QxStack with
Red Hat OpenStack Platform
- Reference Architecture

June, 2017 Version 1.1

Revision History

Version	Date	Author	Description
1.0	2017.05.18	QCT	Initial Release
1.1	2017.06.19	QCT	Major Release: product name modification

Contents

Tables

Figures

Legal Disclaimer

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH QUANTA CLOUD TECHNOLOGY (QCT) PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN QCT'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, QCT ASSUMES NO LIABILITY WHATSOEVER AND QCT DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF QCT PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY QCT, THE QCT PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE QCT PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Quanta Cloud Technology (QCT) may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." QCT reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

All products, computer systems, dates, and figures specified are preliminary based on current expectations, and are subject to change without notice. Contact your local QCT sales office or your distributor to obtain the latest specifications and before placing your product order.

1. Overview

1.1 Trend

OpenStack is emerging as a front-runner in the world of software-defined infrastructure. Containing a set of open source software components for cloud computing platforms, OpenStack is flexible and agile to fit business demands from small-to-medium businesses to cloud providers in different industries. More and more infrastructure and operations (I&O) leaders consider OpenStack a valuable solution and have become involved in the community over the past few years.

1.2 Purpose of Reference Architecture

This reference architecture provides guidance for cloud services providers and enterprises who are seeking a robust, resilient, and highly available solutions in an OpenStack cloud environment. Quanta Cloud Technology (QCT) and Red Hat have co-engineered a QxStack with Red Hat OpenStack Platform reference architecture based on Red Hat® OpenStack® Platform 10 (OSP). To meet the requirements of enterprises, QxStack with Red Hat OpenStack Platform advocates a streamlined setup process, a "one-click" deployment, for solution deployment. The QxStack Auto-Deployment Tool not only enhances agility but also accelerates the time to value in a private cloud. Moreover, QxStack with Red Hat OpenStack Platform features high failure tolerance to reduce business downtime and data-loss risk. The optimized configuration with high availability, functionality, and performance is validated and proven to be ready for enterprises and cloud providers.

2. Solution Overview

2.1 Solution Hardware

Choosing hardware compatible with OpenStack is complex, time consuming and requires expertise and effort to meet the requirements of users. QCT validated *QuantaPlex* and *QuantaGrid* servers with interoperability, flexibility and reliability on Red Hat OpenStack Platform. *QuantaMesh BMS T1048-LB9* and *QuantaMesh 3000 Series T3048-LY8* switches can also be adopted in diverse network operating systems in this reference architecture. In the following section, the hardware and components will be introduced in detail.

2.1.1 QuantaPlex T41S-2U: Compute and Controller Server

QuantaPlex T41S-2U is an ultra-dense design equipped with four independent nodes for computing. It creates the flexibility to set up different workloads independently in a 2U shared infrastructure, providing a system with sufficient IO expansion for storage and networking. *QuantaPlex T41S-2U* supports Intel® Xeon® processor E5-2600 v3 and v4 product families with enhanced QPI bandwidth to 9.6GT/s. The system performance is optimized to run the most demanding applications.

Table 1. Solution Hardware - QuantaPlex T41S-2U.

QuantaPlex T41S-2U	
 	
Default Configuration	
Form Factor	2U 4 Nodes
Processor	Intel® Xeon® processor E5-2600 v3, v4 product family Max. TDP Support: 145W Internal Interconnect: 6.4 / 8.0/ 9.6 GT/s L3 Cache: Up to 45MB
Storage	24x 2.5” Hot-plug Controller: Intel® C610: 6x SATA 6Gb/s ports SATA RAID 0, 1

Memory	Total Slots: 16 Capacity: Up to 512GB RDIMM / Up to 1024GB LRDIMM Memory type: 2133 / 2400 MHz DDR4 RDIMM / LRDIMM
Network	Option 1: Intel® I350 dual-port 1 GbE per node Dedicated 10/100 management port per node Option 2: Intel® X540 dual-port 10GbE BASE-T per node Dedicated 10/100 management port per node Option 3: Intel® 82599ES dual-port 10G SFP+ per node
Expansion Slot	1x PCIe Gen3 x16 Low profile MD-2 1x PCIe Gen3 x8 mezzanine slot 1x PCIe Gen3 x8 OCP LAN mezzanine slot

2.1.2 QuantaGrid D51PH-1ULH: Storage Server

QuantaGrid D51PH-1ULH features hybrid-tiered storage architecture in an ultra-dense hot-swappable 1U platform. It is a rackmount server tailored for hyper-scale data centers and software-defined storage solutions with Intel® Xeon® processor E5-2600 v3 and v4 product families and up to 1TB memory capacity. The design is ideal for tier storage planning in which the solid-state drives are required to accelerate IOPs and throughput without sacrificing large data storage capacity.

Table 2. Solution Hardware - QuantaGrid D51PH-1ULH.

QuantaGrid D51PH-1ULH	
	
Default Configuration	
Form Factor	1U 1 Node
Processor	Intel® Xeon® processor E5-2600 v3, v4 product family Max. TDP Support: 135W Internal Interconnect: 6.4 / 8.0/ 9.6 GT/s L3 Cache: Up to 45MB
Storage	2.5” Hot-plug / 3.5" Hot-plug Options: 12x 3.5"/2.5" hot-plug 12Gb/s SAS or 6Gb/s SATA HDD/SSD 4x 2.5" hot-plug 7mm 6Gb/s SATA SSD 1x Internal SATA DOM Controller: Intel® 610, 10x SATA 6Gb/s ports; SATA RAID 0, 1, 10 Optional Controller: (For more options, please refer to the CCL)

	Quanta LSI® 3008 12Gb/s SAS mezzanine, RAID 0, 1, 10 Quanta LSI® 3108 12Gb/s RAID mezzanine, RAID 0, 1, 5, 10. RAID 6 with additional RAID key
Memory	Total Slots: 16 Capacity: Up to 512GB RDIMM Memory type: 2400/2133/1866/1600/1333 MHz DDR4 RDIMM
Network	Intel® I350 dual-port 1 GbE Dedicated 10/100/1000 management port Optional NIC: Quanta Intel® I350 dual-port OCP mezzanine Quanta Intel® X540 dual-port 10GbE BASE-T OCP mezzanine Quanta Intel® 82599ES dual-port 10G SFP+ OCP mezzanine Quanta Mellanox CX3PRO dual-port 10G SFP+ OCP mezzanine (more options refer to the CCL)
Expansion Slot	1x SAS Mezzanine x8 1x OCP LAN Mezzanine slot x8

2.1.3 QuantaMesh BMS T1048-LB9: Management Switch

By leveraging merchant silicon chips, *QuantaMesh T1048-LB9* is a high-performance and high-density Ethernet switch with an affordable price for the deployment of data center infrastructure. With ONIE (Open Network Installation Environment) pre-loaded, *QuantaMesh T1048-LB9* provides flexibility to install different types of network operating systems. This enables an agile installation process and fast response for changing business demands.

Table 3. Solution Hardware - QuantaMesh T1048-LB9.

QuantaMesh T1048-LB9	
	
Default Configuration	
Physical ports	Port configuration: - 48x 100/100/1000BAE-T - 4x 1/10Gbe SFP+ ports Management Port: Out-of-band management port (RJ-45, 10/100/1000Base-T) Console Port: 1 (RJ-45)
Performance	Switch capacity: 176Gbps Maximum forwarding rate: 131Mpps MAC: 32K
High Availability	Redundant power supply: 1+1 Hot-swappable fan tray: 4 fixed fans

2.1.4 QuantaMesh T3048-LY8: Top-of-Rack (ToR) Switch

QuantaMesh T3048-LY8 is a high-performance and low-latency layer 2/3/4 Ethernet switch with 48 SFP+ ports and up to 6 QSFP+ ports in a 1U form factor. Each 40G QSFP+ port can be independently configured as 40GbE or four 10GbE ports. Built for Infrastructure-as-a-service (IaaS) data center deployment, high performance computing clusters, and financial applications, the *QuantaMesh T3048-LY8* with its high port density and high performance as well as its ultra-low latency characteristic is ideal for demanding workloads and provides optimal total cost of ownership (TCO).

Table 4. Solution Hardware - QuantaMesh T3048-LY8.

QuantaMesh T3048-LY8	
	
Default Configuration	
Physical ports	Port configuration: - 48x 1/10GbE SFP+ - 6x QSFP+ ports Management Port: Out-of-band management port (RJ-45, 10/100/1000Base-T) Console Port: 1 (RJ-45)
Performance	Switching capacity: 1440Gbps Maximum forwarding rate: 1071Mpps Latency: <600ns
CPU Board	CPU: Freescale P2020 Memory: 2GB DDR3/ECC Flash: 128MB Storage: 8GB Micro SD
SDN	OpEN API: RESTful API
High Availability	Redundant power supply: 1+1 Hot-swappable fan tray: N+1
Virtualization	VXLAN: Yes
Data Center Features	Enhanced Transmission Selection (802.1Qaz) Priority-based Flow Control (802.1Qbb) DCBX: DCBX for ETS /DCBX for PFC FCoE Initiation Protocol (FIP) snooping

2.1.5 QCT System Manager (QSM)

To eliminate the complexity of infrastructure management, QCT System Manager (QSM) is a remote management console for hardware monitoring and maintenance. It is agentless, and its manageability converges across computing servers (storage server) and networking. This design improves working efficiency by converging server management and networking devices. The benefits of QSM are listed below:

- **Easy-to-use Interface:** A customizable dashboard for system hardware maintenance.
- **Real-Time Monitoring:** Real-time display and alerts for server health status.
- **One-Click Data Collection:** Converged IT assets management for further analysis and statistics.
- **Hyper-scale Management:** Batch update and retrieve for hardware configurations at data center level, which highly reduces maintenance time of the system.

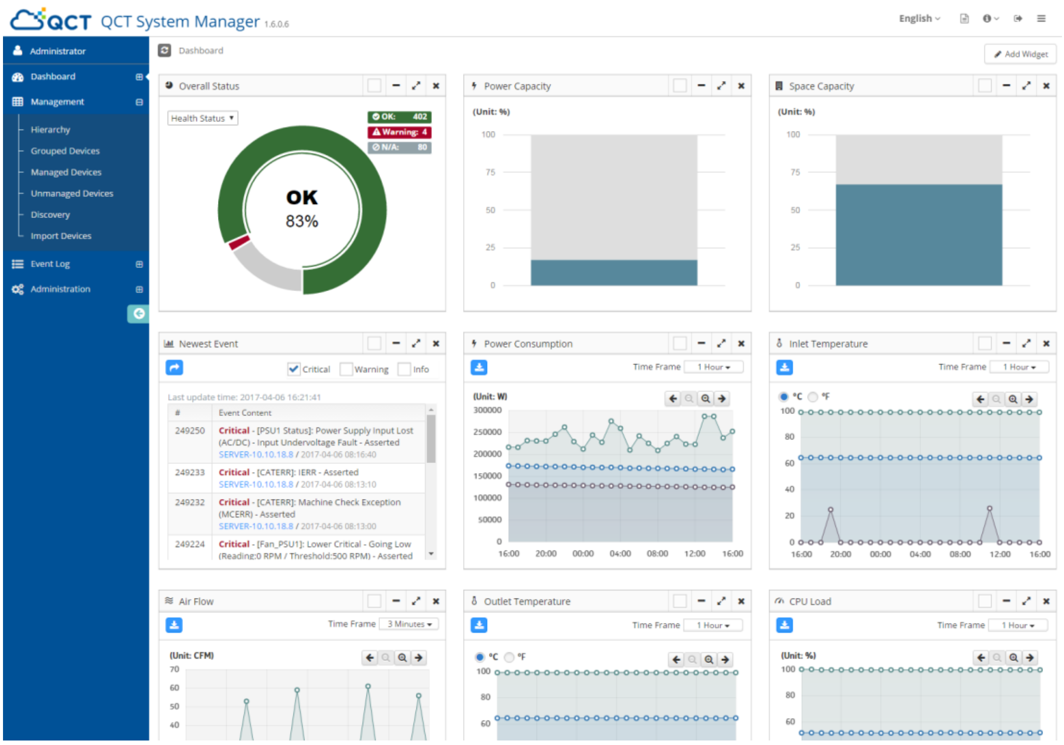


Figure 1. QCT System Manager (QSM) Dashboard Screenshot.

2.2 Solution Software

In QxStack with Red Hat OpenStack Platform, the BIOS and firmware are upgraded to the latest version for hardware management, and the switch software is provided with the flexibility for administrators to adopt in the existing network environment.

Different types of networking operating systems supporting VLAN, MC-LAG (Multi-Chassis Link Aggregation Group) and LACP Bypass can be seamlessly integrated in this reference architecture. The detailed software information regarding QxStack with Red Hat OpenStack Platform is listed below:

Table 5. Solution Software Version.

