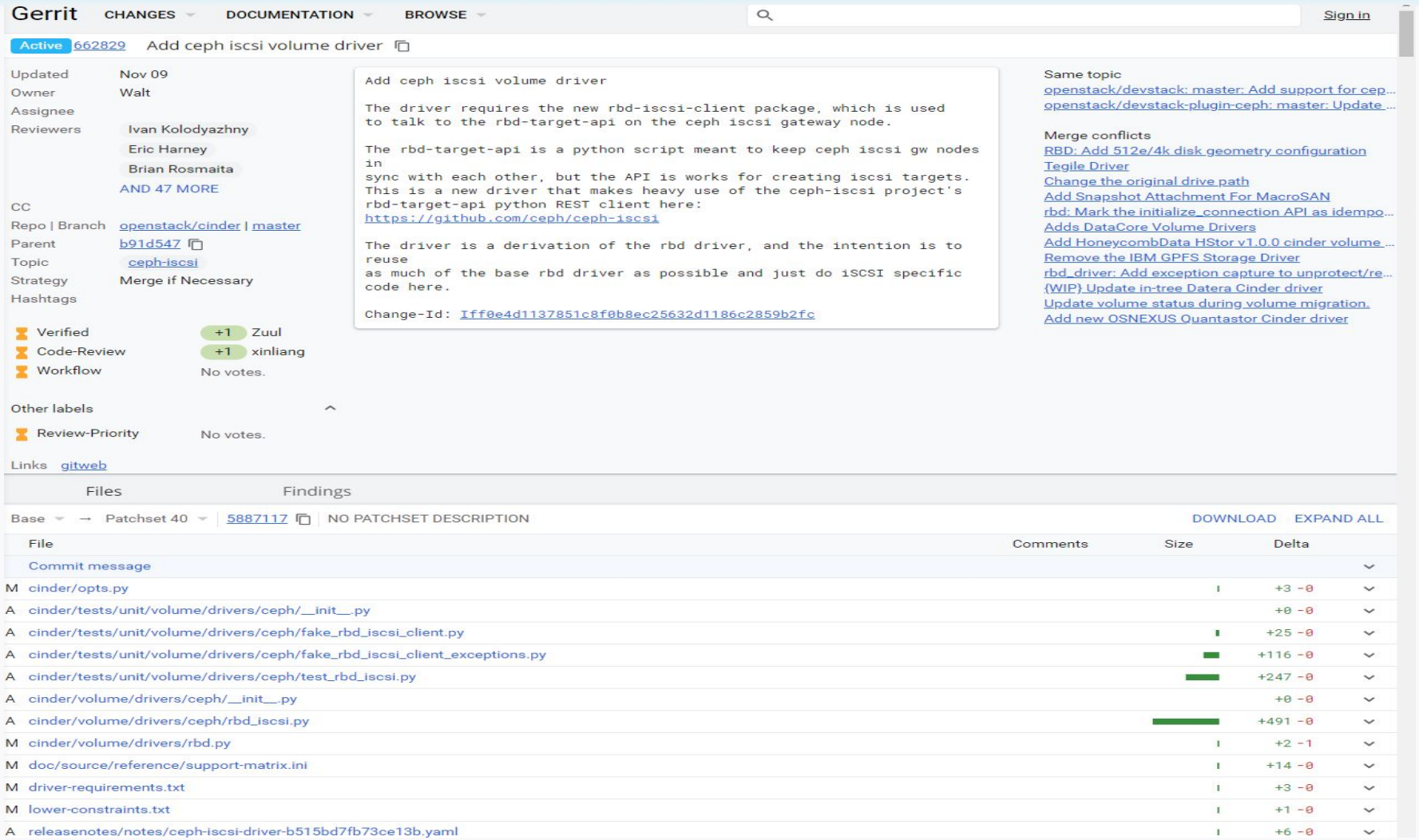
The OpenDev workflow is centered around Gerrit, which uses the concept of **changes** rather than Pull Requests. To propose a change to a git repository, you start by cloning the repository you're interested in, then create a branch to work in. You curate a commit on that branch, then propose it to Gerrit using the **git-review** tool:

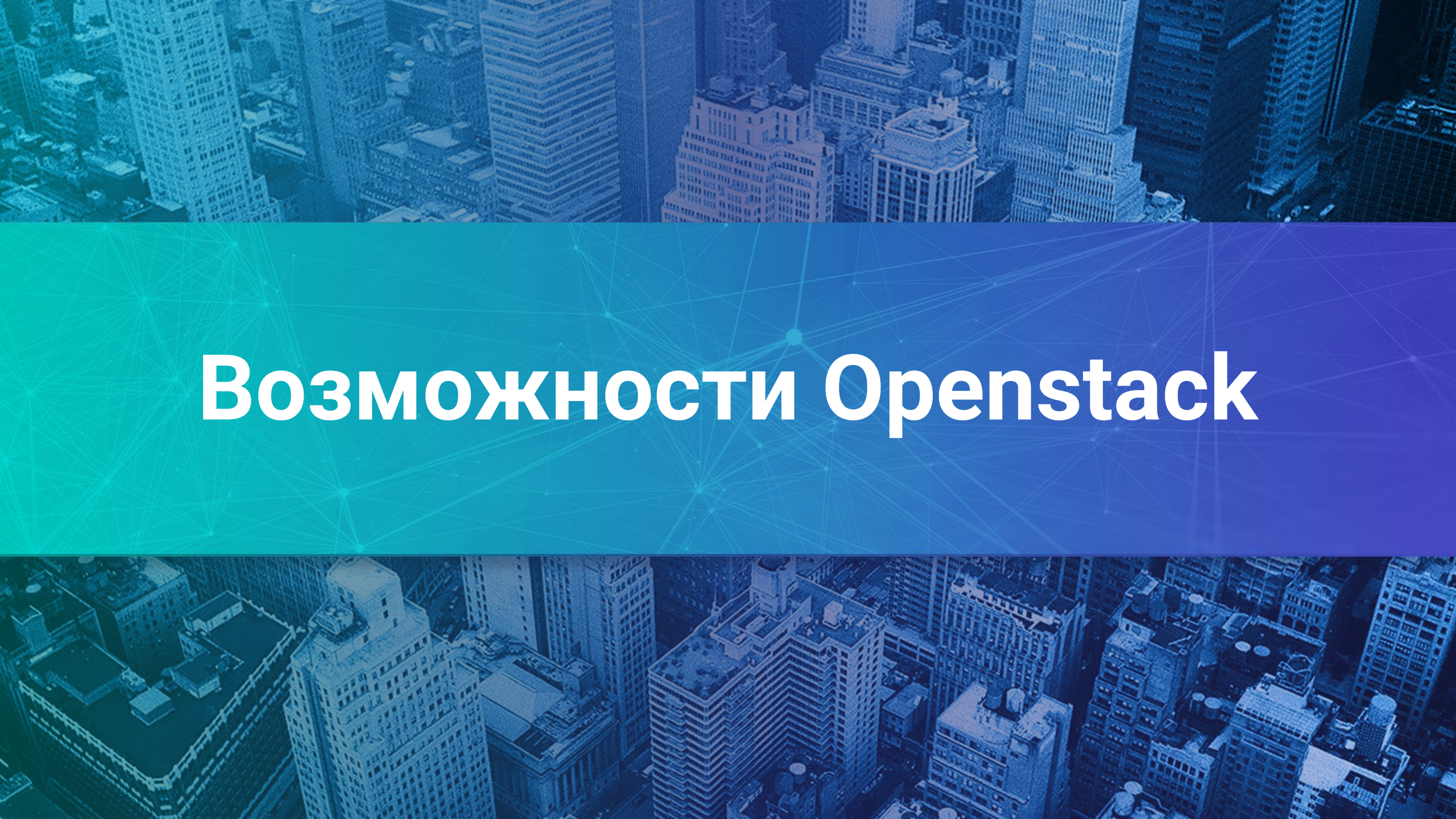


# Openstack релизы

| Series | Status | Initial Release Date | Next Phase | EOL Date |
|---|---|---|---|---|
| Wallaby | Development | 2021-04-14 *estimated* (schedule) | Maintained *estimated 2021-04-14* | |
| Victoria | Maintained | 2020-10-14 | Extended Maintenance *estimated 2022-04-18* | |
| Ussuri | Maintained | 2020-05-13 | Extended Maintenance *estimated 2021-11-12* | |
| Train | Maintained | 2019-10-16 | Extended Maintenance *estimated 2021-05-12* | |
| Stein | Extended Maintenance (see note below) | 2019-04-10 | Unmaintained *TBD* | |
| Rocky | Extended Maintenance (see note below) | 2018-08-30 | Unmaintained *TBD* | |
| Queens | Extended Maintenance (see note below) | 2018-02-28 | Unmaintained *TBD* | |
| Pike | Extended Maintenance (see note below) | 2017-08-30 | Unmaintained *TBD* | |
| Ocata | Extended Maintenance (see note below) | 2017-02-22 | Unmaintained *estimated 2020-06-04* | |
| Newton | End Of Life | 2016-10-06 | | 2017-10-25 |
| Mitaka | End Of Life | 2016-04-07 | | 2017-04-10 |
| Liberty | End Of Life | 2015-10-15 | | 2016-11-17 |
| Kilo | End Of Life | 2015-04-30 | | 2016-05-02 |
| Juno | End Of Life | 2014-10-16 | | 2015-12-07 |
| Icehouse | End Of Life | 2014-04-17 | | 2015-07-02 |
| Havana | End Of Life | 2013-10-17 | | 2014-09-30 |
| Grizzly | End Of Life | 2013-04-04 | | 2014-03-29 |
| Folsom | End Of Life | 2012-09-27 | | 2013-11-19 |
| Essex | End Of Life | 2012-04-05 | | 2013-05-06 |
| Diablo | End Of Life | 2011-09-22 | | 2013-05-06 |
| Cactus | End Of Life | 2011-04-15 | | |
| Bexar | End Of Life | 2011-02-03 | | |
| Austin | End Of Life | 2010-10-21 | | |