Hardware Management	
QuantaPlex T41S-2U	BIOS Version: S2S_3A19 Firmware Version: 3.44.00
QuantaPlex D51PH-1ULH	BIOS Version: S2P_3A16 Firmware Version: 3.16.00
QuantaMesh T3048-LY8	Operating System: Cumulus Linux 3.1.1
QCT System Manager (QSM)	Version 1.5 Browser Recommend - Mozilla Firefox 31 or above - Google Chrome 36 or above - Microsoft Internet Explorer 10 or above
Operating System and Solution Software	
Red Hat Enterprise Linux®	Version 7.3
Red Hat OpenStack Platform	Version 10
Red Hat OpenStack Platform director	Version 10
Red Hat Ceph Storage	Version 2.0

2.3 Red Hat OpenStack Platform

Red Hat OpenStack Platform is a production-ready foundation to help you create, deploy, scale, and manage a secure and reliable public or private OpenStack cloud. If your organization is considering an OpenStack deployment, it matters which OpenStack distribution and which Linux offering you choose. This is why Red Hat — recognizing the critical interdependencies between the two — co-engineered them together with closely aligned product engineering teams. Paired with Red Hat CloudForms' open, hybrid management for workloads and infrastructure as well as Red Hat Ceph Storage's highly redundant, software-defined storage, Red Hat OpenStack Platform is an integrated, optimized, and managed foundation for production-ready clouds.

Red Hat OpenStack Platform paired with QCT QxStack systems can be turned into a private, public, or hybrid cloud platform that includes:

- Authentication and authorization mechanisms
- Fully distributed object storage
- Integrated networking
- Persistent block-level storage
- Virtual machine provisioning engine and image storage
- Web browser-based interface accessible to users and administrators

2.3.1 Benefits of Red Hat OpenStack Platform

- **Reliable deployments with live upgrades**: Red Hat OpenStack Platform director checks systems throughout the installation process to provide consistent and automated cloud deployment. It features live orchestrated system upgrades and updates, ensuring long-term, production-ready stability with little downtime.

- **Hardened for enterprise**: An extensive patching, bug fixing, testing, and certification process ensures broad compatibility and performance with upstream community releases.

- **Reliable storage**: Every Red Hat OpenStack Platform subscription includes 64TB of Red Hat Ceph Storage for users to get started with highly scalable and redundant object, block, and file storage.[1]

- **Highly available infrastructure**: Red Hat OpenStack Platform maintains high availability and policy-driven measures, including infrastructure failure recognition, automated host node evacuation, downed node fencing, and automatic restarts of workloads on remaining available hosts.

- **Enterprise software life cycle**: Red Hat provides stable branch releases of OpenStack and Linux that are supported for an enterprise production life cycle—beyond the six-month release cycle of the OpenStack community. Customers can choose to standardize for up to five years on certain releases or stay on top of the latest features by updating every six months to one year.

- **World-class global support, professional consulting services, and certified training**: Red Hat provides global support services to help customers running critical infrastructure like Red Hat OpenStack Platform and Red Hat Ceph Storage. Customers with an active subscription can contact Red Hat support engineers via telephone, e-mail, and an available web portal. Additionally, Red Hat also has dedicated consulting services and a Technical Account Manager (TAM) team available to work closely with customers. To ensure customers are

[1] Additional capacity of Red Hat Ceph storage sold separately.

prepared to operate the system, Red Hat develops end-user training and certification programs.

- **Technology leadership**: Red Hat is a top code contributor to many OpenStack projects and a long-time leader in the OpenStack, Ceph storage, and broader Linux communities—making Red Hat an ideal supporter of full-scale OpenStack deployments.
- **Expansive ecosystem**: Red Hat has built the world's largest certified OpenStack partner ecosystem for Red Hat OpenStack Platform. It includes thousands of certified servers and third-party software, plus an OpenStack-specific certification program with partners in the compute, storage, networking, ISV software, and service fields.
- **Security**: SELinux military-grade security technologies prevent intrusions and protect data when running in public or private OpenStack clouds.
- **Performance**: The Red Hat Virtualization Hypervisor provides superior performance for OpenStack workloads. Based on Kernel-based Virtual Machine (KVM), the hypervisor holds record-breaking performance scores on the SPECvirt_sc2013 benchmark.[2]
- **Integrated stack**: Red Hat OpenStack Platform helps relieve deployment and management burdens by integrating with a portfolio of Red Hat cloud infrastructure products.
    - Deploy Red Hat Enterprise Linux as host nodes and virtual machines to gain performance, security, and operational advantages.
    - Use Red Hat CloudForms as a unified infrastructure and to manage virtual workloads on Red Hat OpenStack Platform.
    - Utilize Red Hat Ceph Storage for highly redundant, scale-out block, object, and image storage.
    - Use Red Hat Virtualization for traditional scale-up virtualization alongside new OpenStack scale-out workloads.
    - Deploy Red Hat OpenShift Container Platform to build a DevOps Platform-as-a-Service (PaaS) with containers.
    - Access Red Hat Satellite for application and operating system entitlement, including images and host package management displayed by the director

[2] All comparisons are based on a benchmark addressing performance evaluation of data center servers used in virtualized server consolidation at www.spec.org/virt_sc2013/ as of March 10, 2016. SPEC® and the benchmark name SPECvirt_sc® are registered trademarks of the Standard Performance Evaluation Corporation (SPEC).

in Red Hat OpenStack Platform.

2.3.2 Red Hat OpenStack Platform Components

The Red Hat OpenStack Platform IaaS cloud is implemented as a collection of interacting services that control compute, storage, and networking resources. The cloud can be managed with a web-based dashboard or command-line clients, which allow administrators to control, provision, and automate OpenStack resources. OpenStack also has an extensive API, which is also available to all cloud users. The following diagram provides a high-level overview of the OpenStack core services and their relationship with each other.

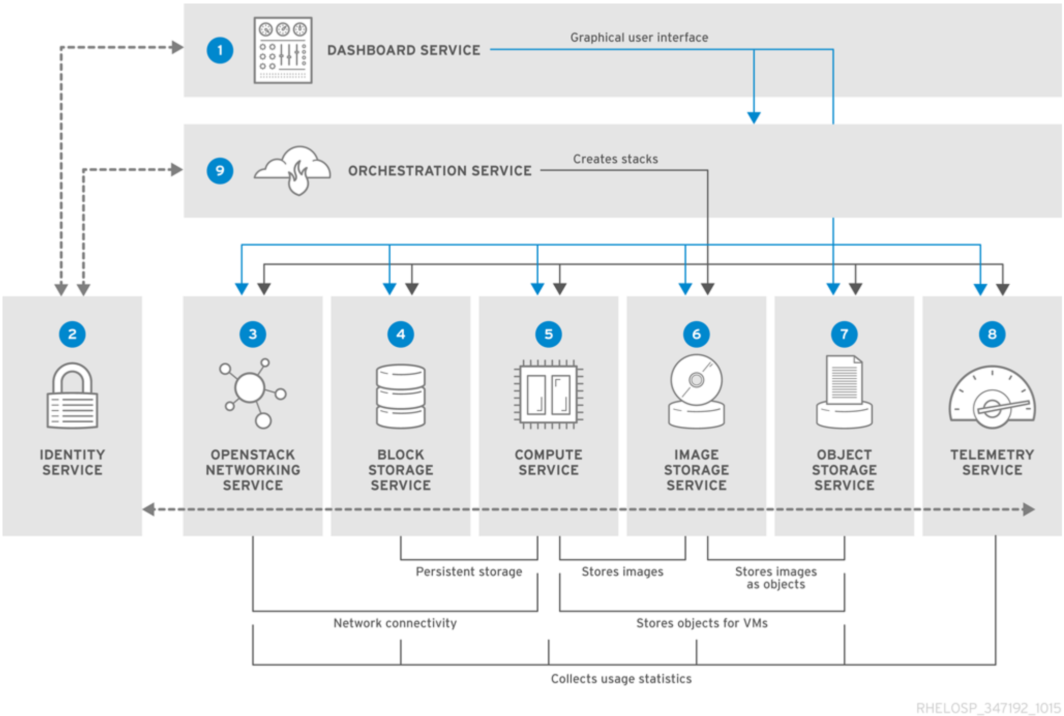


Figure 2. Overview of the OpenStack Core Services.

Red Hat OpenStack Platform provides the software necessary to build a private or public cloud on top of Red Hat Enterprise Linux. The list of the OpenStack projects supported in Red Hat OpenStack Platform 10 is listed below:

- OpenStack Bare Metal (ironic)
- OpenStack Block Storage (cinder)
- OpenStack Compute (nova)
- OpenStack Dashboard (horizon)
- OpenStack Data Processing (sahara)
- OpenStack Identity (keystone)
- OpenStack Image Service (glance)

- OpenStack Networking (neutron)
- OpenStack Object Storage (swift)
- OpenStack Orchestration (heat)
- OpenStack Shared File Systems (manila)
- OpenStack Telemetry (ceilometer)
- OpenStack Telemetry Metrics (gnocchi)
- OpenStack Telemetry Alarming (aodh)

It also includes the following technology previews[3]:

- OpenStack Benchmarking (rally)
- OpenStack DNS-as-a-Service (designate)
- OpenDaylight SDN Platform
- OpenStack Networking (neutron): VLAN aware VMs and OVS firewall driver
- CephFS Manila driver
- Real Time KVM

2.3.3 Red Hat OpenStack Platform Director

The Red Hat OpenStack Platform director is a toolset for installing and managing a complete OpenStack environment. It is based primarily on the OpenStack project TripleO[4], which is an abbreviation for "OpenStack-On-OpenStack." This project takes advantage of OpenStack components to install a fully operational OpenStack environment. This includes new OpenStack components that provision and control bare metal systems to use as OpenStack nodes. This provides a simple method for installing a complete Red Hat OpenStack Platform environment that is lean, robust, and easily repeatable.

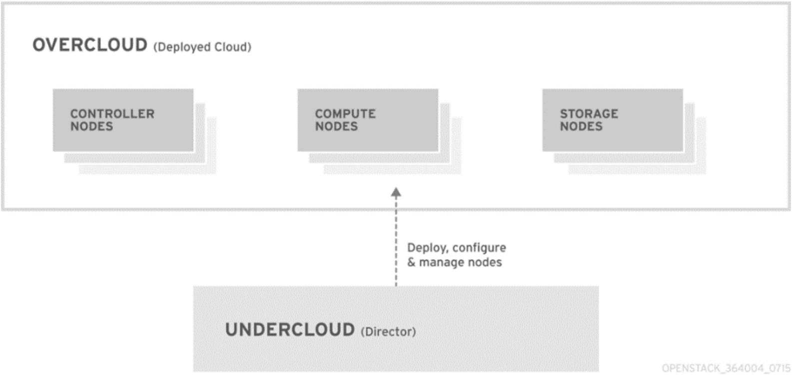


Figure 3. Basic Layout of Undercloud and Overcloud.

---

[3] The projects offered as Technology Preview are not fully supported by Red Hat and should only be used for non-production environments.

[4] TripleO documentation: http://tripleo.org/

Red Hat OpenStack Platform director uses two main concepts: an undercloud and an overcloud. The undercloud installs and configures the overcloud. The next few sections outline the concept of each.

Undercloud

The undercloud is the main director node. It is a single-system OpenStack installation that includes components for provisioning and managing the OpenStack nodes that comprise your OpenStack environment (the overcloud). The components that form the undercloud provide the following functions:

- **Environment planning** - The undercloud provides planning functions for users to assign Red Hat OpenStack Platform roles, including Compute, Controller, and various storage roles.
- **Bare metal system control** - The undercloud uses various configurable drivers, including Intelligent Platform Management Interface (IPMI) for power management control and a PXE-based service for provisioning bare metal systems as OpenStack nodes.
- **Orchestration** - The undercloud provides and reads a set of YAML templates to create an OpenStack environment.

Red Hat OpenStack Platform director utilizes these undercloud functions through both a web-based graphical user interface and a terminal-based command line interface. The undercloud uses the following components:

- **OpenStack Identity (keystone)** - Provides authentication and authorization for the director's components.
- **OpenStack Bare Metal (ironic) and OpenStack Compute (nova)** - Manages bare metal nodes.
- **OpenStack Networking (neutron) and Open vSwitch** - Controls networking for bare metal nodes.
- **OpenStack Image Service (glance)** - Stores images that are written to bare metal machines.
- **OpenStack Object Storage (swift)** - Provides object storage for the node's information.
- **OpenStack Orchestration (heat) and Puppet** - Provides orchestration of nodes and configuration of nodes after the director writes the overcloud image to disk.
- **OpenStack Telemetry (ceilometer)** - Performs monitoring and data collection. The telemetry also includes OpenStack Telemetry Metrics (gnocchi) and OpenStack Telemetry Alarming (aodh), which respectively

provides a time series database for metrics and an alarming component for monitoring.

- **OpenStack Workflow Service (mistral)** - Provides a set of workflows for certain director-specific actions, such as importing and deploying plans.
- **OpenStack Messaging Service (zaqar)** - Provides a messaging service for the OpenStack Workflow Service.

Red Hat OpenStack Platform director can be deployed both as a bare metal node or virtualized on KVM. Red Hat recommends a virtualized undercloud for greater deployment flexibility.

Overcloud

The overcloud is the resulting Red Hat OpenStack Platform environment created using the undercloud, and it is where the user-facing workloads will run. This includes one or more of the following node types:

Controller nodes provide administration, networking, and high availability for the OpenStack environment. An ideal OpenStack environment comprises at least three of these nodes together in a high-availability cluster. In Red Hat OpenStack Platform 10, a new concept—composable roles— enables having more than three servers as part of the control plane. A default Controller node contains Horizon, Keystone, Nova API, Neutron Server, Open vSwitch, Glance, Cinder Volume, Cinder API, Swift Storage, Ironic, Swift Proxy, Heat Engine, Heat API, Ceilometer, Sahara, Manila, Open vSwitch, MariaDB, MongoDB, and RabbitMQ. The Controller also uses Pacemaker and Galera for services high availability.

Compute nodes provide computing resources for the OpenStack environment. You can add more Compute nodes to scale out your environment over time. By default, compute nodes contain Nova Compute, Nova KVM, Ceilometer Agent, Open vSwitch, and Neutron L2 agent. Storage nodes provide storage for the OpenStack environment. This includes nodes for:

- **Ceph Storage nodes:** Used to form storage clusters. Each node contains a Ceph Object Storage Daemon (OSD). In addition, the director installs Ceph Monitor onto the Controller nodes in situations where it deploys Ceph Storage nodes.
- **Block storage (cinder):** Used as external block storage for HA Controller nodes. This node contains the following components: Cinder Volume, Ceilometer Agent, and Open vSwitch.
- **Object storage (swift):** These nodes provide an external storage layer for

OpenStack Swift. The Controller nodes access these nodes through the Swift proxy. This node contains the following components: Cinder Storage, Ceilometer Agent, and Open vSwitch.

2.3.4 Overview of Red Hat OpenStack Platform Overcloud Services

Red Hat OpenStack Platform provides the three predefined roles for overcloud nodes: Controller, Compute and Storage. Starting from Red Hat OpenStack Platform 10, you will have the flexibility to add or remove services from each of these three roles and create customize roles. The diagram below illustrates different services contained within the three predefined roles:

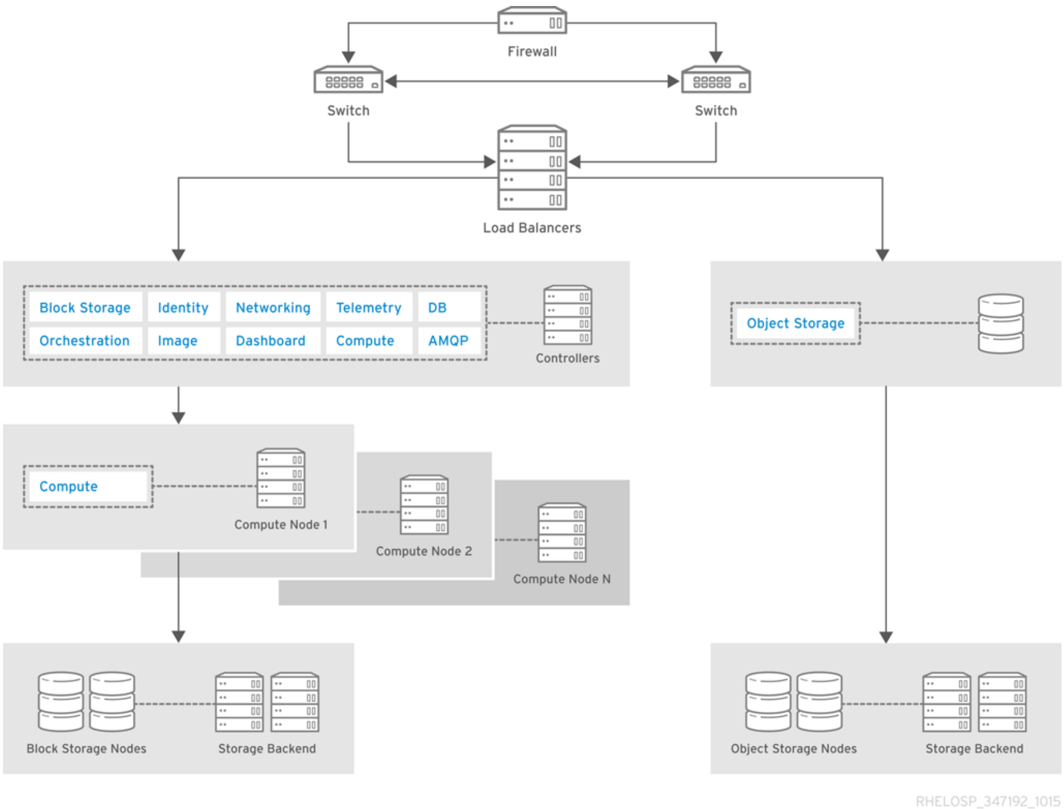


Figure 4. Software Diagram of Overcloud Services.

3. Solution Configuration

3.1 Hardware Configuration for Solution Validation

The hardware configuration designed in this reference architecture includes one director node, three controller nodes, three compute nodes, and three Red Hat Ceph Storage nodes. For Red Hat OpenStack Platform director and the controller nodes, Intel® Xeon® Processor E5-2660 v3 with 10 CPU cores is chosen for providing sufficient processing power. As for the compute nodes, the Intel® Xeon® Processor E5-2670 v3 with 12 cores is chosen for providing outstanding performance in computing. The Solid-State Drives (SSDs) with RAID 1 are enabled on the director, controller, and compute node to provide high availability for the operating system. Red Hat Ceph Storage allows considerable number of HDDs and SSDs for data store. OSD data and OSD journals are stored in separate drives with the ratio of HDD to SSD is 4 to 1. The testing hardware configuration is shown below.

Table 6. Solution Hardware Configuration.

Role	Model	Specification Per Node	Node Quantity
Red Hat OpenStack Director	QuantaPlex T41S-2U (2U4node)	- CPU: 2x E5-2660 v3 - Memory: 256GB - Storage: 2x 200G S3710 SSD with Raid 1 - NIC: 2x 10GbE SFP+ dual ports - NIC of MGMT: 1 dedicated mgmt port	1
Controller	QuantaPlex T41S-2U (2U4node)	- CPU: 2x E5-2660 v3 - Memory: 128GB - Storage: 2x 200G S3710 SSD with Raid 1 - NIC: 2x 10GbE SFP+ dual ports - NIC of MGMT:1 dedicated mgmt port	3
Compute	QuantaPlex T41S-2U (2U4node)	- CPU: 2x E5-2670 v3 - Memory: 256GB - Storage: 2x 200G S3710 SSD with Raid 1 - NIC: 2x 10GbE SFP+ dual ports - NIC of MGMT: 1 dedicated mgmt port	3

Storage	QuantaGrid D51PH-1ULH	- CPU: 2x E5-2660 v3
		- Memory: 128GB
		- Storage:
		  - System: 1x SATA 200G S3710 SSD
		  - Ceph OSD: 12x 6TB HDD 7.2krpm
		  - Ceph Journal: 3x SATA 200G S3710 SSD
		- NIC: 1x 10GbE SFP+ dual ports
		- NIC of MGMT: 1 dedicated mgmt port
			3

Management Switch	QuantaMesh T1048-LB9	- Port Configuration: - 48x 100/100/1000 BAE-T - 4x 1/10Gbe SFP+ ports	1
ToR Switch	QuantaMesh T3048-LY8	- Port Configuration: - 48 x 1/10GbE SFP - 6 x QSFP+ ports - CPU: Freescale P2020 - Memory: 2GB DDR3/ECC - Flash: 128MB - Storage: 8GB Micro SD	2

3.2 Network Planning

A well-designed network topology ensures the efficiency, correctness, and availability of the communication between running services. Red Hat OpenStack Platform uses Neutron networking service to manage the software-based networks, static and floating IP addresses, and the DHCP service. The OpenStack services are mapped to separate network traffic types assigned with various network subnets. In order to optimize the network performance and the system reliability, the network topology of QxStack with Red Hat OpenStack Platform is designed based on the Red Hat OpenStack Platform 10. For more details, please refer to Planning Your Overcloud[5].

In the following section, we will present the default network configurations in QxStack with Red Hat OpenStack Platform, including network types introduction, subnet assignment, and the logical network topology.

3.2.1 Network Types and Subnet Assignment

The network types, VLAN IDs, and subnets are used in QxStack with Red Hat OpenStack Platform by default, as shown in Table 7.

Table 7. Network Types and Subnet Assignment in Overcloud.

Network Type	VLAN ID	Subnet Details	Description
External Network	517	10.5.17.0/24	The External network not only hosts the OpenStack Dashboard (Horizon) for graphical system management but also handles the public API for OpenStack services. Moreover, this network performs SNAT service, which provides the external access for the running VMs.

[5] Planning Your OverCloud: https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installation_and_usage/chap-planning_your_overcloud#sect-Planning_Networks

Provisioning Network	519	10.5.19.0/24	This network traffic type handles the deployment of the overcloud nodes over PXE boot and orchestrates the installation of the overcloud environment on the bare metal servers. This network is predefined before the installation of the Red Hat OpenStack Platform director.
Internal API Network	201	172.16.0.0/24	The Internal API network is used for the communication between the OpenStack services using API communication, RPC messages, and database communication.
Tenant Network	202	172.17.0.0/24	The Neutron service allows each cloud user (tenant) to manage their own network environment using either VLAN segregation or tunneling.
Storage Network	203	172.18.0.0/24	This network traffic type handles in and out traffics of the storage service, including block storage, NFS, and iSCSI. Ideally, storage network can be isolated to a dedicated network interface for performance optimization.
Storage Management Network	204	172.19.0.0/24	Storage management network is used to synchronize data between replica nodes as well as to handle the proxy traffic between user requests and the underlying storage.

3.2.2 Logical Layout

As mentioned in Section 2.3, Red Hat OpenStack Platform director provides multiple node types to build the overcloud environment, including controller, compute, and Red Hat Ceph Storage. Different individual services are assigned to each node for easy management and optimal system resource utilization. Likewise, network types are attached to the overcloud nodes based on the assigned services.

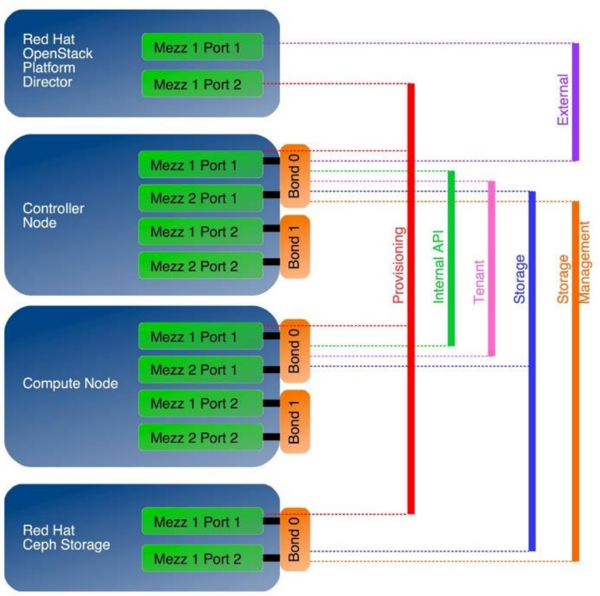


Figure 5. Network Layout in Overcloud.

Figure 5 demonstrates the default network topology of QxStack with Red Hat OpenStack Platform. Each overcloud node uses at least two interfaces in a bond to deliver networks on respective VLANs. All the overcloud nodes can communicate with Red Hat OpenStack Platform director through the provisioning network with native VLAN setting on the first port of NIC 1.

3.3 Physical Cabling and Network Diagram

In order to guarantee the network availability and maximize the network throughput, the top-of-rack (ToR) switch networking architecture is implemented in QxStack with Red Hat OpenStack Platform. From the perspective of the server, two network ports are bonded as an interface; on the other hand, link aggregation technology is enabled in the switch operating system to aggregate the ports from separated switches for HA purpose. The diagram below shows the logical wiring of our solution. Ports in a bonded interface are connected to different switches separately where each is tagged with the corresponding VLAN IDs.