# Openstack код ревью





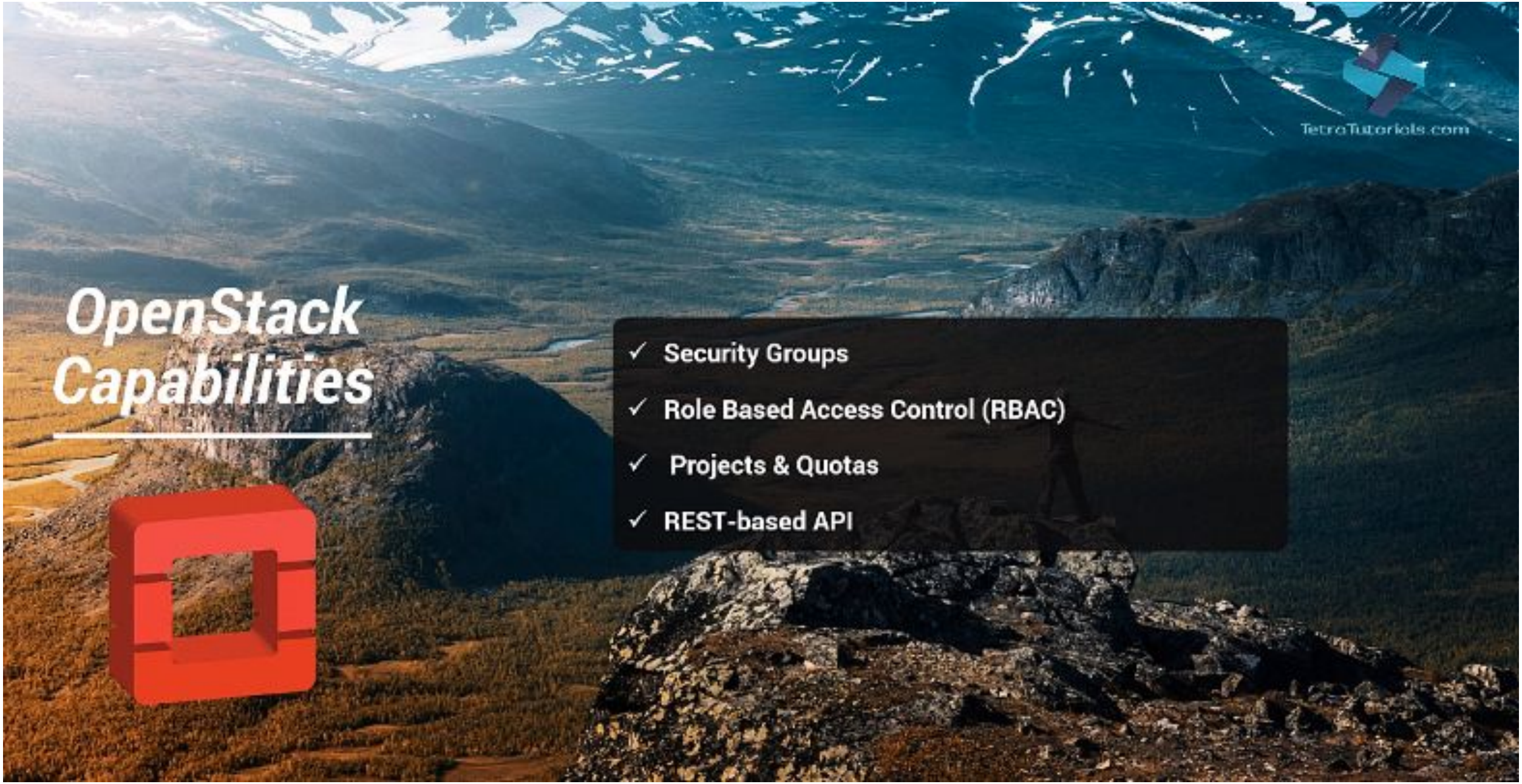
# Возможности Openstack

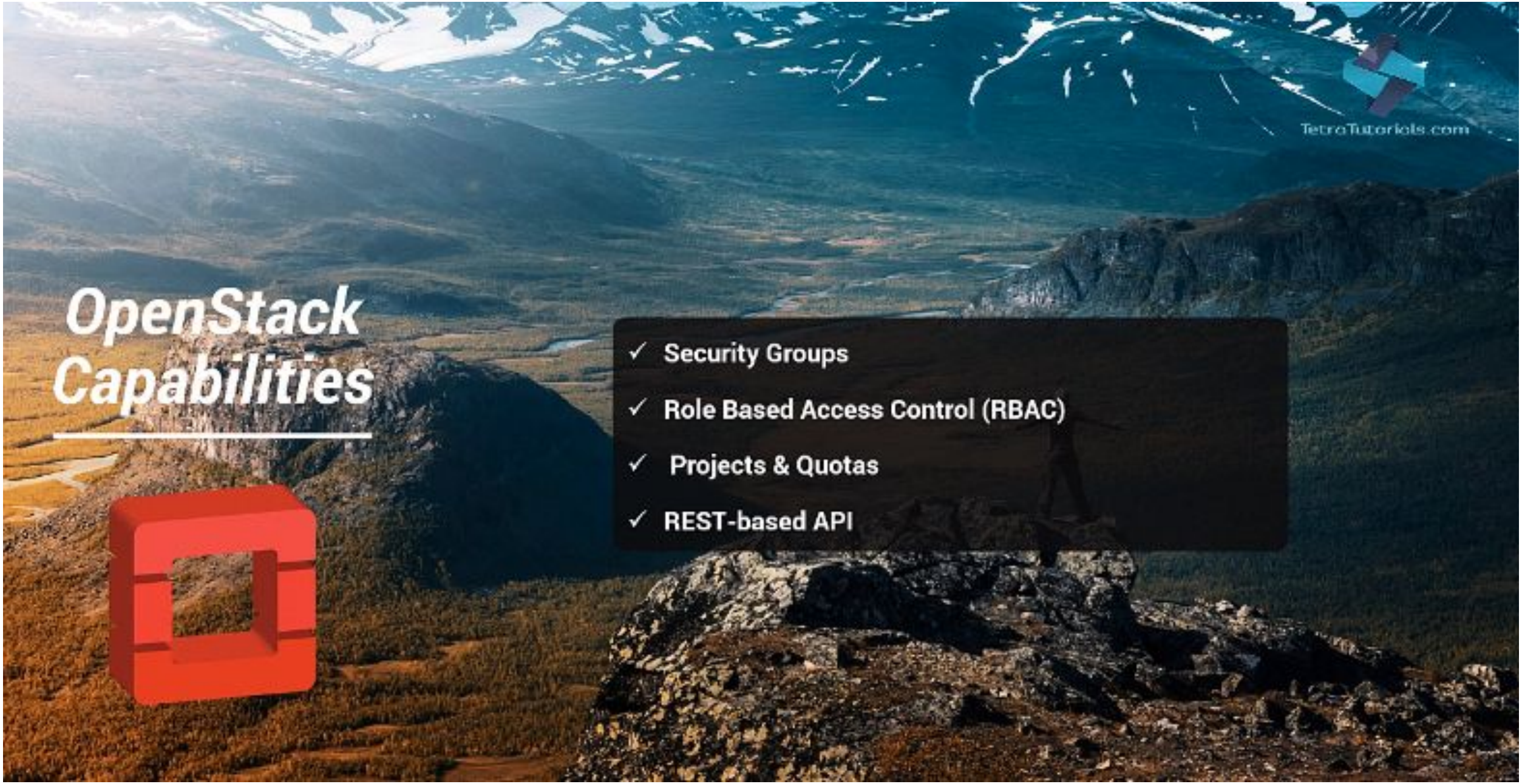
# OpenStack Capabilities

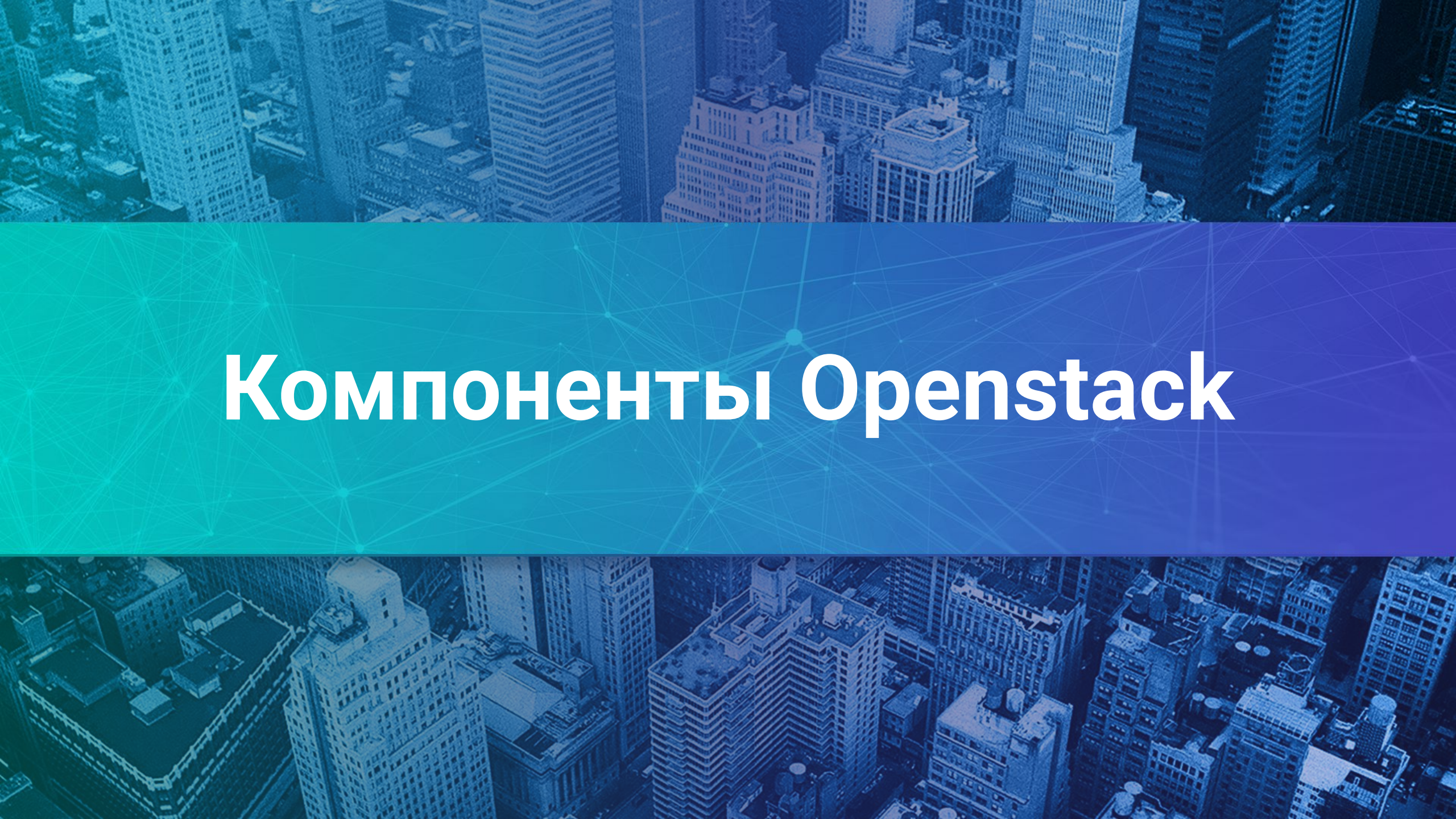
- ✓ Self-service Instance life cycle management: run, reboot, suspend, resize and terminate instances
- ✓ Management of compute resources: CPU, memory, disk, and network interfaces
- ✓ Management of Network Resources (Flat, Flat DHCP, VLAN DHCP and IPv6)
- ✓ API with rate limiting and authentication to manage who has access to compute resources
- ✓ Distributed and asynchronous architecture for massively scalable and highly available system
- ✓ Virtual Machine (VM) image management i.e. store, import, share, and query images
- ✓ Floating IP addresses i.e. Ability to assign (and re-assign) IP addresses to VMs

# OpenStack Capabilities

- ✓ Self-service Instance life cycle management: run, reboot, suspend, resize and terminate instances
- ✓ Management of compute resources: CPU, memory, disk, and network interfaces
- ✓ Management of Network Resources (Flat, Flat DHCP, VLAN DHCP and IPv6)
- ✓ API with rate limiting and authentication to manage who has access to compute resources
- ✓ Distributed and asynchronous architecture for massively scalable and highly available system
- ✓ Virtual Machine (VM) image management i.e. store, import, share, and query images
- ✓ Floating IP addresses i.e. Ability to assign (and re-assign) IP addresses to VMs