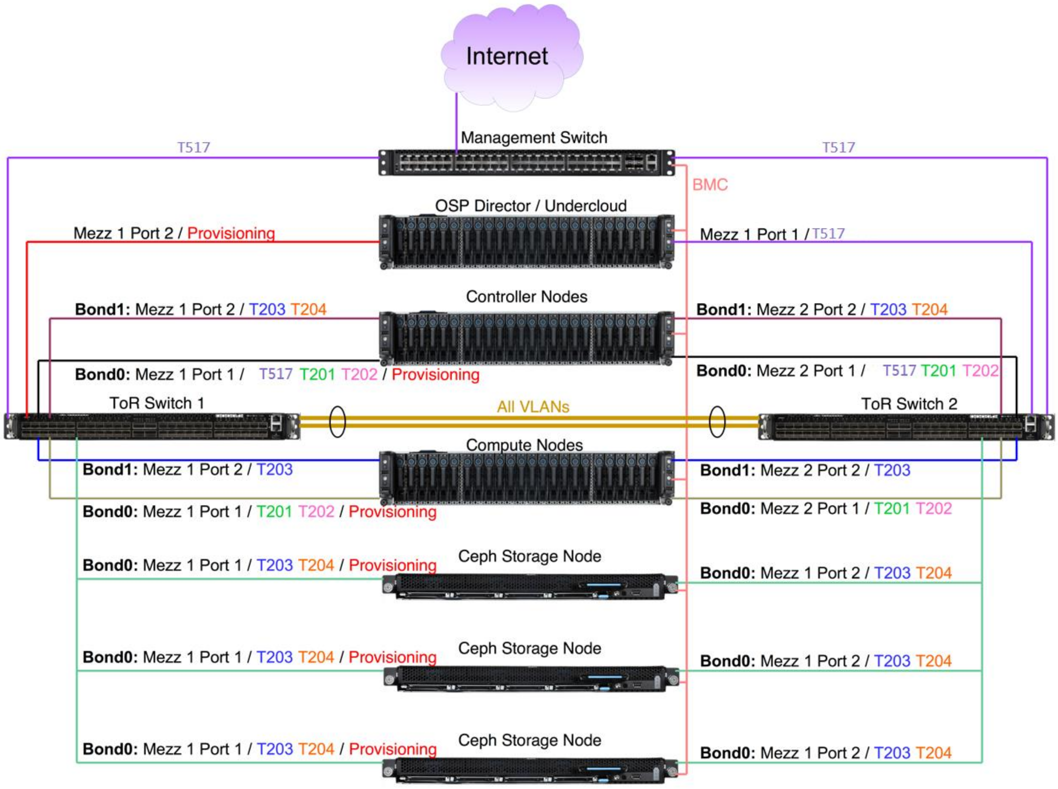


Figure 6. Physical Cabling Arrangement.

3.4 Server Network Interface Configuration

Table 8 shows the detailed network interface configurations in role basis:

Table 8. Server Network Interface Configuration.

Server Network Interface Configuration			
Node Role	Interface	LACP Details	VLANs
Red Hat OpenStack Platform director	1G Interface (IPMI management)	N/A	517
	10G interface	Active/Active, Mode 4	517 519 201-204
Controller	1G Interface (IPMI management)	N/A	517
	10G interface - Bond 1	Active/Active, Mode 4	517 519 201-202
	10G interface - Bond 2	Active/Active, Mode 4	203-204
Compute	1G Interface (IPMI management)	N/A	517
	10G interface - Bond 1	Active/Active, Mode 4	519 201-202
	10G interface - Bond 2	Active/Active, Mode 4	203
Red Hat Ceph Storage	1G Interface - Port 1 (IPMI Management)	N/A	517
	1G Interface - Port 2 (Not Connected)	N/A	N/A
	10G interface - Bond 1	Active/Active, Mode 4	519 203-204

4. QxStack Auto-Deployment Tool

4.1 Solution Deployment

Using Red Hat OpenStack Platform director for Red Hat OpenStack Platform 10 deployment is much easier than manually installing OpenStack platform. Red Hat OpenStack Platform director takes advantage of OpenStack components to install a fully operational OpenStack for users to run the guest workloads. It not only provisions and controls bare metal systems as overcloud nodes but also leverages the Heat Orchestration Templates (HOT) as a definition template for overcloud deployment. Heat templates include some features such as compute resources, network isolation, storage configuration, security groups, and other custom resources. By collecting resources, the overcloud environment can be deployed to fit individual needs.

Director Installation and Usage[6] introduces the deployment steps in detail for creating overcloud environment, while Advanced Overcloud Customization[7] proposes the methods for customizing advanced features using Red Hat OpenStack director. To deploy Red Hat OpenStack Platform director, administrators should first install Red Hat Enterprise Linux 7.3 and register the system to Red Hat Content Delivery Network (CDN). Before the deployment, some preliminaries should be finished, such as creating users and directories for deployment and installing the required packages. Then, the network configuration and required parameters for the director installation should be defined in `undercloud.conf`. After all the parameters are set, administrators can run commands and move on to the overcloud deployment.

To deploy the overcloud environment, all the bare metal systems should be prepared in advance, and all the information should be defined in template `instackenv.json`. Subsequently, the hardware is introspected for various purposes, such as profile tagging, benchmarking, and disk assignment. To fulfill administrators' demands in the overcloud environment, configurations including network, storage, and other custom resources should be set up properly in the heat template files.

---

6 Director Installation and Usage: https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html-single/director_installation_and_usage

7 Advanced Overcloud Customization: https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html-single/advanced_overcloud_customization/

When all the configurations are set, the commands are run for the overcloud deployment. However, customizing heat templates is the most difficult step to build the entire TripleO solution. Among hundreds of the templates, QCT has developed a straightforward way for customers to build the environment based on their requirements.

4.2 Automation Process Overview

In order to facilitate the deployment of QxStack with Red Hat OpenStack Platform, QCT developed a QxStack Auto-Deployment Tool which can dramatically reduce time and minimize guest effort required for building private cloud solutions. The QxStack Auto-Deployment Tool aims to simplify the heat template customization as well as to package all the installation commands to one single program with error tolerance during installation.

Ansible is an open-sourced automation and configuration management technology that automates cloud provisioning, application deployment, and cloud infrastructure management across virtual and physical environments. It not only simplifies the deployment and management process but also automates the entire application life cycle for the continuous delivery pipeline. Ansible, known as a key to DevOps, is a simple and powerful IT automation engine that drives complexity out of the environments. It relies on OpenSSH, the most popular and most widely used implementation for remote access, to connect to the managed nodes for DevOps initiatives. Ansible leveraged in the QxStack Auto-Deployment Tool can not only configure heat templates based on customers' demands but also quickly deploy OpenStack environments without human intervention. Most importantly, multiple cloud environments can be deployed in a single run.

Using the QxStack Auto-Deployment Tool is easy and straightforward. QCT defined a set of parameters to generate the templates and scripts for overcloud deployment. As shown in Figure 7, administrators need to prepare a list of host inventory, the QxStack Auto-Deployment Tool, and all the required parameters in key-value format. By merely running a command, the customized cloud environment can be automatically deployed within an hour. The deployment process can be simplified as follows:

List all the required parameters and information:

- Information such as hostname, user password, the scale of the overcloud, etc.
- An information list for all machines, including MAC addresses, IPMI username, password, etc.

- Network configurations for undercloud and overcloud environment.
- Storage configurations for Ceph Storage nodes.

Run the following command for deployment:

```
$ ansible-playbook -i host install/QxStack_auto-deploy.yml
```

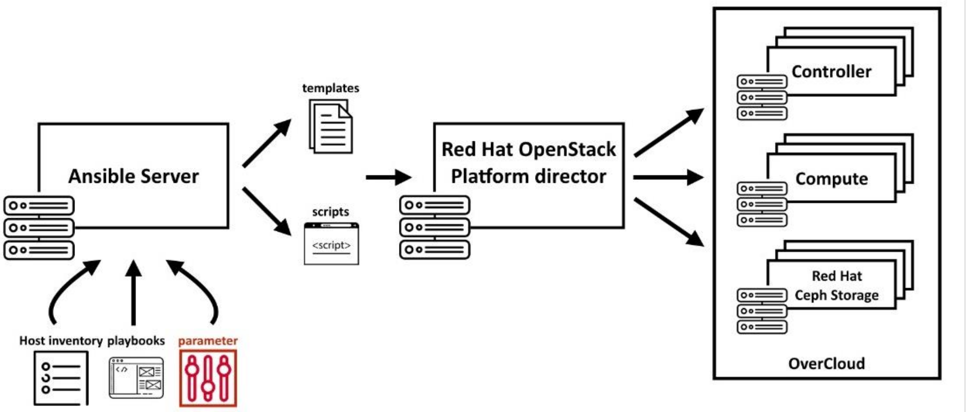


Figure 7. QxStack Auto-Deployment Workflow

4.3 Environment Customization and Heat templates

In this section, we will demonstrate how to customize the overcloud environment with respect to networking, Red Hat Ceph Storage, and other custom settings.

4.3.1 Network Environment

The network environment is the most important but complicated part in overcloud customization. Red Hat OpenStack Platform director is used to install the Red Hat OpenStack Platform overcloud. Please note that the heat template for network environment should be modified based on customers' demands. The `network-environment.yaml` defines the overcloud network environment, including the links to the custom network interface templates for each node role as well as the subnets and VLANs for different network types. Moreover, this file also defines the IP allocation pool and network bonding options for assigning IP addresses to those network types in your environment.

In overcloud creation, a set of network interface templates should be defined for interface configuration and settings, subnet creation, network bonding options, etc. Administrators should customize these templates on a role basis before installing the overcloud. Take the `controller.yaml` as an example; the original network interface template of controller node is demonstrated as follows. For other networking scenarios, please refer to Network Interface Template Examples[8].

```
heat_template_version: 2015-04-30
description:
  ...

parameters:
  ControlPlaneIp:
  ...

resources:
  OsNetConfigImpl:
    type: OS::Heat::StructuredConfig
    properties:
      group: os-apply-config
      config:
        os_net_config:
          network_config:
           # the first network interface (bonded) is specified here
           -
             type: ovs_bridge
             name: {get_input: bridge_name}
             dns_servers:{get_param: DnsServers}
             addresses:
               -
                 ip_netmask:
                   list_join:
                     - '/'
                     - - {get_param: ControlPlaneIp}
                       - {get_param: ControlPlaneSubnetCidr}
             routes:
               -
                 ip_netmask: 169.254.169.254/32
                 next_hop: {get_param: EC2MetadataIp}
             members:
               # the members for this bonded interface is specified here
               -
                 type: linux_bond
                 name: bond-traffic
                 bonding_options: {get_param: BondInterfaceOvsOptions}
                 members:
                   -
                     type: interface
                     name: nic1
                     primary: true
                   -
                     type: interface
                     name: nic3
                     primary: true
               # subnets that running on this interface are listed here
               -
                 type: vlan
                 device: bond-traffic
                 vlan_id: {get_param: ExternalNetworkVlanID}
                 addresses:
                   -
                     ip_netmask: {get_param: ExternalIpSubnet}
                 routes:
                   -
                     default: true
```

[8] Network Interface Template Examples:https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html-single/advanced_overcloud_customization/#appe-Network_Interface_Template_Examples

```
                      next_hop: {get_param: ExternalInterfaceDefaultRoute}

                 # specify more network subnets if needed.
```

The QxStack Auto-Deployment Tool gathers a set of required parameters, enabling administrators to create those heat templates by providing parameters in key-value format. All the parameters for network interface configurations should be defined in a variable file `overcloud_networking.yaml`:

```
# customize the bond option based on your switch configuration
bond_options: "mode=802.3ad"

control_nic:

  # specify the network interface
  - type: ovs_bridge
    name: br-traffic
    interface_type: linux_bond

    # list out the information for your network interface
    bond_name: bond-traffic
    bond_member:
      - type: interface
        nic: nic1
        primary: true
      - type: interface
        nic: nic3

    # define the subnets which runs on this network interface
    subnets:
      - External
      - InternalApi
      - Tenant

# specify more network interfaces on your demands
```

By customizing those parameters in the `overcloud_networking.yaml` file, the corresponding heat templates can be generated for overcloud deployment. In order to flexibly customize overcloud, the QxStack Auto-Deployment Tool is designed as modularized as possible. Every variable file is noted with detailed instructions to help customers to easily configure their cloud. All parameters from Red Hat OpenStack Platform are supported. Please refer to Network Interface Parameters[9] for more details.

4.3.2 Storage Environment

[9] Network Interface Parameters: https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html-single/advanced_overcloud_customization/#appe-Network_Interface_Parameters

In QxStack with Red Hat OpenStack Platform, QCT only provides Red Hat Ceph Storage for the storage backend. Although the overcloud image has original built-in puppet modules, administrators still need to pass the intended configurations to the `storage-environment.yaml` file for enabling Red Hat Ceph Storage in the overcloud environment. The `storage-environment.yaml` file defines the overcloud storage environment, including the storage backend for OpenStack services, the OSD mapping, and the links for the Red Hat Ceph storage resources.

With the QxStack Auto-Deployment Tool, administrators can simply define those parameters in key-value format in `overcloud_storage.yml`. Take OSD mapping as an example below:

```
# set to TRUE to co-located journals on the OSD disks
journal_with_osd: False

# list your OSD disks and journals
osd_list:

  # the ratio of OSD disks (HDD) to OSD journals (SSD) is 4 to 1 – group 1
  - disk: /dev/sde
    journal: /dev/sdb
  - disk: /dev/sdf
    journal: /dev/sdb
  - disk: /dev/sdg
    journal: /dev/sdb
  - disk: /dev/sdh
    journal: /dev/sdb

  # the ratio of OSD disks (HDD) to OSD journals (SSD) is 4 to 1 – group 2
  - disk: /dev/sdi
    journal: /dev/sdc
  - disk: /dev/sdj
    journal: /dev/sdc
  - disk: /dev/sdk
    journal: /dev/sdc
  - disk: /dev/sdl
    journal: /dev/sdc

  # the ratio of OSD disks (HDD) to OSD journals (SSD) is 4 to 1 – group 3
  - disk: /dev/sdm
    journal: /dev/sdd
  - disk: /dev/sdn
    journal: /dev/sdd
  - disk: /dev/sdo
    journal: /dev/sdd
  - disk: /dev/sdp
    journal: /dev/sdd
```

Administrators can define whether to co-locate journals on the OSD disks simply through a True/False flag. If the `journal_with_osd` is set to `True`, the value above for `journal` can be ignored. In the default configuration of QxStack with Red Hat OpenStack Platform, twelve 6 TB hard disks can act as the Ceph OSD and three

200 GB SSD as the journal disk. Every 4 OSD is mapped to 1 journal disk to optimize the performance and resource utilization.

4.3.3 Other Customization Settings

Besides the abovementioned environment settings, Red Hat OpenStack Platform director also allows administrators to define their own resources or scripts running on all nodes upon the initial creation. For example, to avoid unnecessary errors, the QxStack Auto-Deployment Tool can wipe all the disks on the overcloud nodes on first boot through `wipe-disk.sh` script. Meanwhile, the root passwords and ssh settings are assigned through `firstboot-setup.sh` script. Administrators can provide their own scripts during the boot time in the overcloud nodes. The variables can be specified in `overcloud_custom.yml` as follows:

```
# list out the script information you want to execute during the boot time
custom_list:
  - name: firstboot_config
    filename: “firstboot-setup.sh”
    description: >
      assign root password and SSH setup.

  - name: wipe_disk
    filename: “wipe-disk.sh”
    description: >
      wipe all disks for the overcloud nodes.

  # specify more scripts if required
```

5. Solution Operations: Functionality Test

5.1 OpenStack Tempest

Tempest, the OpenStack integration test suite, consists of functionality tests for the individual software modules on an OpenStack platform. Administrators can verify the operation status of OpenStack APIs as well as find fundamental bugs through Tempest tests between deployments or after disaster recovery. Red Hat OpenStack Platform also leverages Tempest to conduct the integration tests on the overcloud environment. The Tempest package certified by Red Hat, `openstack-tempest`, is designed based on the upstream Tempest to fit Red Hat OpenStack Platform architecture and its complex networking design. Red Hat also developed python scripts to automatically detect the current environment and prepare the necessary resources. For more details, please refer to Validating the Overcloud[10].

5.2 Test Result

Currently, over 1500 functionality tests are defined in the `openstack-tempest` package. The QxStack with Red Hat OpenStack Platform is validated over the major services including Nova compute service, Glance image service, Keystone identity service, Neutron network service, Heat orchestration service, Cinder volumes and Swift object storage service.

The Tempest test results are categorized as one of the three functions: “Passed,” “Skipped,” or “Failed.” The successful requests are listed in the “Passed Functions” with response duration, while the failed and skipped tests are respectively listed in the “Failed Functions” and “Skipped Functions” with root causes. As shown in Figure 8, the test result reveals that no failed test occurs in QxStack with Red Hat OpenStack Platform. For the complete results, please refer to the QxStack GitHub repository[11]. Since 1403 of 1497 tests are passed, the success rate of the overall Tempest test is 93.7 percent. The functionality and interoperability between service APIs can be fully guaranteed in QxStack with Red Hat OpenStack Platform. Take the Nova compute service as an example; Tempest is used to validate that the Nova service API can create, delete, list, and update agents properly. The list of function names and corresponding response time are presented in Figure 9.

[10] Validating the Overcloud: https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installation_and_usage/chap-performing_tasks_after_overcloud_creation#sect-Validating_the_Overcloud

[11] QxStack GitHub Repository: https://github.com/QCT-QxStack/QxStack_with_RedHat_OpenStack_Platform

QCT QxStack with Red Hat OpenStack Platform - Tempest Test Report

Summary | Passed Functions (1403) | Failed Functions (0) | Skipped Functions (94)

```
======
Totals
======
Ran: 1497 tests in 2774.0000 sec.
 - Passed: 1403
 - Skipped: 94
 - Expected Fail: 0
 - Unexpected Success: 0
 - Failed: 0
Sum of execute time for each test: 5774.2967 sec.
```

Figure 8. Summary of Test Result in Tempest.

QCT QxStack with Red Hat OpenStack Platform - Tempest Test Report

Summary | Passed Functions (1403) | Failed Functions (0) | Skipped Functions (94)

```
 1. tempest.api.compute.admin.test_agents.AgentsAdminTestJSON.test_create_agent .... [1.765051s]
 2. tempest.api.compute.admin.test_agents.AgentsAdminTestJSON.test_delete_agent .... [2.061112s]
 3. tempest.api.compute.admin.test_agents.AgentsAdminTestJSON.test_list_agents .... [0.062957s]
 4. tempest.api.compute.admin.test_agents.AgentsAdminTestJSON.test_list_agents_with_filter .... [0.113958s]
 5. tempest.api.compute.admin.test_agents.AgentsAdminTestJSON.test_update_agent .... [1.180232s]
 6. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_add_host_create_server_with_az ....
    [27.171331s]
 7. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_add_host_get_details .... [0.313636s]
 8. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_add_host_list .... [0.666054s]
 9. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_add_remove_host .... [0.384586s]
10. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_create_delete .... [0.316073s]
11. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_create_delete_with_az .... [0.254212s]
12. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_create_update_metadata_get_details ....
    [0.363792s]
13. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_create_update_with_az .... [0.436994s]
14. tempest.api.compute.admin.test_aggregates.AggregatesAdminTestJSON.test_aggregate_create_verify_entry_in_list .... [0.151651s]
15. tempest.api.compute.admin.test_aggregates_negative.AggregatesAdminNegativeTestJSON.test_aggregate_add_existent_host ....
    [2.629884s]
16. tempest.api.compute.admin.test_aggregates_negative.AggregatesAdminNegativeTestJSON.test_aggregate_add_host_as_user ....
    [0.863156s]
17. tempest.api.compute.admin.test_aggregates_negative.AggregatesAdminNegativeTestJSON.test_aggregate_add_non_exist_host ....
```

Figure 9. Passed Functions in Tempest.

Tempest is designed to fit a wide range of deployments for different architectures. It is very common to skip tests which do not fit certain environments. Therefore, some functions are reasonably not enabled and the skipped functions do not influence the overall performance in this reference architecture.

6. Solution Operations: Load Testing

6.1 OpenStack Rally

Rally, the OpenStack benchmarking tool, is an ecosystem of cooperative services for cloud verification, benchmarking, and profiling. Rally is not only a verification tool that leverages Tempest for functionality test in the live OpenStack environment but also a benchmark engine used to create parameterized load for performance evaluation. Rally is proven to be of great use in the OpenStack CI/CD system, which allows administrators to continuously improve the performance and stability of a production environment. Several criteria such as budget, expected workload, and flexibility in scale are considered to build a private cloud. Using OpenStack Rally, QCT conducted a comprehensive load test to demonstrate the maximum capability of the QxStack with Red Hat OpenStack Platform. In this section, the testing methodology, the service-level agreement (SLA), and the testing results are presented.

6.2 Test Methodology

An SLA is an official commitment between a solution provider and its users regarding service quality, availability, and stability. In this reference architecture, reasonable and generally acceptable SLAs are defined for each OpenStack service. The result of load test is based on the premise of SLA definitions to present the maximum concurrent workloads in real-world use cases. Administrators can easily determine the solution scale based on their expected workload, such as the maximum number of running VMs.

In order to ensure the reliability of the test result, QCT designed a methodology to recursively execute a test for each OpenStack service to find the accurate maximum performance. The binary search algorithm is used to approach the maximum value under SLA. To ensure stability, this long-running test is continuously executed for verifying maximum concurrent workloads for each scenario. However, load tests manually executed over the solution consumes considerable time for administrators. QCT leverages the continuous integration and delivery (CI/CD) concept into the testing pipeline, aimed at performing the 24-hour non-stop load test on the solution. Once a Rally test is initialized, the CI server can automatically launch all necessary tasks using the abovementioned methodology on the designated cloud. All tests are iteratively executed, and the testing status can be simultaneously updated on real-time messaging applications. The result for each test is published to QxStack GitHub repository.

6.3 Test Results

Considering the workload in real-world use cases, QCT redefined the SLAs in OpenStack Rally tests based on the hardware configuration of QxStack with Red Hat OpenStack Platform. No error should occur in each single test which means `failure_rate` should be equal to zero. The maximum average duration (`max_avg_duration`) is defined under SLA constraints.

The load test is conducted based on the major OpenStack services, including authentication, Nova compute service, Glance image service, Keystone identity service, Neutron network service, Cinder volumes and Swift object storage service. All services tested on OSP are demonstrated in the Rally report.

In the following section, the three common scenarios are addressed to demonstrate the maximum capability of QxStack with Red Hat OpenStack Platform. For more scenarios, please refer to QxStack GitHub repository.

Scenario 1: Authenticate OpenStack Service
Scenario 2: Boot Instances and Associate Floating IPs
Scenario 3: Boot Instances and Live Migrate Servers

<u>Scenario 1: Authenticate OpenStack Service</u>

The authentication tests validate OpenStack services such as Keystone, Cinder, Glance, Heat, Neutron, and Nova. Take `Authenticate.validate_cinder` as an example; Rally issued a token validation process through the Cinder client. All the tasks are passed with no failure, and the maximum average duration of one iteration is 2.71 seconds under the SLA constraints, as shown in Figure 10. The concurrency value is 94 tasks in the testing source code, as shown in Figure 11. In short, approximately 94 concurrent actions on the QxStack with Red Hat OpenStack Platform are able to get a response within 3 seconds.

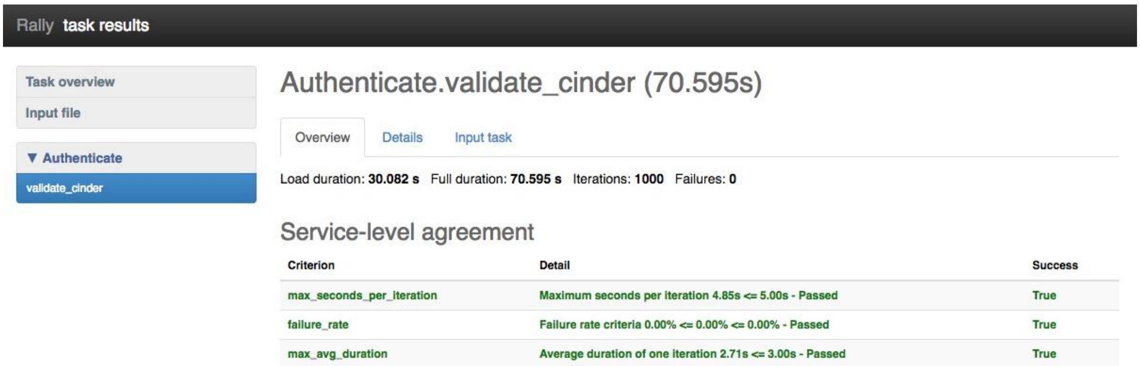


Figure 10. Summary of Test Result in Rally - “Authenticate.validate_cinder”.