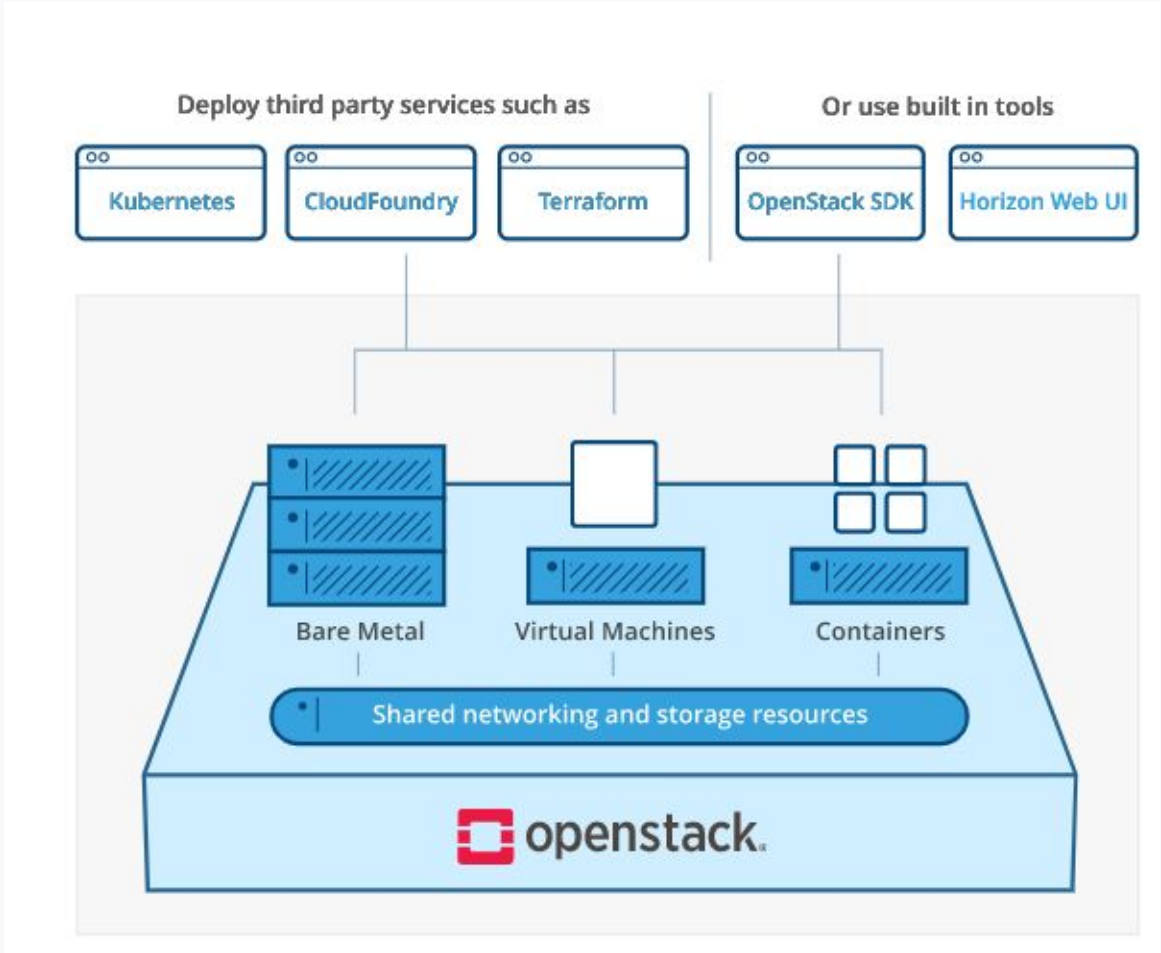

TetroTutorials.com
OpenStack Capabilities
✓ Security Groups
✓ Role Based Access Control (RBAC)
✓ Projects & Quotas
✓ REST-based API


TetroTutorials.com
OpenStack Capabilities
✓ Security Groups
✓ Role Based Access Control (RBAC)
✓ Projects & Quotas
✓ REST-based API