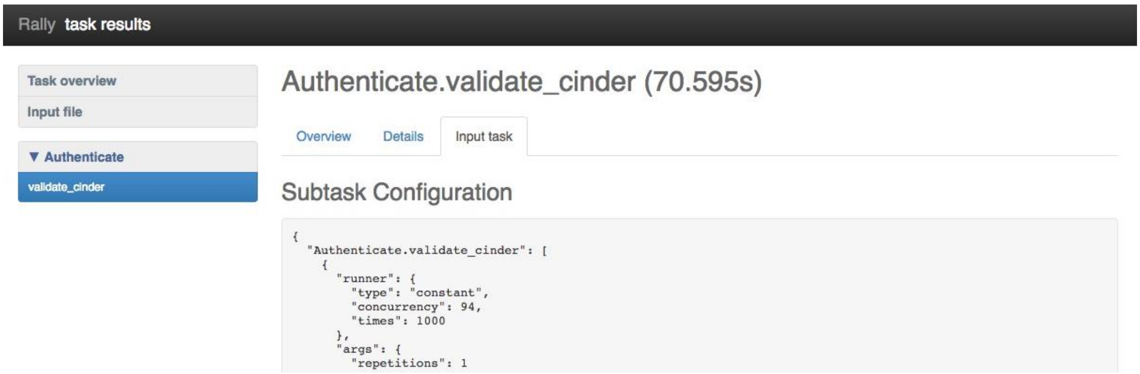


Figure 11. Task Configuration in Rally - “Authenticate.validate_cinder”.

Scenario 2: Boot Instance and Associate Floating IPs

“Boot instance and associate floating IP” is one of the most frequent scenarios in a cloud platform. QCT leverages the `NovaServers.boot_and_associate_floating_ip` scenario to create an instance with CirrOS and associate floating IP for users to access. The SLA report shown in Figure 12 reveals that all the tasks are passed with no failure, and the maximum average duration of one iteration is 28.64 seconds under the SLA constraints.

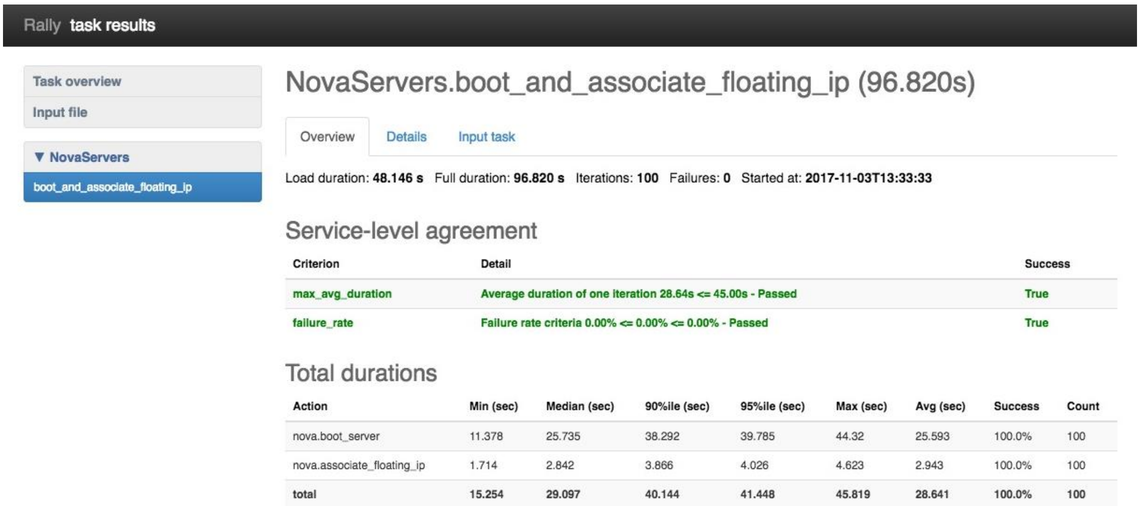


Figure 12. Summary of Test Result in Rally - “NovaServers.boot_and_associate_floating_ip”.

The “Load Profile” report shows the parallel iterations on time basis, which is considered to be the concurrency workload within the duration. In this case, the report reveals that QxStack with Red Hat OpenStack Platform is able to sustain approximately 100 concurrent workloads in “Boot instance and associate floating IP” scenario under the SLA constraints, as shown in Figure 13.

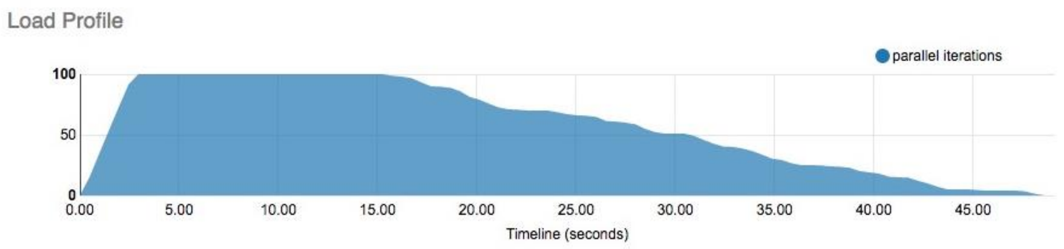


Figure 13. Load Profile in Rally - "NovaServers.boot_and_associate_floating_ip".

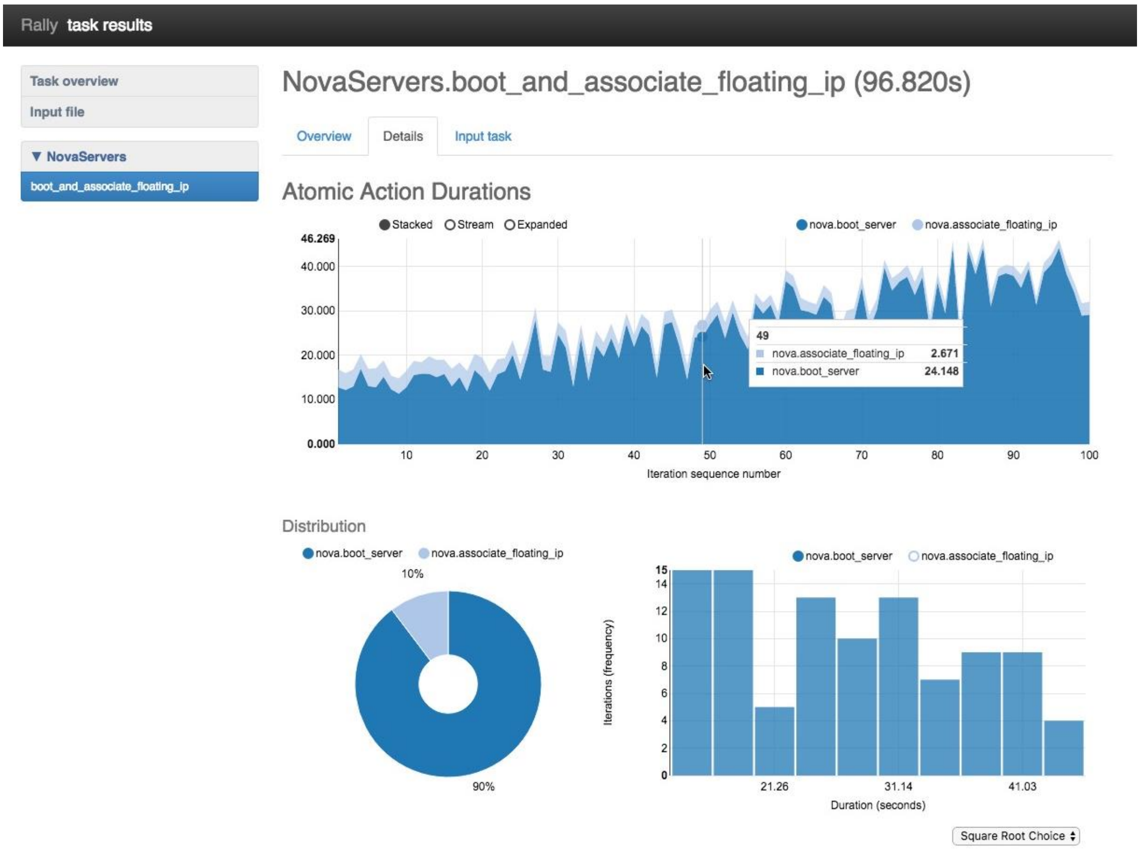
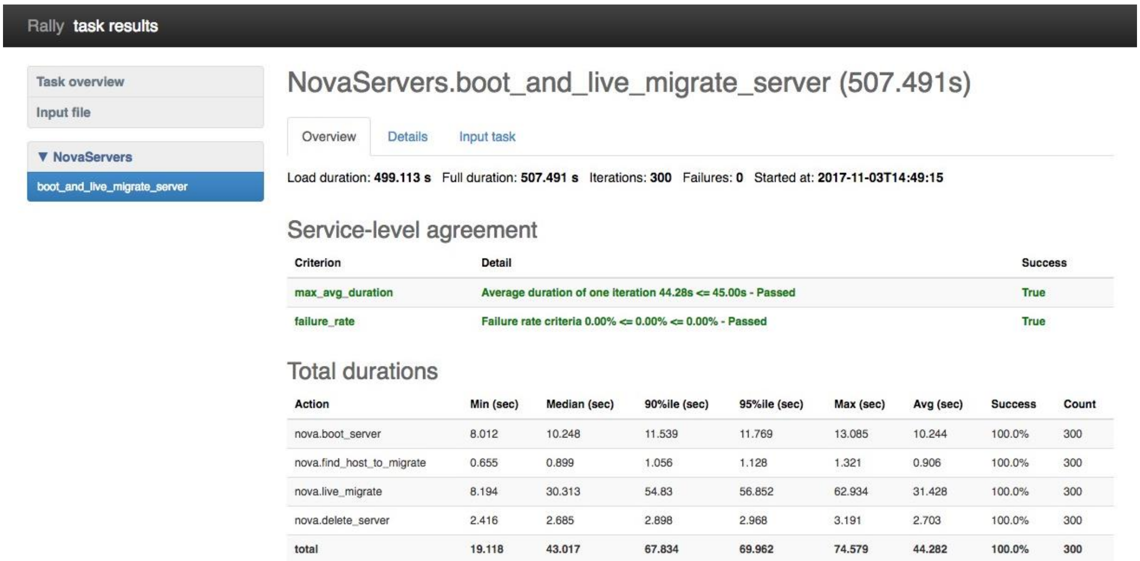


Figure 14. Task Duration and Distribution in Rally - "NovaServers.boot_and_associate_floating_ip".

The `NovaServers.boot_and_associate_floating_ip` scenario consists of two operations, namely, `boot_server` and `associate_floating_ip`. The "Atomic Action Durations" report presents the duration of each iteration in a stacked chart, while the "Distribution" report presents the distribution of the durations in operation basis, as shown in Figure 14. The "Atomic Action Durations" report shows that the first several tasks are rapidly executed on OpenStack platform. As the constantly activated tasks increase the system load, the action duration is prolonged until it fails to meet the SLA. The "Distribution" report shows that `boot_server` operation takes approximately 90 percent of overall duration, while `associate_floating_ip` takes approximately 10 percent. Most of the `boot_server` operations are completed within 20 seconds.

Scenario 3: Boot and Live Migrate Servers

To maximize resource utilization, the “boot and live migrate server” scenario is also critical for administrators. A series of operations including `boot_server`, `find_host_to_migrate`, `live_migrate`, and `delete_server` are performed in the `NovaServers.boot_and_live_migrate_server` scenario. The “service-level agreement” report shows that all the iterations are passed with no failure. The maximum average duration of an iteration is 44.28 seconds under the SLA constraints.


Rally task results

Task overview
Input file
▼ NovaServers
boot_and_live_migrate_server

NovaServers.boot_and_live_migrate_server (507.491s)

Overview Details Input task

Load duration: **499.113 s** Full duration: **507.491 s** Iterations: **300** Failures: **0** Started at: **2017-11-03T14:49:15**

Service-level agreement

Criterion	Detail	Success
max_avg_duration	Average duration of one iteration 44.28s <= 45.00s - Passed	True
failure_rate	Failure rate criteria 0.00% <= 0.00% <= 0.00% - Passed	True

Total durations

Action	Min (sec)	Median (sec)	90%ile (sec)	95%ile (sec)	Max (sec)	Avg (sec)	Success	Count
nova.boot_server	8.012	10.248	11.539	11.769	13.085	10.244	100.0%	300
nova.find_host_to_migrate	0.655	0.899	1.056	1.128	1.321	0.906	100.0%	300
nova.live_migrate	8.194	30.313	54.83	56.852	62.934	31.428	100.0%	300
nova.delete_server	2.416	2.685	2.898	2.968	3.191	2.703	100.0%	300
total	**19.118**	**43.017**	**67.834**	**69.962**	**74.579**	**44.282**	**100.0%**	**300**

Figure 15. Summary of Test Result in Rally - “NovaServers.boot_and_live_ migrate_server”.

The “Atomic Action Durations” report reveals that the execution time of live migration is relatively increased with the system load. However, live migration does not constantly occur in real-world use cases. The first several iterations with light load cases infer that the execution time of live migration generally takes 8 seconds on QxStack with Red Hat OpenStack Platform, as shown in Figure 16. Each live migration task can be finished in 20 seconds, while 100 tasks are constantly executed. In the “Distribution” report, the `live_migrate` operation takes approximately 69 percent of overall duration, while `boot_server` takes approximately 23 percent. Most of the `boot_server` operations are completed within 10 seconds even if the tasks are running under heavy system load.

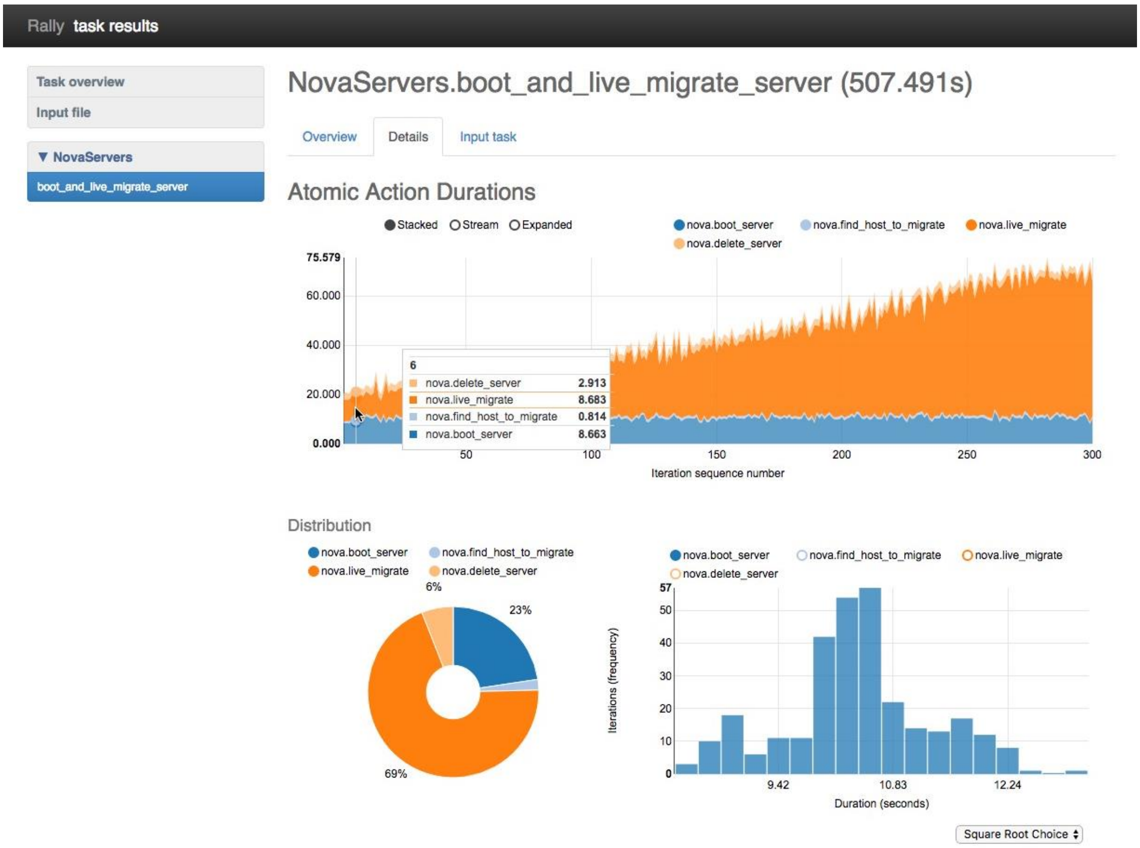


Figure 16. Task Duration and Distribution in Rally - “NovaServers.boot_and_live_ migrate_server”.

7. Solution Operations: HA Testing

7.1 High-Availability Feature in QxStack with Red Hat OpenStack Platform

High-availability (HA) tests, also known as failover tests, ensure the service availability when one or more servers are failed or become unavailable. The HA tests are designed to simulate failures and verify the fault tolerance, reliability, and recoverability of the deployed system. They not only measure the business impact from system failures but also help in planning for reducing impact.

To keep an OpenStack environment up and running with strong reliability, Red Hat OpenStack Platform employs several technologies to provide HA and load balancing across all major services in OpenStack. Pacemaker and HAProxy, the major technologies in HA deployment, are implemented on OpenStack controllers. By configuring virtual IP addresses, services, and other features in a cluster, Pacemaker is running as a cluster resource manager to make sure the defined set of OpenStack resources are running and available. On the other hand, HAProxy handles the load balance of network traffic for the OpenStack services. Administrators can log into the overcloud nodes and run commands to investigate how high availability service is implemented as well as the status of the running process.

Red Hat OpenStack Platform director provides the ability to create an overcloud environment with its own Red Hat Ceph Storage as the backend of OpenStack services. Red Hat Ceph Storage is a distributed object storage with extraordinary reliability and scalability. A Ceph Storage Cluster consists of two types of daemons: a Ceph Monitor maintains a master copy of the cluster map, while Ceph OSD Daemon is responsible for storing data as objects on a storage node and providing access over the network. In order to achieve high availability and fault tolerance, Ceph supports a cluster of monitors to prevent system failure. Ceph OSD Daemon stores at least three copies of data for data safety and periodically checks its own state for data retrieving and OSD rebalancing.

In addition to existing HA features in Red Hat OpenStack Platform and Red Hat Ceph Storage, QCT also provides high availability over networking in QxStack with Red Hat OpenStack Platform. The top-of-rack (ToR) networking switches are designed using link aggregation technology to reduce the occurrence of high severity faults on network infrastructures. For controller and compute nodes in QxStack with Red Hat OpenStack Platform, two 2-port Network Interface Cards (NICs) are used, and the ports on the same NIC are connected to different switches to prevent system crash resulting from switch failure. The channel of the corresponding port is bonded from different NICs for fault tolerance. On the other hand, Red Hat Ceph Storage nodes have one 2-port NIC in which each port is separately connected to two switches for high availability.

7.2 Scenario Definition and Tests

The HA test in OpenStack deployment is executed to ensure service reliability and recoverability when unexpected hardware or software failures are experienced. In order to simulate the worst situation, we manually unplug the server power, unplug the network cable and power supply, or even remove the hard drive while instances are running. The changes of system and service status are recorded, and the approximate latency users might experience is estimated. In a successful test, we expect that neither any error nor any interrupted active communication should occur. As for the system architecture, the minimum requirement to enable the high availability feature is listed as follows:

- 3 x Controller nodes
- 3 x Compute nodes
- 3 x Red Hat Ceph Storage nodes
- 2 x Top-of-Rack (ToR) networking switches

The high-availability test is running on top of the standard scale: 3 controller nodes, 3 compute nodes, and 3 Red Hat Ceph Storage nodes together with two ToR networking switches. RAID 1 is used on the system disks for controller and compute nodes to prevent system crash resulting from disk failure.

In order to verify the high-availability function on QxStack with Red Hat OpenStack Platform, test scenarios related to node failures, network failures and disk failures are summarized as follows:

Table 9. HA Test Scenarios for Controller Nodes.

Controller		
Scenario	Description	Expected Results
Node Failure	Unplug the power cord	- Service status should reflect the current status. - All running VMs should not be affected.
	Restore the power	- System should be back online. - Service status should reflect the current status.
Network Failure	Unplug the network cable	- All running VMs should not be affected.
	Reconnect the network cable	

Disk failure	Remove one system disk	- The entire system should not be affected.

Table 10. HA Test Scenarios for Compute Nodes.

Compute		
Scenario	Description	Expected Results
Node Failure	Unplug the power cord	- Service status should reflect the current status. - VM on the node will be forced offline.
	Restore the power	- System should be back online. - Service status should reflect the current status. - VM should be functional after reboot.
Network Failure	Unplug the network cable	- All running VMs should not be affected.
	Reconnect the network cable	
Disk failure	Remove one system disk	- The entire system should not be affected.

Table 11. HA Test Scenarios for Red Hat Ceph Storage Nodes.

Red Hat Ceph Storage		
Scenario	Description	Expected Results
Node Failure	Unplug the power cord	- Service status should reflect the current status. - All running VMs should not be affected.
	Restore the power	- System should be back online and start data recovery. - Service status should reflect the current status. - All running VMs should not be affected.
Network Failure	Unplug the network card	- All running VMs should not be affected.
	Reconnect the network card	
Disk failure	Remove OSD disk	- Data on the disk should be auto rebalanced. - All running VMs should not be affected.

Table 12. HA Test Scenario for ToR Switches.

ToR Switch		
Scenario	Description	Expected Results
Node Failure	Unplug the power cord	- All running VMs should not be affected.
	Restore the power	

7.2.1 High availability tests on Controller nodes

Node Failure Tests

As mentioned in section 7.1, Pacemaker and HAProxy are the major technologies in controller HA deployment. Administrators can check the service status through Red Hat OpenStack Platform director or log in to the overcloud nodes to check the resource location through Pacemaker commands. The Pacemaker command `pcs status` shows the status below:

- Current designated coordinator is `overcloud-controller-0`.
- Currently, three controllers are all online.
- All resources and hosting machines are listed in "Full list of resources".

For example, the virtual IP for controller services is 10.5.19.59, hosted by `overcloud-controller-1`.

```
[heat-admin@overcloud-controller-1 ~]$ sudo pcs status
...
Current DC: overcloud-controller-0 (version 1.1.15-11.el7_3.2-e174ec8) - partition with
quorum
Online: [ overcloud-controller-0 overcloud-controller-1 overcloud-controller-2 ]

Full list of resources:
 Master/Slave Set: galera-master [galera]
     Masters: [ overcloud-controller-0 overcloud-controller-1 overcloud-controller-2 ]
 ip-172.18.0.13    (ocf::heartbeat:IPaddr2): Started overcloud-controller-0
 ip-172.16.0.12    (ocf::heartbeat:IPaddr2): Started overcloud-controller-1
 ip-172.19.0.16    (ocf::heartbeat:IPaddr2): Started overcloud-controller-2
 ip-172.16.0.14    (ocf::heartbeat:IPaddr2): Started overcloud-controller-0
 ip-10.5.19.59     (ocf::heartbeat:IPaddr2): Started overcloud-controller-1
```

If we subsequently crash the current designated coordinator, `overcloud-controller-0`, users might experience a 30-second latency. In this period, the OpenStack commands for overcloud are temporarily unworkable. Users should log into the dashboard again for further executions. One of the remaining controllers takes over the service and becomes the new designated coordinator. The resources cannot be reassigned unless the hosting machine fails again.

The service recovery status can be checked either from the Red Hat OpenStack Platform director using `nova service-list` command or from the system information of the dashboard. Once the `overcloud-controller-0` is restored, it can be automatically back online without any manual configurations. The resources can be reassigned for load balancing among all the controllers.

```
[stack@qct-ospd ~ ]$ source overcloudrc
[stack@qct-ospd ~ ]$ nova service-list
+----+--------------------+-----------------------------------+-----+---------  +----+
| Id | Binary             | Host                              |...  | State     |... |
+----+--------------------+-----------------------------------+-----+---------  +----+
|  41 | nova-consoleauth  | overcloud-controller-2.localdomain |...  | up        |... |
|  44 | nova-consoleauth  | overcloud-controller-0.localdomain |...  | down      |... |
| 104 | nova-scheduler    | overcloud-controller-2.localdomain |...  | up        |... |
| 107 | nova-scheduler    | overcloud-controller-0.localdomain |...  | down      |... |
| 110 | nova-conductor    | overcloud-controller-2.localdomain |...  | up        |... |
| 131 | nova-conductor    | overcloud-controller-0.localdomain |...  | down      |... |
| 152 | nova-consoleauth  | overcloud-controller-1.localdomain |...  | up        |... |
| 155 | nova-scheduler    | overcloud-controller-1.localdomain |...  | up        |... |
| 158 | nova-conductor    | overcloud-controller-1.localdomain |...  | up        |... |
+----+--------------------+-----------------------------------+-----+---------  +----+
```

Note: The troubleshooting for HA controller resources is not within the scope of this document. Please refer to UnderStanding Red Hat OpenStack Platform High Availability[12] for more detail.

Network Failure Tests

QxStack with Red Hat OpenStack Platform is designed to provide fault tolerance when network failures occur. (See networking high availability features in section 7.1.) From a hardware perspective, both unplugged or damaged network cables and an ineffective network card might cause errors. We simulate network card failure, network port failure, and network cable failure by simply unplugging one or more network cables. Meanwhile, instances connected to the internet are launched to estimate the approximate latency that users may experience. According to the test results, the overcloud environment can still be functional in terms of the following damage extent:

- 1 network port or network cable failure for each bonding interface
- 1 network interface card failure

Although the situations above do not cause severe damage on the running system, users still experience a 20-second latency for partial network failure on the designated coordinating controller.