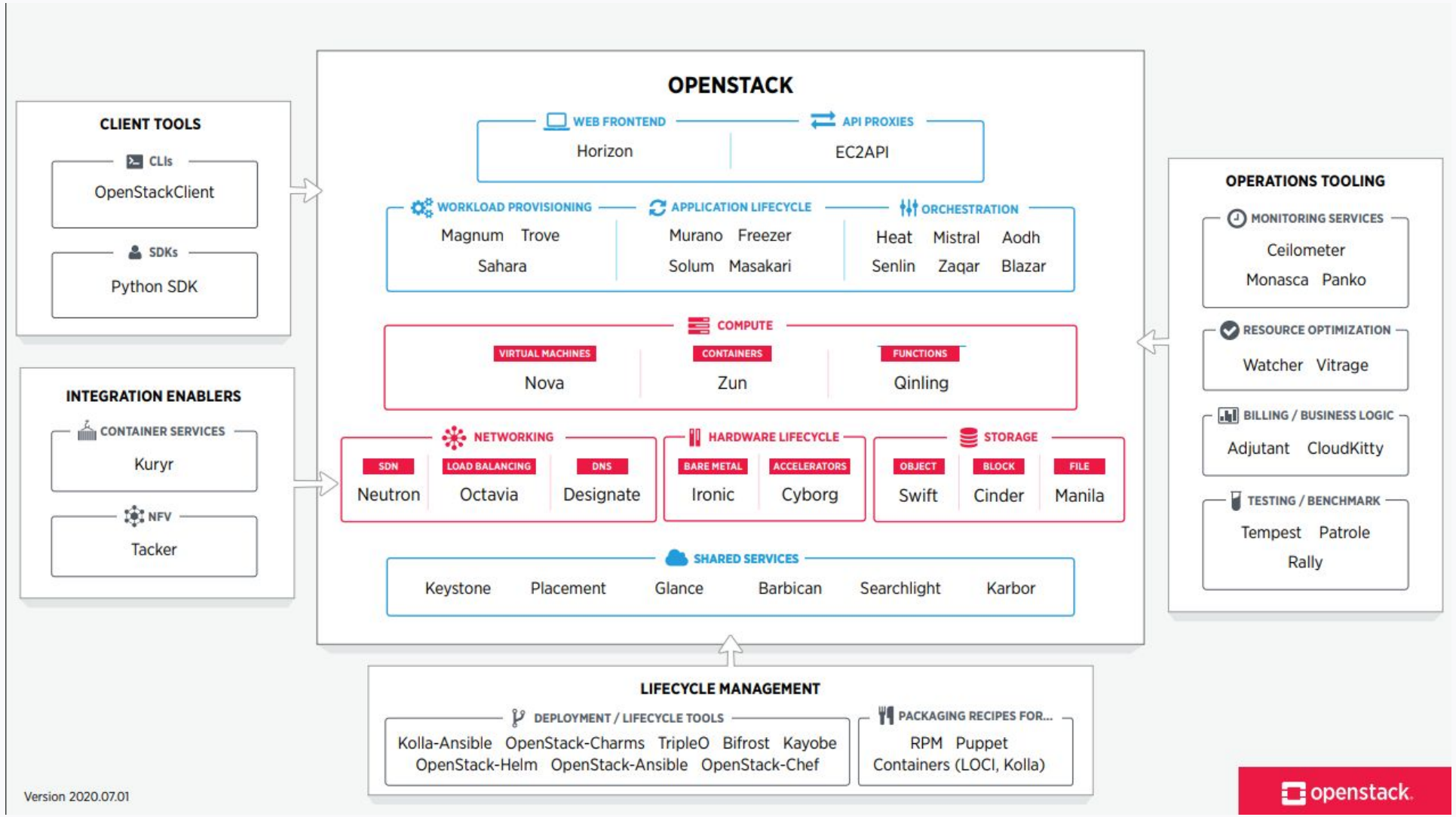


# Компоненты Openstack

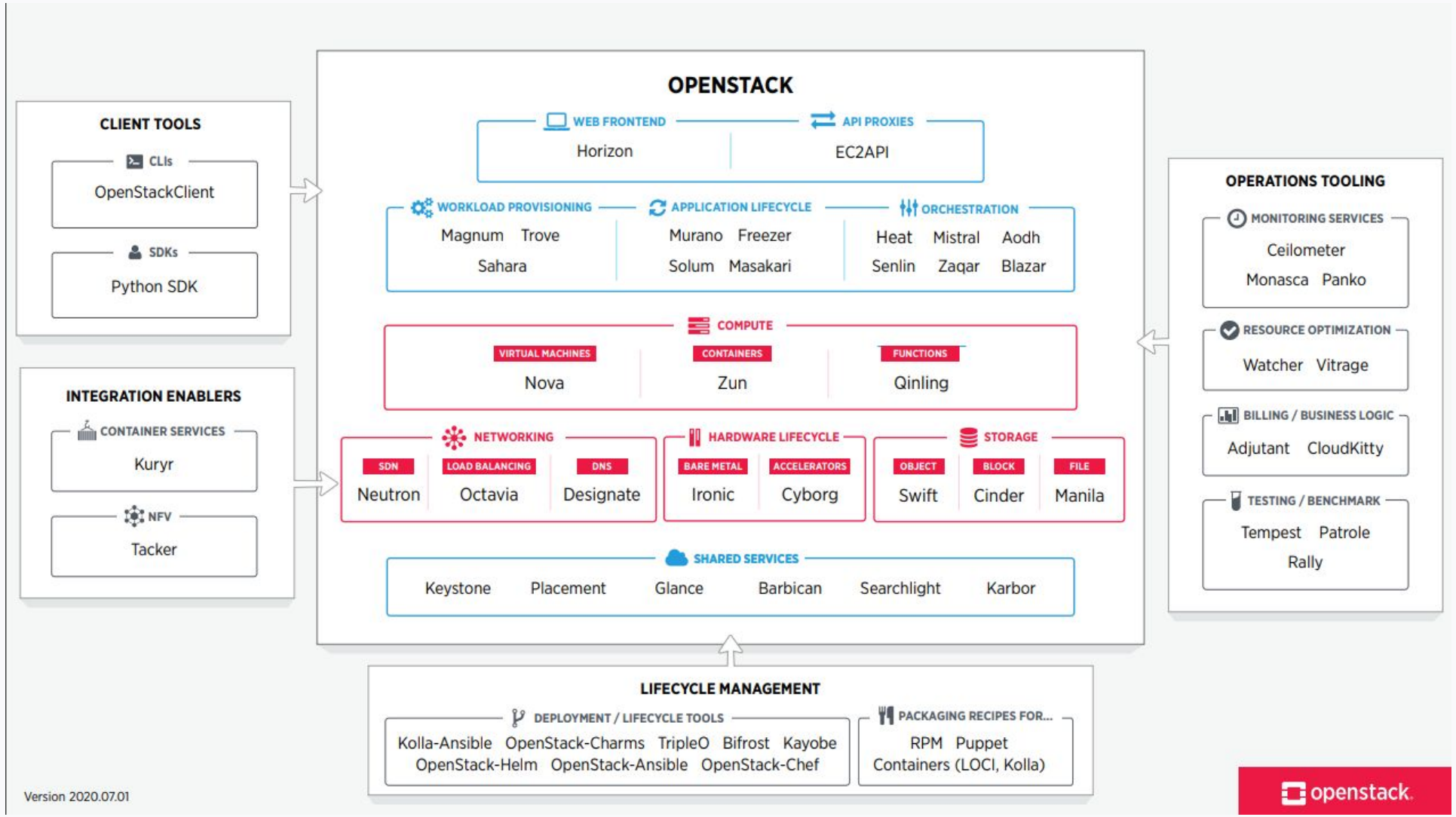
# Openstack общая схема



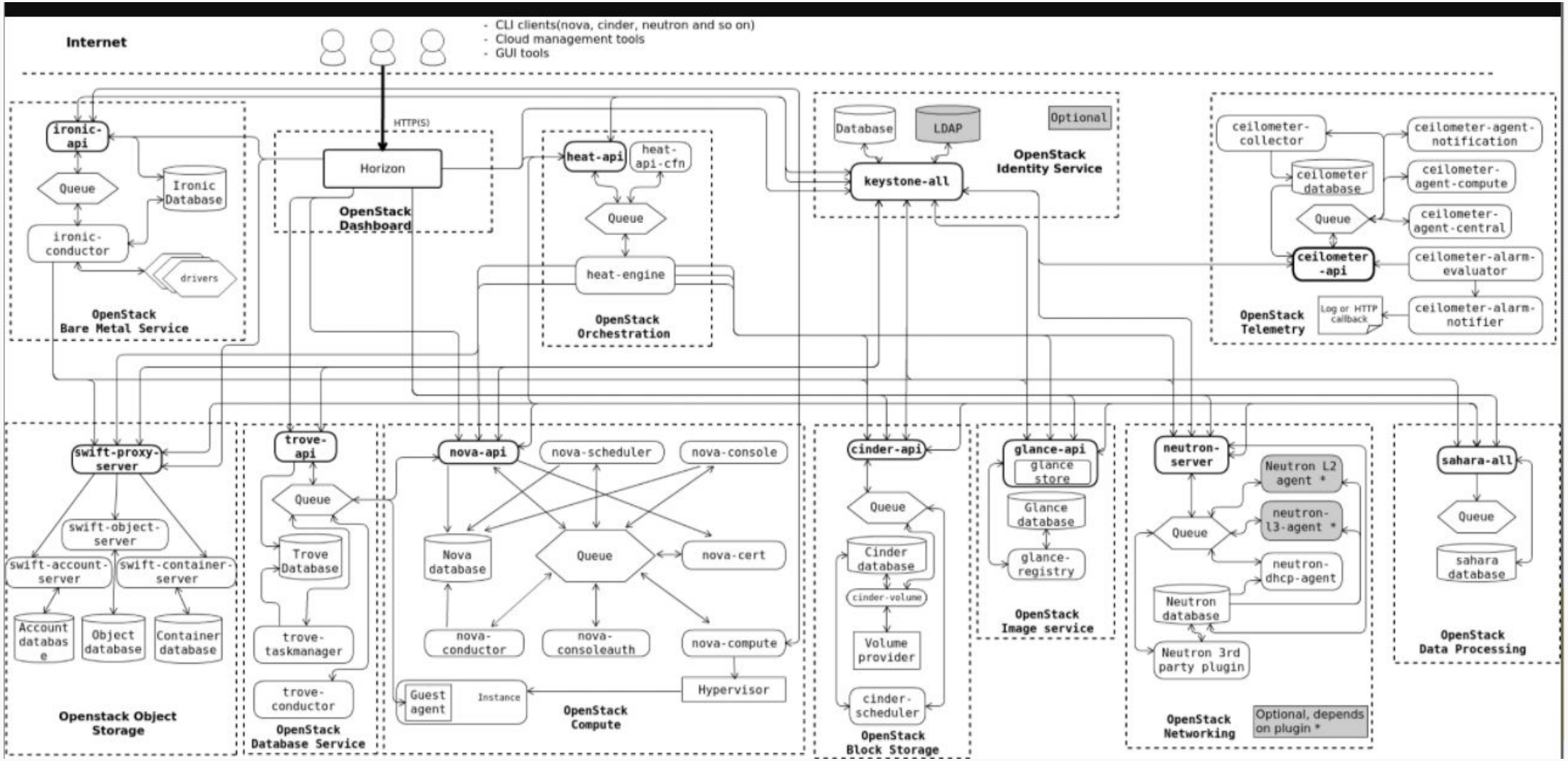
# Openstack - компонентная схема



# Возможности Openstack



# Openstack логическая архитектура



# Openstack Что делают?

- Приватные облака
- Публичные облака
- Виртуализация сетевых функций (network functions virtualization)
- Контейнеризация

# Openstack Что делают?

## What can I do with OpenStack?

### Private clouds

Private cloud distributions run on OpenStack are better than DIY approaches—and there's proof. 451 Research found that it only takes a 6% increase in the number of virtual machines—facilitated by OpenStack's easy installation and management—for OpenStack distributions to become significantly more valuable than private clouds created on your own.

**Read more about private clouds →**

### Network functions virtualization

451 Research found that using OpenStack for network functions virtualization (NFV)—which involves separating a network's key functions so they can be distributed among environments—could very well be the next big thing. It's on the agenda of virtually every global communications services provider surveyed by the analyst.

**Read more about NFV →**

### Public clouds

OpenStack is the leading open source option for building public cloud environments. Whether your company is a multibillion-dollar publicly traded enterprise or a startup, you can use OpenStack to set up public clouds with services that compete with major public cloud providers.

**Read more about public clouds →**

### Containers

OpenStack is a stable foundation for public and private clouds. Containers speed up application delivery while simplifying application deployment and management. Running containers on OpenStack can scale containers' benefits from single, siloed teams to enterprise-wide interdepartmental operations.

**Read more about containers →**

# Основные компоненты

## Compute (Nova)

OpenStack Compute is a cloud computing fabric controller, which manages pools of computer resources and work with virtualization technologies, bare metals, and high-performance computing configurations. Nova's architecture provides flexibility to design the cloud with no proprietary software or hardware requirements and also delivers the ability to integrate the legacy systems and third-party products.

Nova can be deployed using hypervisor technologies such as KVM, VMware, LXC, XenServer, etc. It is used to manage numerous virtual machines and other instances that handle various computing tasks.

## Image Service (Glance)

OpenStack image service offers discovering, registering, and restoring virtual machine images. Glance has client-server architecture and delivers a user REST API, which allows querying of virtual machine image metadata and also retrieval of the actual image. While deploying new virtual machine instances, Glance uses the stored images as templates.

OpenStack Glance supports Raw, VirtualBox (VDI), VMWare (VMDK, OVF), Hyper-V (VHD), and Qemu/KVM (qcow2) virtual machine images.

# Основные компоненты

## Object Storage (Swift)

OpenStack Swift creates redundant, scalable data storage to store petabytes of accessible data. The stored data can be leveraged, retrieved and updated. It has a distributed architecture, providing greater redundancy, scalability, and performance, with no central point of control.

Swift is a profoundly available, shared, eventually consistent object store. It helps organizations to store lots of data safely, cheaply and efficiently. Swift ensures data replication and distribution over various devices, which makes it ideal for cost-effective, scale-out storage.

## Dashboard (Horizon)

Horizon is the authorized implementation of OpenStack's Dashboard, which is the only graphical interface to automate cloud-based resources. To service providers and other commercial vendors, it supports with third party services such as monitoring, billing, and other management tools. Developers can automate tools to manage OpenStack resources using EC2 compatibility API or the native OpenStack API.

## Identity Service (Keystone)

Keystone provides a central list of users, mapped against all the OpenStack services, which they can access. It integrates with existing backend services such as LDAP while acting as a common authentication system across the cloud computing system.

Keystone supports various forms of authentication like standard username & password credentials, AWS-style (Amazon Web Services) logins and token-based systems. Additionally, the catalog provides an endpoint registry with a queryable list of the services deployed in an OpenStack cloud.

# Основные компоненты

## Networking (Neutron)

Neutron provides networking capability like managing networks and IP addresses for OpenStack. It ensures that the network is not a limiting factor in a cloud deployment and offers users with self-service ability over network configurations. OpenStack networking allows users to create their own networks and connect devices and servers to one or more networks. Developers can use SDN technology to support great levels of multi-tenancy and massive scale.

Neutron also offers an extension framework, which supports deploying and managing of other network services such as virtual private networks (VPN), firewalls, load balancing, and intrusion detection system (IDS)

## Block Storage (Cinder)

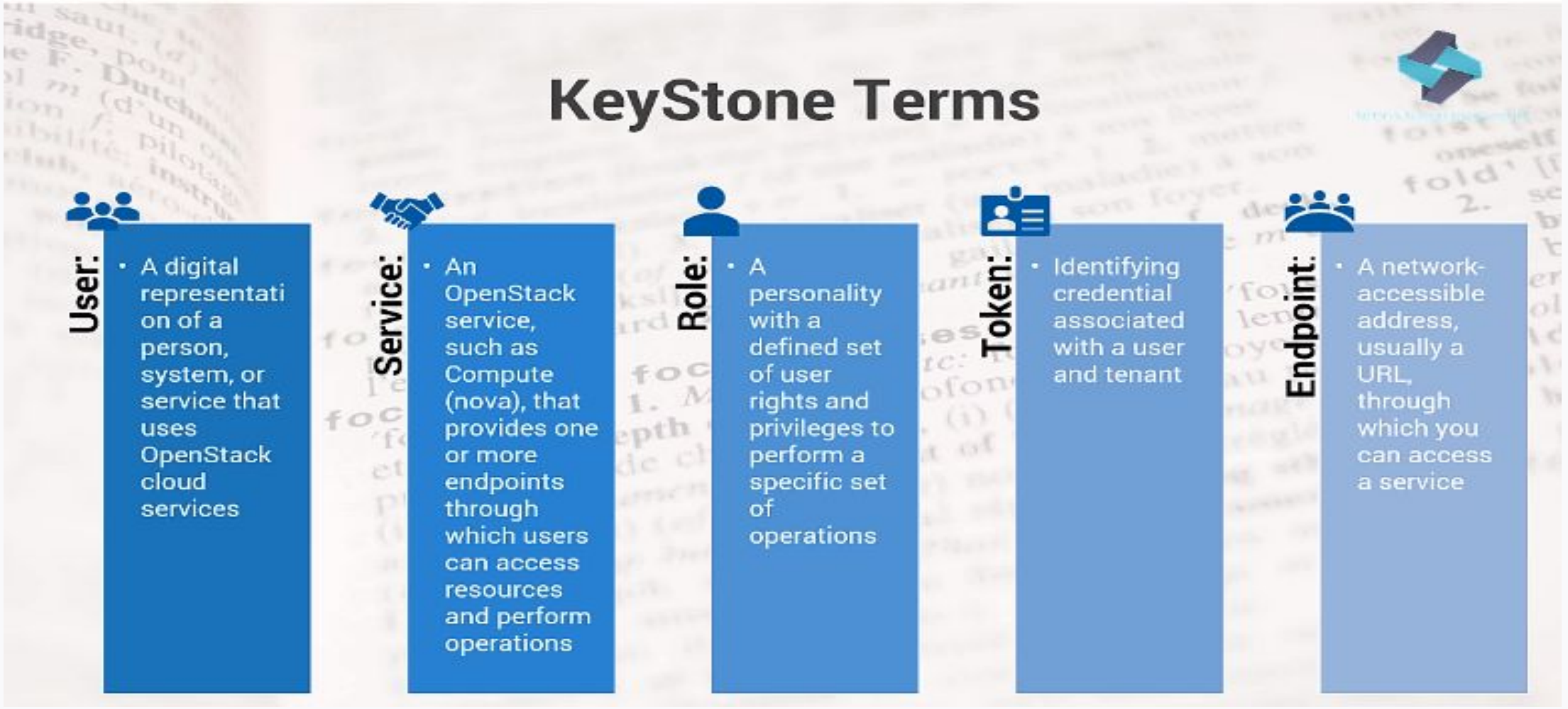
OpenStack Cinder delivers determined block-level storage devices for application with OpenStack compute instances. A cloud user can manage their storage needs by integrating block storage volumes with Dashboard and Nova.

Cinder can use storage platforms such as Linux server, EMC (ScaleIO, VMAX, and VNX), Ceph, Coraid, CloudByte, IBM, Hitachi data systems, SAN volume controller, etc. It is appropriate for expandable file systems and database storage.