Disk Failure Tests

Since all the storage data are placed in the Red Hat Ceph Storage Cluster, only the system disk is on controller nodes. Two SSDs are used in RAID 1 to prevent system crash resulting from disk failure. If one of the disks is removed or damaged, no error should occur on the running system.

[12] UnderStanding Red Hat OpenStack Platform High Availability: https://access.redhat.com/documentation/en/red-hat-openstack-platform/10/paged/understanding-red-hat-openstack-platform-high-availability/

7.2.2 High availability tests on Compute nodes

Node Failure Tests

On the QxStack with Red Hat OpenStack Platform, a volume can be created and attached by default when a new instance is launched with all the data stored in the Ceph system. Meanwhile, the data safety and recoverability in terms of running instances can be guaranteed through the Red Hat Ceph Storage Cluster. Once the compute node is crashed, users can easily launch another instance with the original volume to retrieve the working files.

Network Failure Tests

As with controller nodes, compute nodes have fault tolerance features that can prevent system crash resulting from network infrastructure failures. The utmost damage extent is listed as follows:

- 1 network port or network cable failure for each bonding interface
- 1 network interface card failure

Although the situations above cannot cause severe damage on the running system, users still experience a 10-second latency for partial network failure on the hosting compute node.

Disk Failure Tests

As with controller nodes, two SSDs are used in RAID 1 to prevent system crash resulting from disk failure on all the compute nodes. If one of the disks is removed or damaged, no error should occur on the running system.

7.2.3 High availability tests on Ceph Storage nodes

Node Failure Tests

As mentioned in section 7.1, a Red Hat Ceph Storage Cluster consists of two types of daemons: Ceph Monitor and Ceph OSD Daemon. A cluster of Ceph Monitor maintains a master copy of the cluster map, while a Ceph OSD Daemon is responsible for storing data as objects on a storage node and provides access over the network. Administrators can log in to the Ceph Storage nodes to manually check the OSD status using `ceph osd stat` command and can use `ceph -s` command for health check of the entire Ceph cluster.

```
[heat-admin@overcloud-cephstorage-0 ~]$ sudo ceph osd stat
     osdmap e92: 36 osds: 36 up, 36 in
            flags sortbitwise
```

```
[heat-admin@overcloud-cephstorage-0 ~]$ sudo ceph -s
    cluster 691947fe-d6dc-11e6-9148-2c600cbc3036
     health HEALTH_OK
     monmap e1: 3 mons at {overcloud-controller-0=10.7.19.14:6789/0, … }
            election epoch 4, quorum 0,1,2 ...
     osdmap e92: 36 osds: 36 up, 36 in
            flags sortbitwise
      pgmap v253: 224 pgs, 6 pools, 0 bytes data, 0 objects
            1279 MB used, 196 TB / 196 TB avail
                 224 active+clean
```

The system crash is simulated by manually unplugging the server. Users can hardly aware if only a portion of the data is disconnected. From the administrator perspective, the data recovery process can be triggered automatically. Once the Ceph Storage node is restored, the data recovery process can be triggered again to ensure the load balance between OSDs.

Network Failure Tests

Unlike the controller and compute nodes, the Ceph Storage node has only one network card responsible for all traffics. We bond the two network ports to provide fault tolerance for network failures. The utmost damage extent is 1 network port or a network cable failure.

However, the high availability features in Ceph Storage hold at least 3 copies in data replication and reliability is increased by spreading load across multiple hosts. If the network card cannot work properly, other Ceph nodes can still provide the data safety and availability.

Disk Failure Tests

For the Red Hat Ceph Storage nodes, only one SSD is used for the system, which means no HA function is provided for system disk failure on storage nodes. However, Red Hat Ceph Storage guarantees data safety and recoverability among disks and servers. The data recovery process can be automatically triggered to prevent data loss resulting from disk failure. The command `ceph -s` can be used to check data recovery process and current health status.

7.2.4 High availability tests on Top-of-Rack (ToR) Switch

Besides network failure on each server, hardware or software failure might also occur on the ToR networking switches. The two ports of each network bond are separately connected to the ToR switches. If only one switch is out of order, users might experience the latency of a few seconds. Nevertheless, no system error or data loss should possibly occur during the recovery process.

8. Conclusion

QxStack with Red Hat OpenStack Platform is a well-validated cloud solution based on Red Hat OpenStack Platform (OSP). To guarantee data safety and system availability, Red Hat OpenStack Platform director is adopted to create an overcloud environment with its own Red Hat Ceph Storage as the backend of OpenStack services. To quickly deploy the QxStack with Red Hat OpenStack Platform, QCT provides the QxStack Auto-Deployment Tool for a streamlined setup process which can highly enhance agility and greatly minimize efforts of deployment. The tool can not only help configure the OpenStack environment based on customer's demands but also deploy the environment without human intervention. Most importantly, the automation tool is regarded as the key to DevOps and can be further integrated to the continuous integration and delivery (CI/CD) life cycle.

QCT introduced the CI/CD concept into the testing pipeline and conducted comprehensive tests for QxStack with Red Hat OpenStack Platform. The functionality test result from Tempest reveals that all the OpenStack service APIs can work successfully with no failure. The load test result from Rally indicates the maximum capability for each service under predefined SLA constraints. HA functions in Red Hat OpenStack Platform and Red Hat Ceph Storage are enabled in QxStack with Red Hat OpenStack Platform, and the ToR network switch architecture designed by QCT ensures HA on network infrastructures. To simulate worst-case scenarios, QCT strictly tested the availability in the cases of node failure, network failure, and disk failure while instances are running. The result reveals that all running VMs are not affected, which means the HA function for each scenario is surely enabled.

QCT utilizes state-of-the-art technologies to develop cloud solutions with high performance, reliability, and flexibility in an agile software development life cycle. This reference architecture documents a comprehensive series of functionality, load, and HA tests that collectively validate the QxStack with Red Hat OpenStack Platform solution. QxStack with Red Hat OpenStack Platform is proven to be a robust and resilient solution with high availability and outstanding performance for enterprises and cloud providers.

Appendix A

Based on the extensive tests described above, the following table is a recommended minimum configuration of the QCT QxStack with Red Hat OpenStack Platform for you to order:

Role	Model	Specification Per Node	Node Quantity
Red Hat OpenStack Platform director (Infra Node)	QuantaPlex T41S-2U (2U 4node)	- CPU: 2x E5-2660 v4 - Memory: 128GB - Storage: 2x 400G S3710 SSD with Raid 1 - NIC: 2x 10GbE SFP+ dual ports - NIC of MGMT: 1 dedicated mgmt port	1
Controller	QuantaPlex T41S-2U (2U 4node)	- CPU: 2x E5-2660 v4 - Memory: 128GB - Storage: 2x 400G S3710 SSD with Raid 1 - NIC: 2x 10GbE SFP+ dual ports - NIC of MGMT:1 dedicated mgmt port	3
Compute	QuantaPlex T41S-2U (2U 4node)	- CPU: 2x E5-2680 v4 - Memory: 256~512GB - Storage: 2x 400G S3710 SSD with Raid 1 - NIC: 2x 10GbE SFP+ dual ports - NIC of MGMT: 1 dedicated mgmt port	8

Storage	QuantaGrid D51PH-1ULH	- CPU: 2x E5-2620 v4
		- Memory: 64GB
		- Storage:
		  - Ceph OSD: 12x 8~10TB HDD 7.2krpm
		  - Ceph Journal: 3x SATA 200G S3710 SSD
		  - Boot Disk: SATADOM 128GB
		- NIC: 1x 10GbE SFP+ dual ports
		- NIC of MGMT: 1 dedicated mgmt port
			3

ToR Switch	QuantaMesh T3048-LY8	- Port Configuration:
		  - 48 x 1/10GbE SFP
		  - 6 x QSFP+ ports
			2

Management Switch	QuantaMesh T1048-LB9	- Port Configuration:
		  - 48x 100/100/1000 BAE-T
		  - 4x 1/10Gbe SFP+ ports
			1

For further inquiry, please contact <QCT-QxStack@qct.io> or visit www.QCT.io.

References

QCT QxStack with Red Hat OpenStack Platform

- QCT QxStack with Red Hat OpenStack Platform website
  http://qct.io/solution/index/Compute-Virtualization/QxStack-with-Red-Hat-OpenStack-Platform
- QCT QxStack with Red Hat OpenStack Platform GitHub Repository
  https://github.com/QCT-QxStack/QxStack_with_RedHat_OpenStack_Platform

Red Hat Enterprise Linux 7

- Red Hat Enterprise Linux 7 Product Documentation
  https://access.redhat.com/documentation/en/red-hat-enterprise-linux/?version=7/

Red Hat OpenStack Platform 10

- Red Hat OpenStack Platform
  https://access.redhat.com/products/red-hat-openstack-platform/
- Red Hat OpenStack Platform Life Cycle
  https://access.redhat.com/support/policy/updates/openstack/platform
- Red Hat OpenStack Platform director Life Cycle
  https://access.redhat.com/support/policy/updates/openstack/platform/director
- Red Hat OpenStack Platform 10 Documentation Home Page
  https://access.redhat.com/documentation/en/red-hat-openstack-platform/
- Red Hat OpenStack Platform 10 director Installation and Usage
  https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/director_installation_and_usage/
- Red Hat Ceph Storage for the Overcloud
  https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/10/html/red_hat_ceph_storage_for_the_overcloud/

Red Hat Ceph Storage 2

- Red Hat Ceph Storage Administration Guide
  https://access.redhat.com/documentation/en-us/red_hat_ceph_storage/2/html/administration_guide/
- Red Hat Ceph Object Gateway Guide
  https://access.redhat.com/documentation/en-us/red_hat_ceph_storage/2/html-single/object_gateway_guide_for_red_hat_enterprise_linux/

About QCT

QCT (Quanta Cloud Technology) is a global datacenter solution provider extending the power of hyperscale datacenter design in standard and open SKUs to all datacenter customers. Product lines include servers, storage, network switches, integrated rack systems and cloud solutions, all delivering hyperscale efficiency, scalability, reliability, manageability, serviceability and optimized performance for each workload. QCT offers a full spectrum of datacenter products and services from engineering, integration and optimization to global supply chain support, all under one roof. The parent of QCT is Quanta Computer Inc., a Fortune Global 500 technology engineering and manufacturing company.   http://www.QCT.io

About Red Hat

Red Hat is the world's leading provider of open source software solutions, using a community- powered approach to provide reliable and high-performing cloud, Linux, middleware, storage, and virtualization technologies. Red Hat also offers award-winning support, training, and consulting services. As a connective hub in a global network of enterprises, partners, and open source communities, Red Hat helps create relevant, innovative technologies that liberate resources for growth and prepare customers for the future of IT.

United States

QCT LLC., Silicon Valley office
1010 Rincon Circle, San Jose, CA 95131
TOLL-FREE: 1-855-QCT-MUST
TEL: +1-510-270-6111
FAX: +1-510-270-6161
Support: +1-510-270-6216

QCT LLC., Seattle office
13810 SE Eastgate Way, Suite 190, Building 1,
Bellevue, WA 98005
TEL: +1-425-633-1620
FAX: +1-425-633-1621

China

云达科技，北京办公室（Quanta Cloud Technology）
北京市朝阳区东三环中路 1 号，环球金融中心东楼 1508 室
Room 1508, East Tower 15F, World Financial Center
No.1, East 3rd Ring Zhong Rd., Chaoyang District,
Beijing, China
TEL: +86-10-5920-7600
FAX: +86-10-5981-7958
云达科技，杭州办公室（Quanta Cloud Technology）
浙江省杭州市西湖区古墩路浙商

Japan

Quanta Cloud Technology Japan 株式会社日本国東京都港区芝大門二丁目五番八号牧田ビル３階
Makita Building 3F, 2-5-8, Shibadaimon ,
Minato-ku, Tokyo 105-0012, Japan
TEL: +81-3-5777-0818
FAX: +81-3-5777-0819

Taiwan

雲達科技（Quanta Cloud Technology）
桃園市龜山區文化二路 211 號 1 樓
1F, No. 211 Wenhua 2nd Rd., Guishan Dist., Taoyuan City 33377, Taiwan
TEL: +886-3-286-0707
FAX: +886-3-327-0001

Other regions

Quanta Cloud Technology
No. 211 Wenhua 2nd Rd., Guishan Dist.,
Taoyuan City 33377, Taiwan
TEL: +886-3-327-2345
FAX: +886-3-397-4770