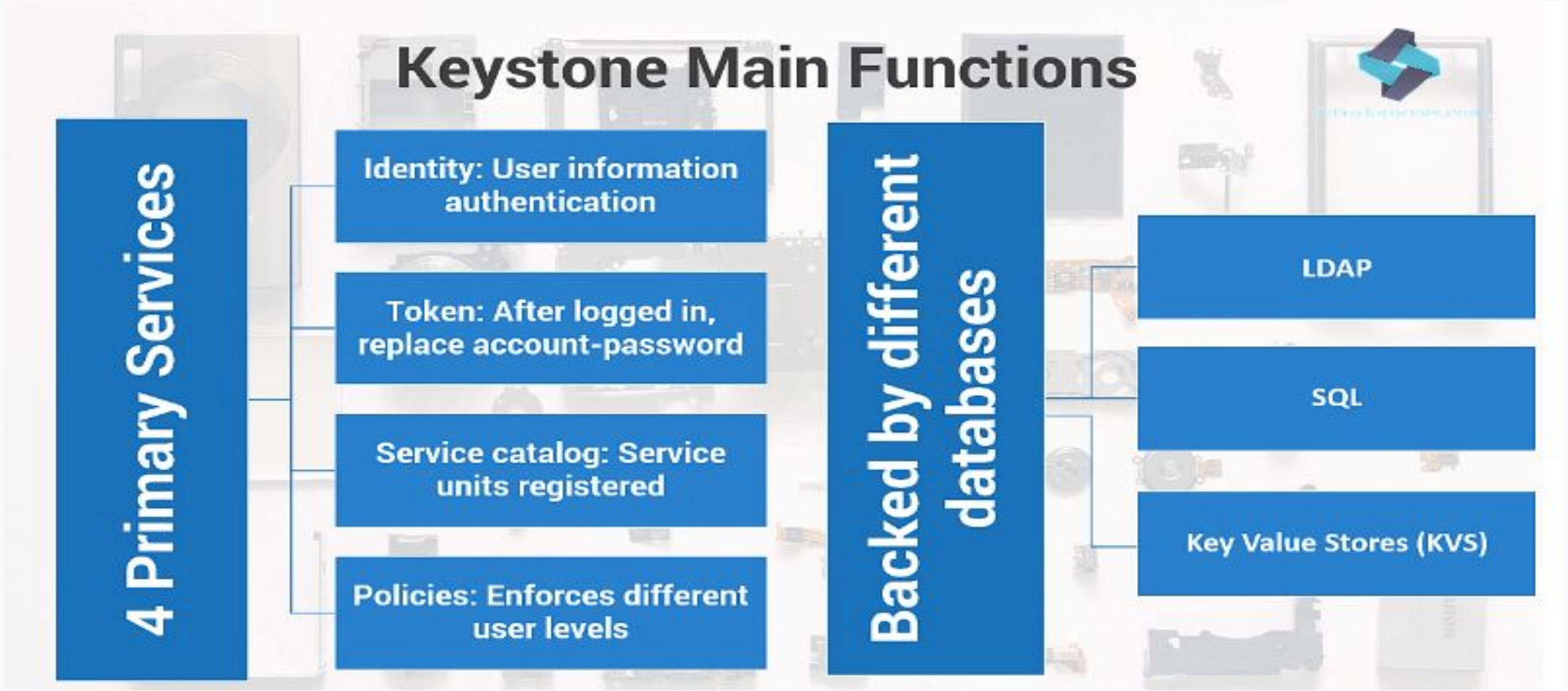

Keystone

# Keystone компонент

# Keystone компонент



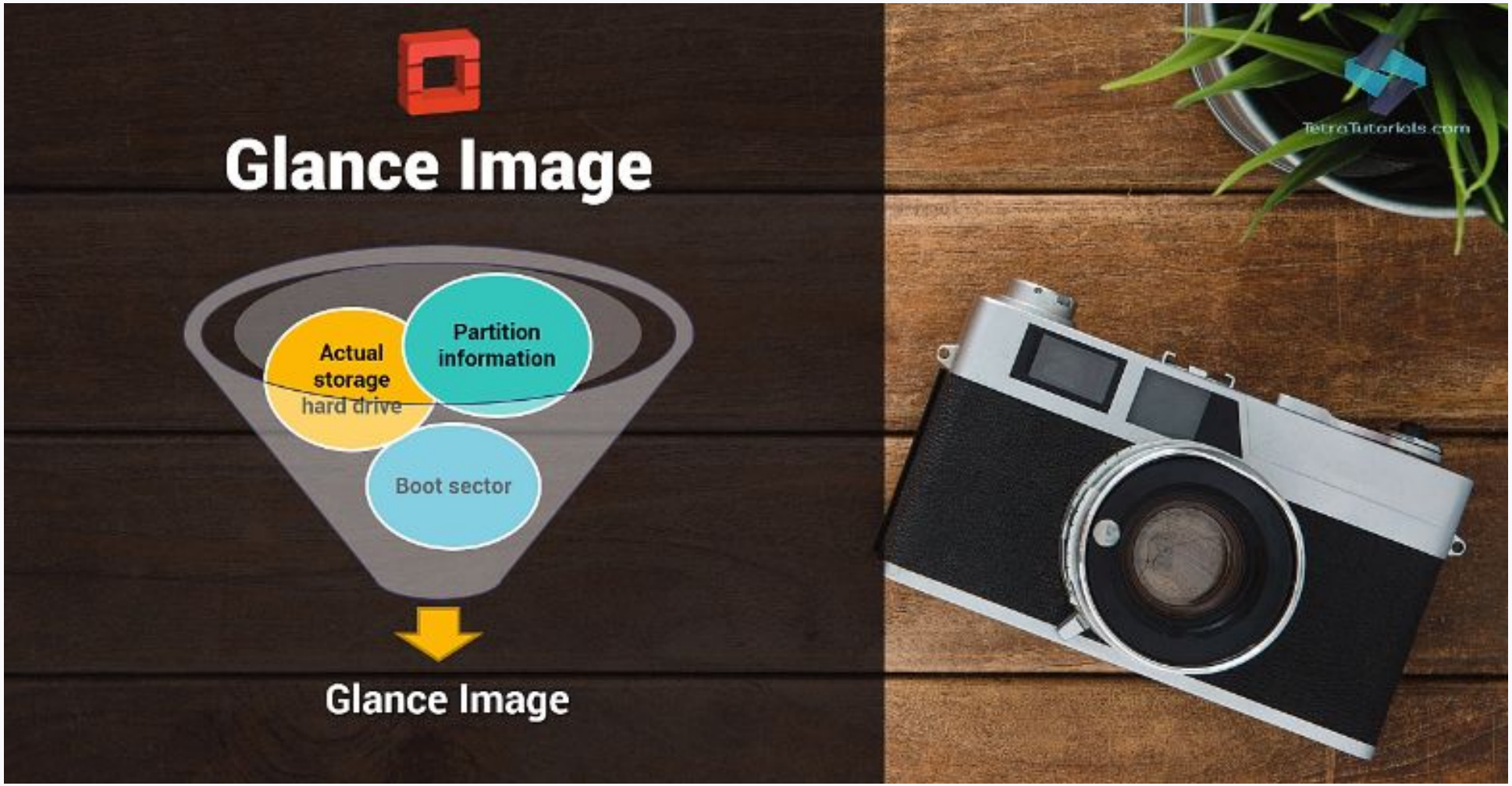
# Keystone компонент



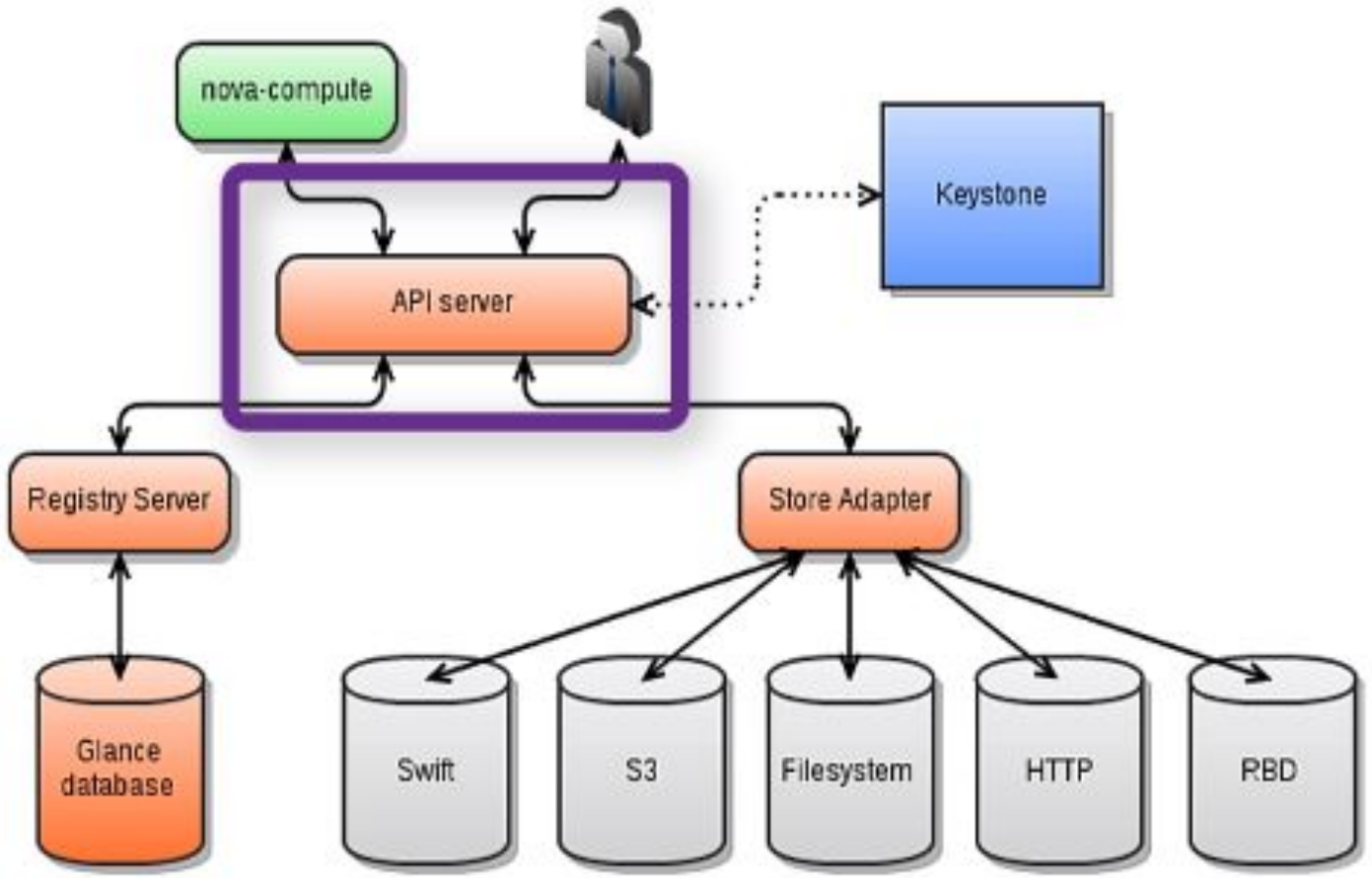

Glance

# Glance: Image management

# Glance: Image management



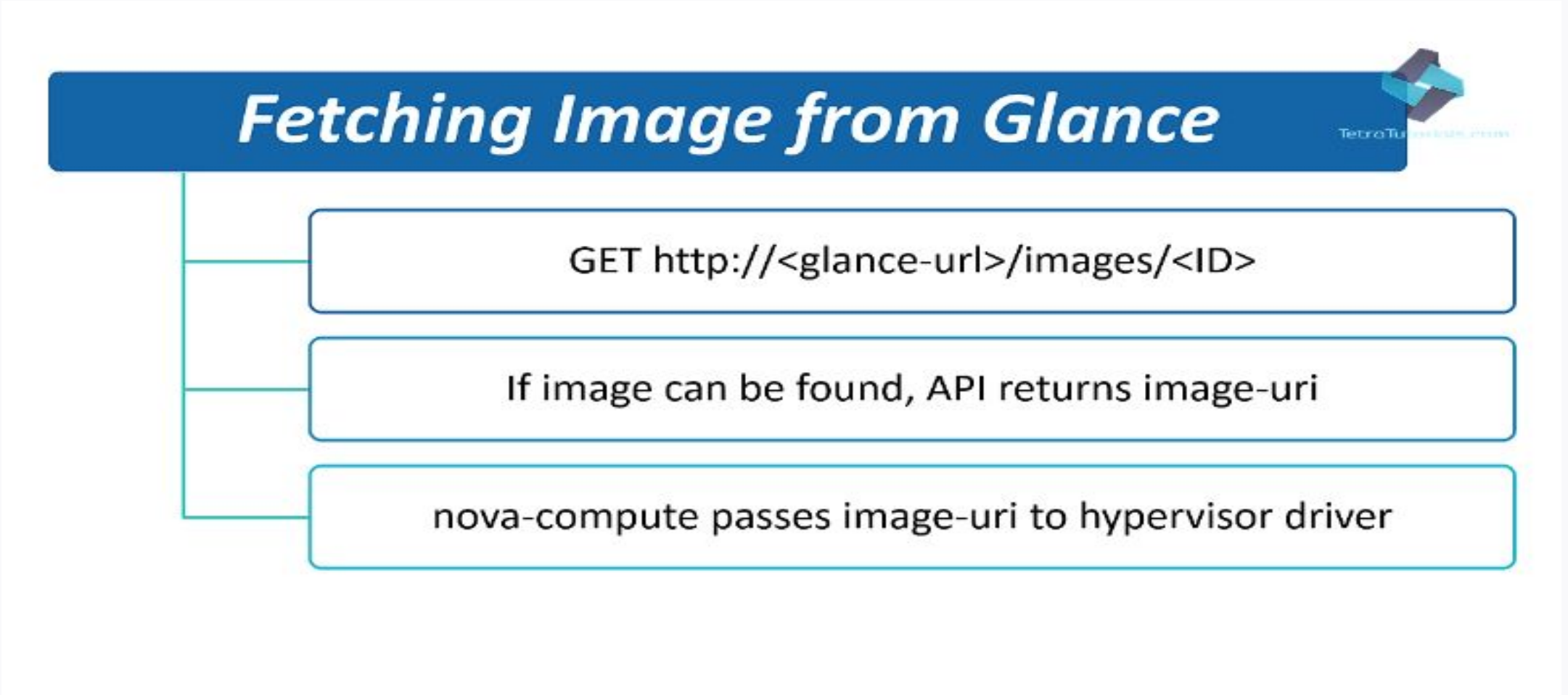
# Glance: Image management



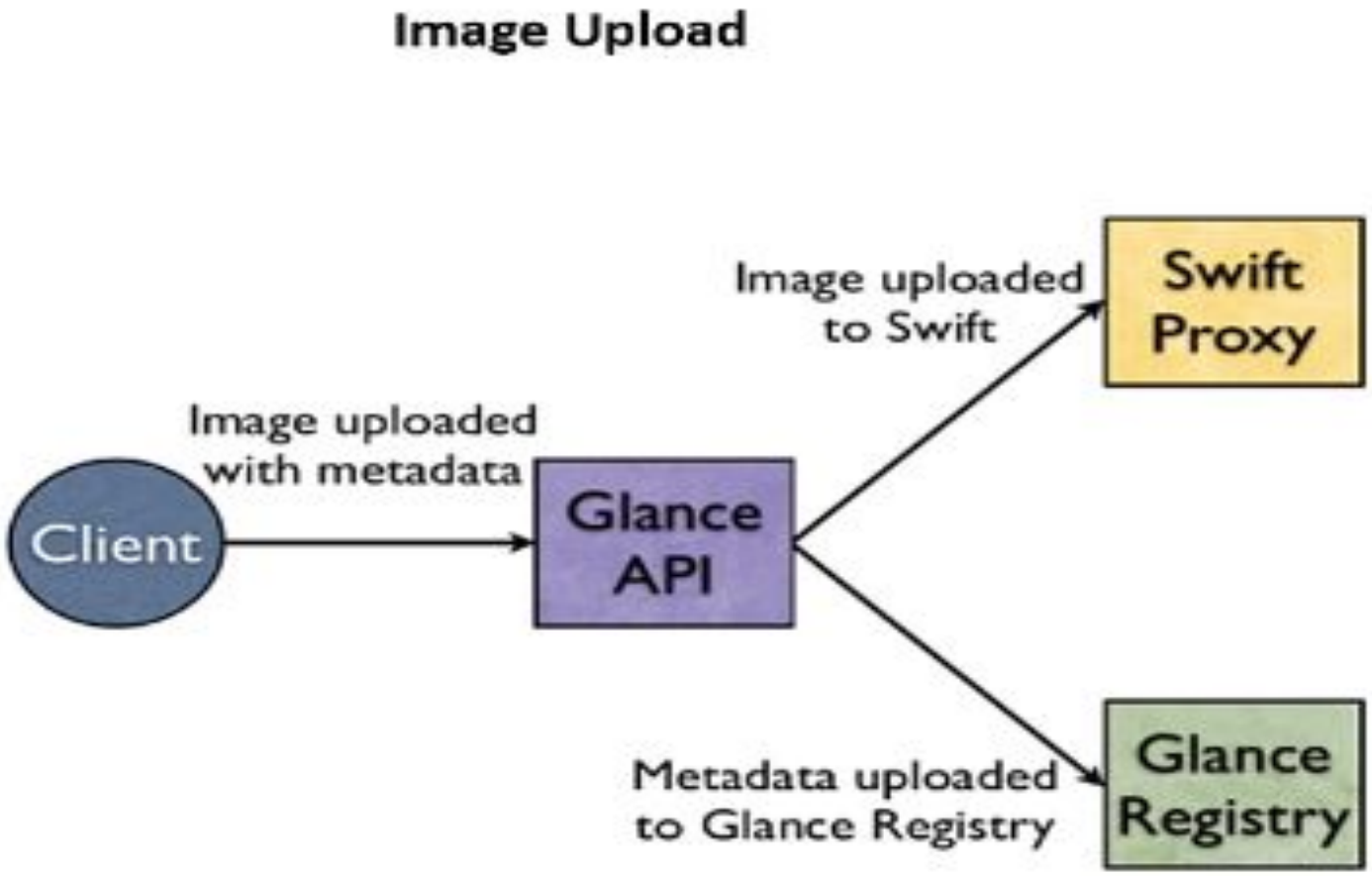
# Glance: Image management

| Disk | Format Description |
|---|---|
| raw | This is an unstructured disk image format |
| vhd | This is the VHD disk format, a common disk format used by virtual machine monitors from VMWare, Xen, Microsoft, VirtualBox, and others |
| vmdk | Another common disk format supported by many common virtual machine monitors |
| vdi | A disk format supported by VirtualBox virtual machine monitor and the QEMU emulator |
| iso | An archive format for the data contents of an optical disc (e.g. CDROM). |
| qcow2 | A disk format supported by the QEMU emulator that can expand dynamically and supports Copy on Write |
| aki | This indicates what is stored in Glance is an Amazon kernel image |
| ari | This indicates what is stored in Glance is an Amazon ramdisk image |
| ami | This indicates what is stored in Glance is an Amazon machine image |

# Glance: Image management



# Glance: Image management

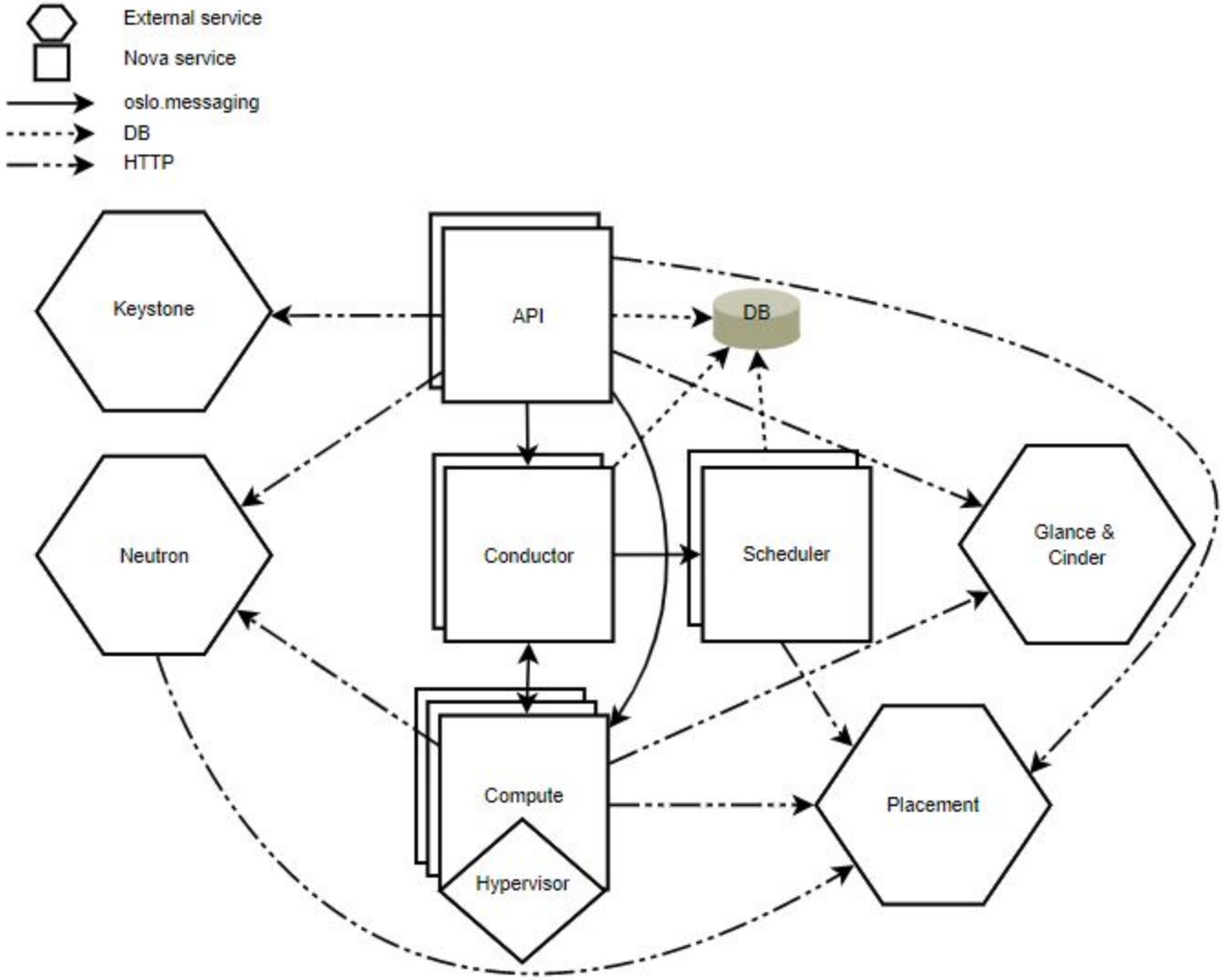


# Glance: Image management

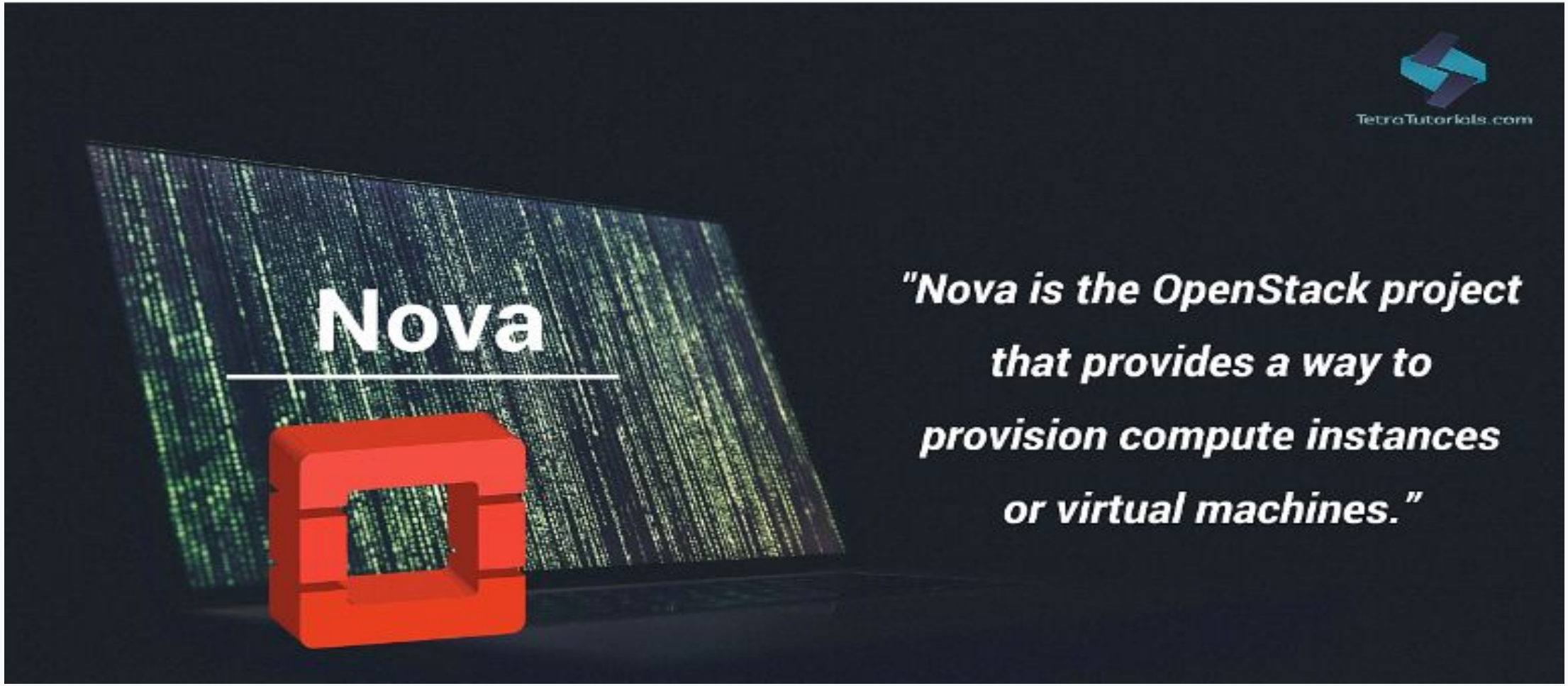
```
[root@controller ~]# ls -lh cirros-0.3.4-x86_64-disk.img
-rw-r--r-- 1 root root 13M May  7  2015 cirros-0.3.4-x86_64-disk.img
[root@controller ~]# . admin-openrc
[root@controller ~]# openstack image create "cirros" --file cirros-0.3.4-x86_64-disk.img --disk-format qcow2 --container-format bar
e --public
+------------------+------------------------------------------------------+
| Field            | Value                                                |
+------------------+------------------------------------------------------+
| checksum         | ee1eca47dc88f4879d8a229cc70a07c6                     |
| container_format | bare                                                 |
| created_at       | 2016-07-05T02:38:43Z                                 |
| disk_format      | qcow2                                                |
| file             | /v2/images/ad7d0aac-e3dc-4e8d-ae3d-1005c49e814e/file |
| id               | ad7d0aac-e3dc-4e8d-ae3d-1005c49e814e                 |
| min_disk         | 0                                                    |
| min_ram          | 0                                                    |
| name             | cirros                                               |
| owner            | caa7a686fd194d6e89d5e304e050e710                     |
| protected        | False                                                |
| schema           | /v2/schemas/image                                    |
| size             | 13287936                                             |
| status           | active                                               |
| tags             |                                                      |
| updated_at       | 2016-07-05T02:38:44Z                                 |
| virtual_size     | None                                                 |
| visibility       | public                                               |
+------------------+------------------------------------------------------+
[root@controller ~]#
```

# Compute: Nova

# Nova: compute service



# Nova: compute service



# Nova: compute service

# Nova: compute service

## nova database

Can be any relational database.

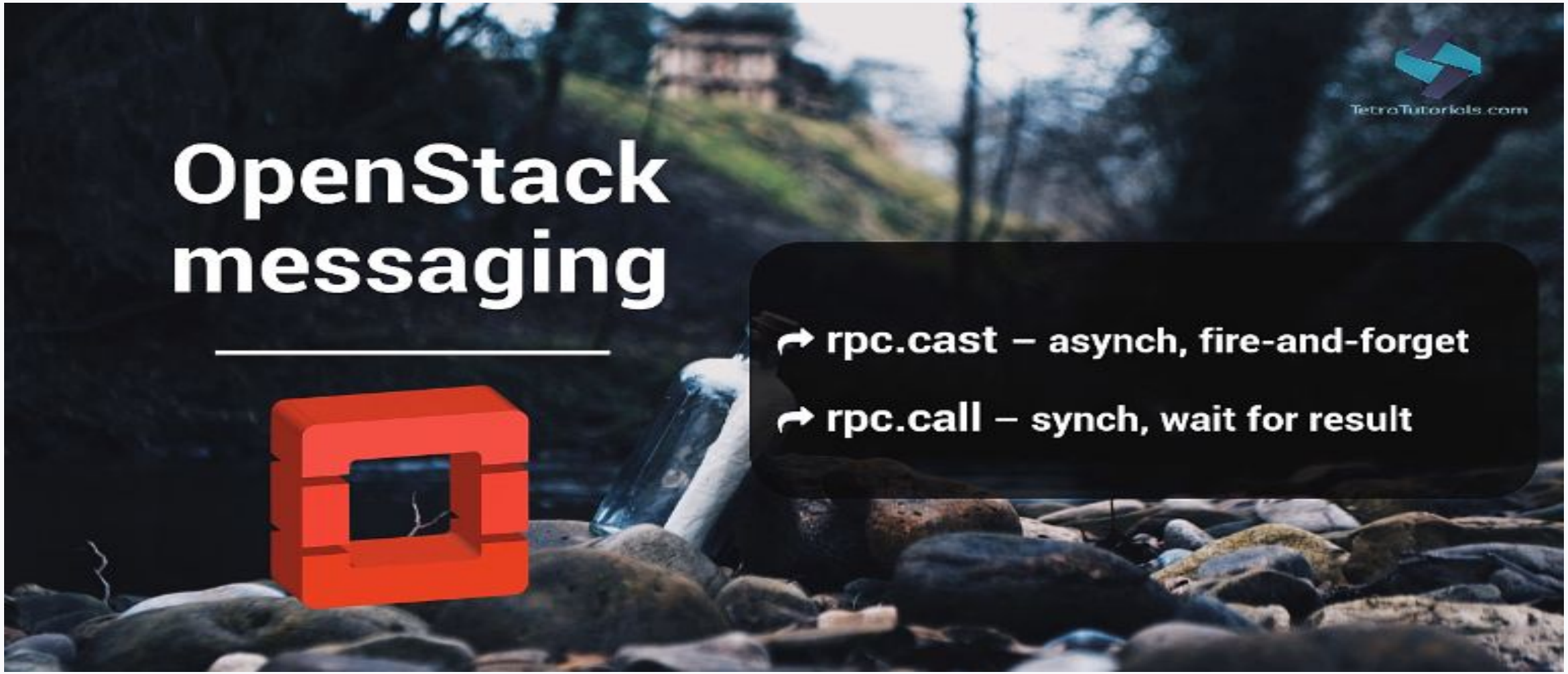

nova-api talks to DB via SQLAlchemy (python ORM(Object Relational Mapper)).


Most of the deployments are done with MySQL or PostreSQL.

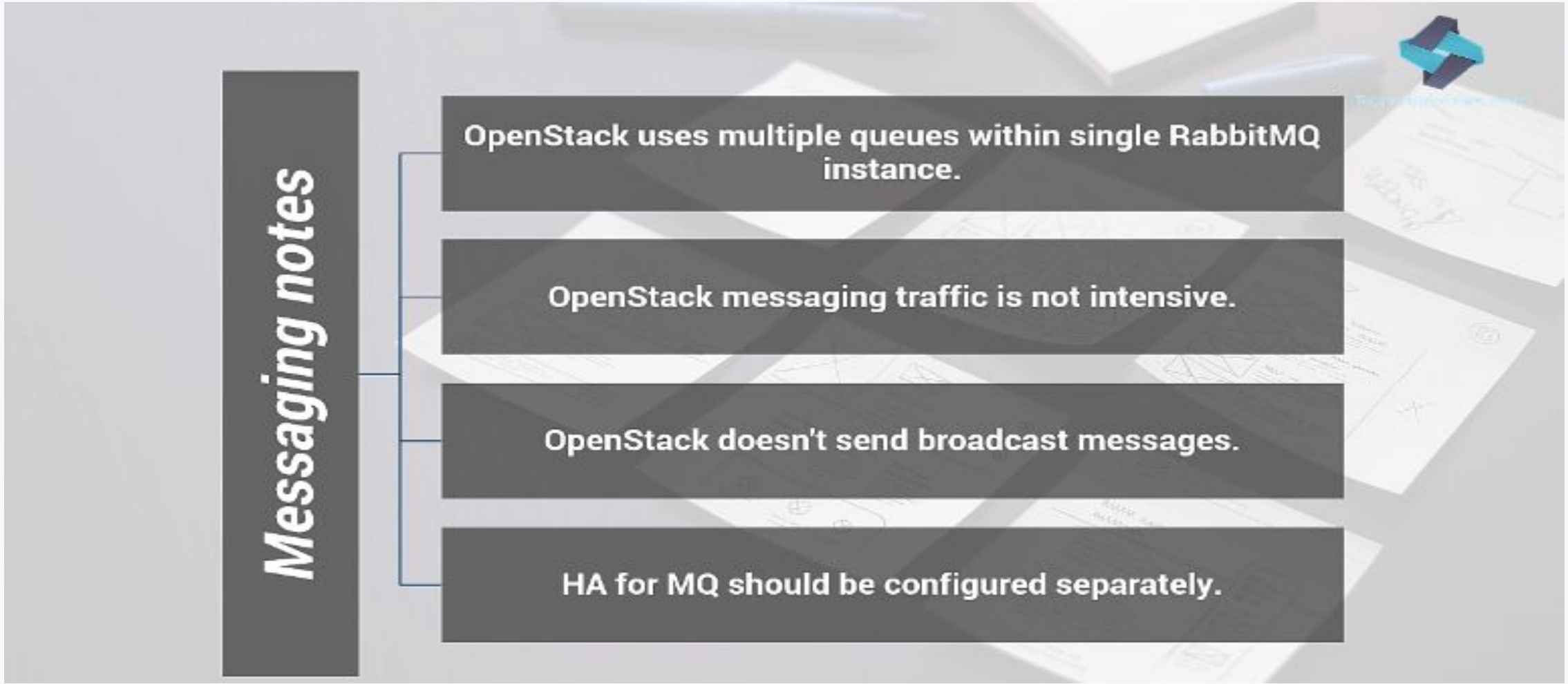
DB HA should be done via 3rd party tools:

- Galera
- Multi-Master replication Model for MySQL (MMM)

# Nova: compute service

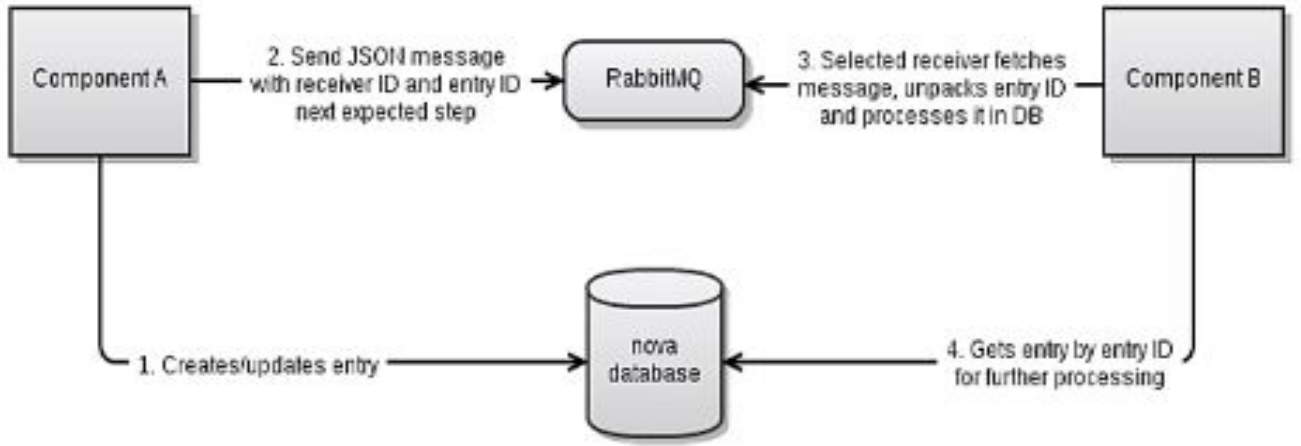


# Nova: compute service



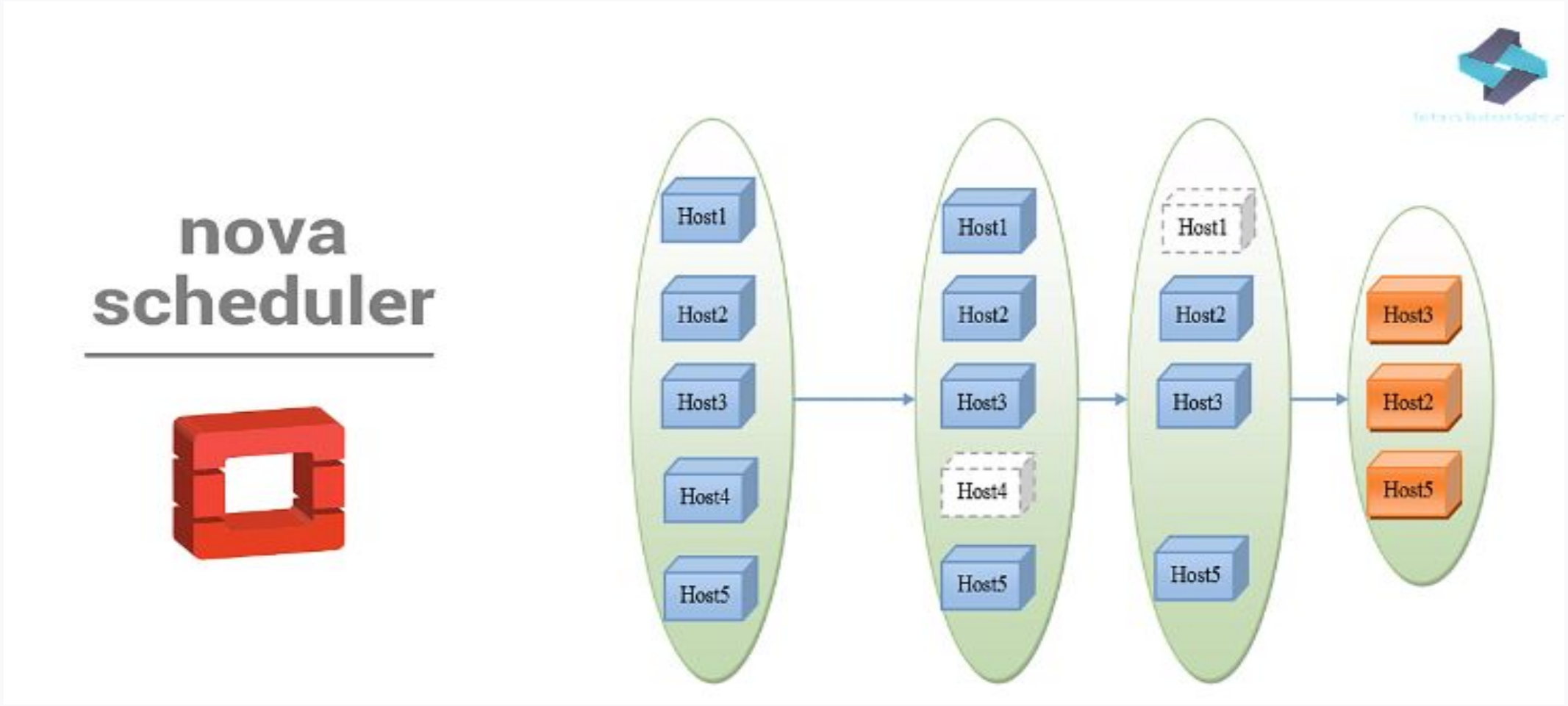
# Nova: compute service

OpenStack messaging

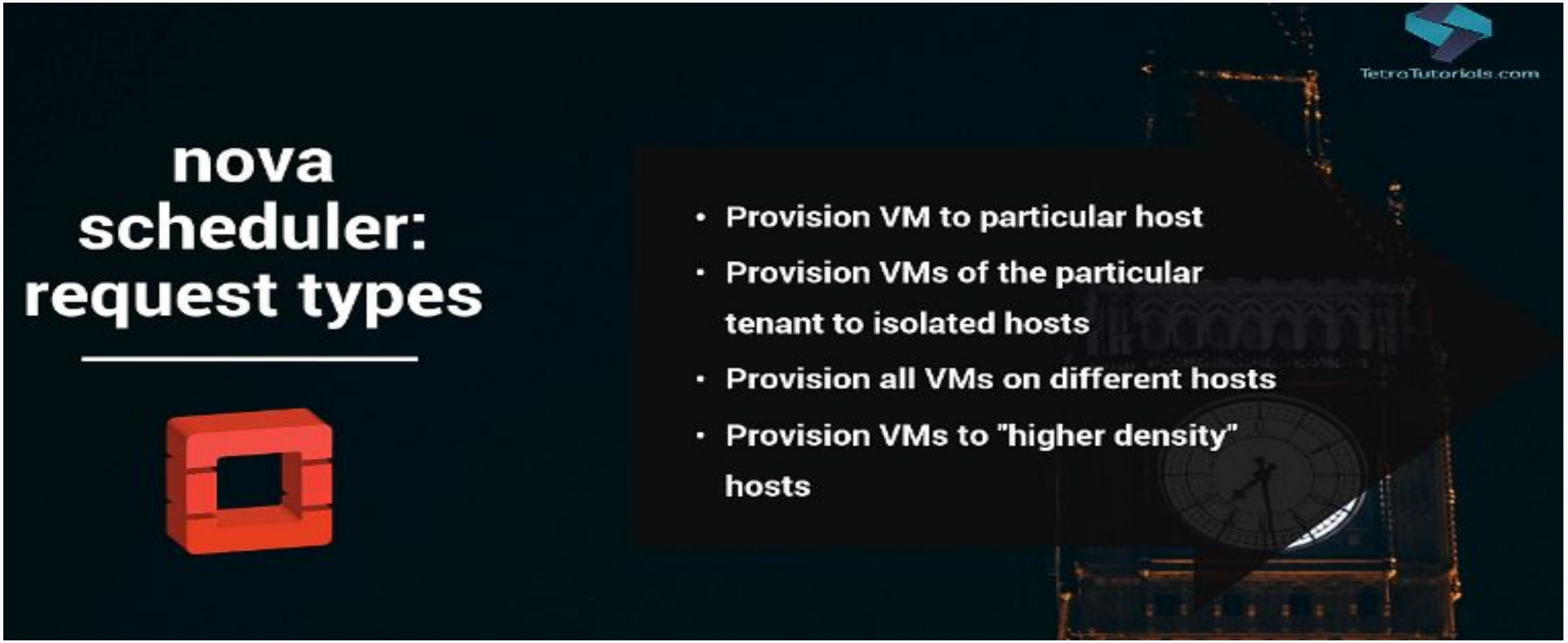


# Nova: compute service

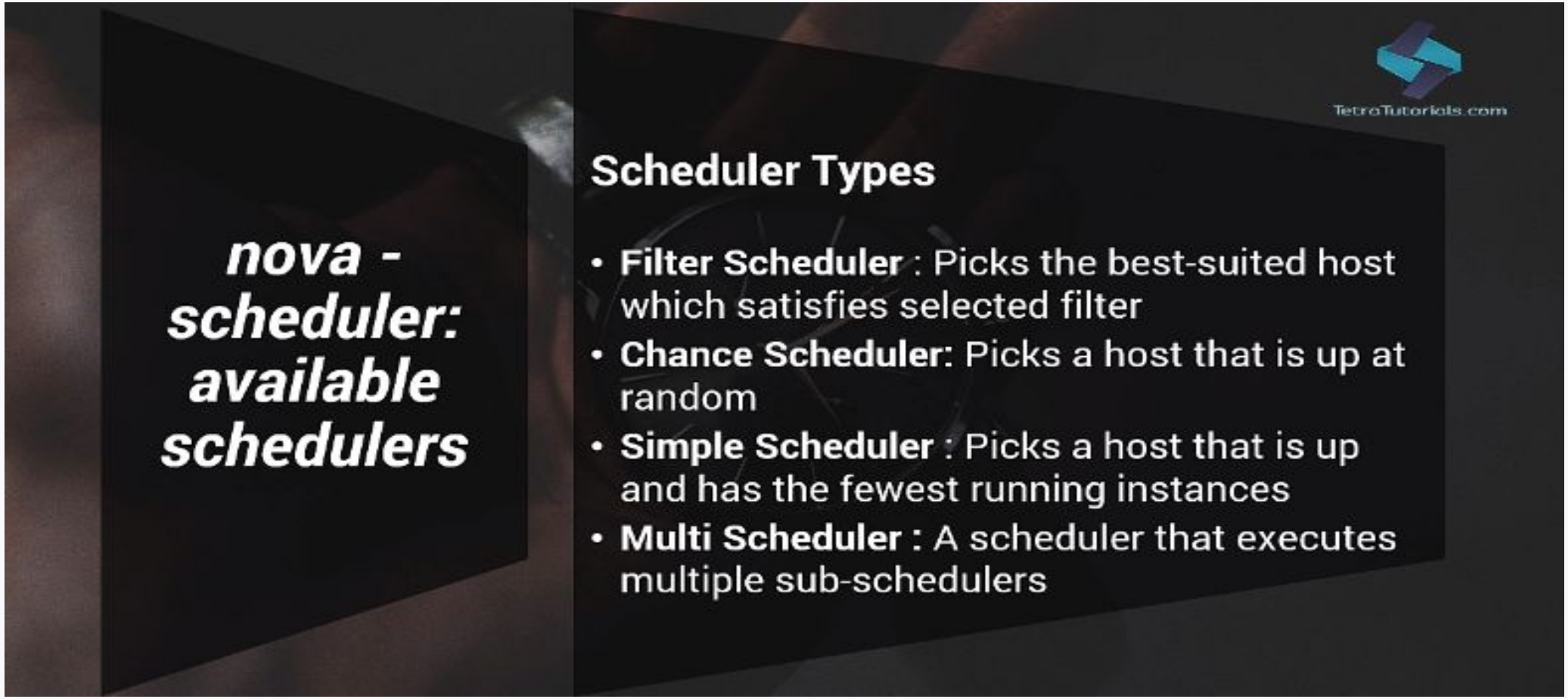
# Neutron: network service



# Openstack installation



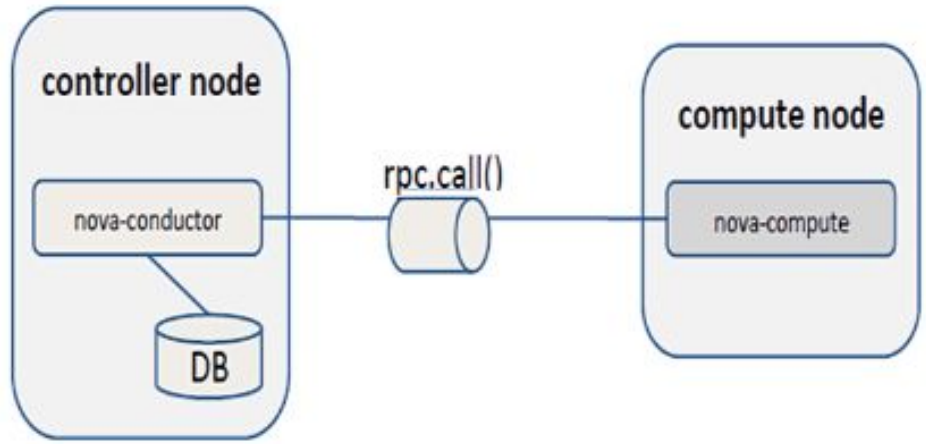
# Openstack installation



# Openstack installation

## Nova Conductor

- *Eliminates the need to connect to Nova remote DB*
- *Horizontal scalability: spawns multiple worker threads operating in parallel*
- *Hides DB implementation/schema from Nova Compute*
- *Possible offloading of long-running operations from other services, not just Nova Compute*
- *Beneficial for operations that cross multiple compute nodes (migration, resizes)*



# Рефлексия

С какими основными мыслями и инсайтами уходите  с вебинара

Каких целей вебинара не удалось достичь

# Полезные ссылки

- https://www.openstack.org/
- https://docs.openstack.org/devstack/latest/
- https://www.stackalytics.com/
- https://opendev.org/

Заполните, пожалуйста,
опрос о занятии по ссылке в чате

# Спасибо за внимание!
# Приходите на следующие вебинары

Шаповалов Евгений
TG: @shapovalovdev