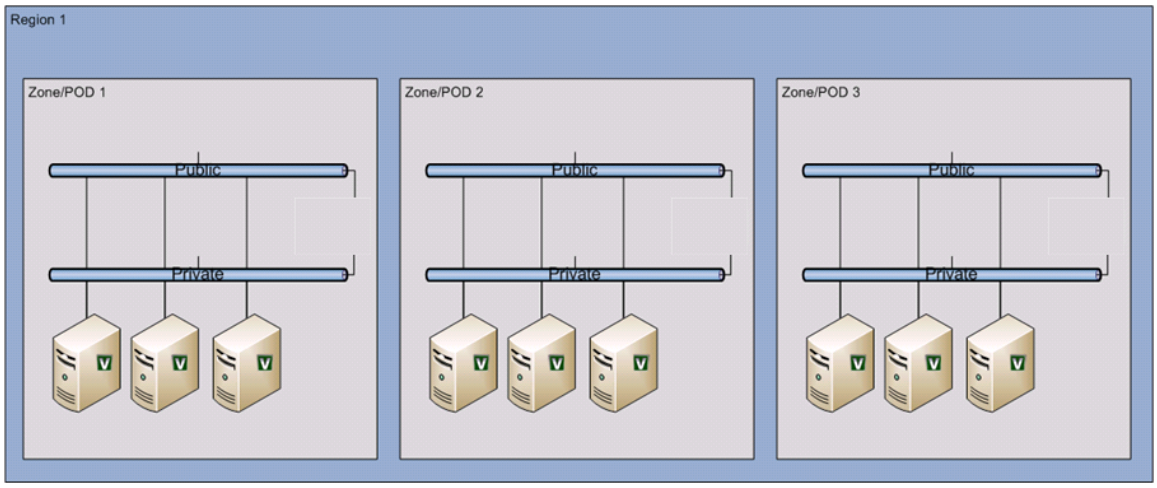
LBaaS integration with Quantum

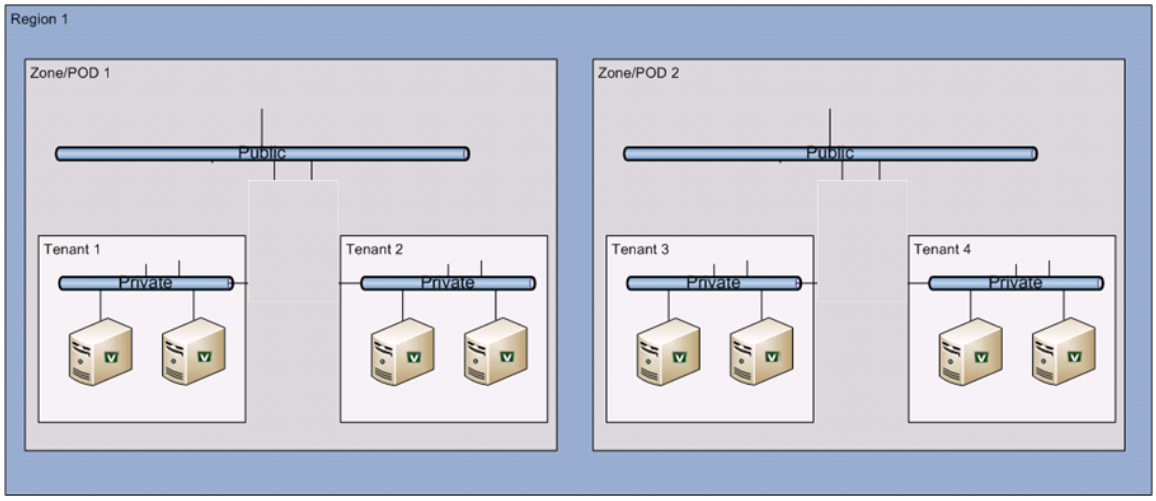
Background

OpenStack/Essex release supports out of Nova a static or semi static network topologies. Those topologies include

- Public flat network and a private flat network

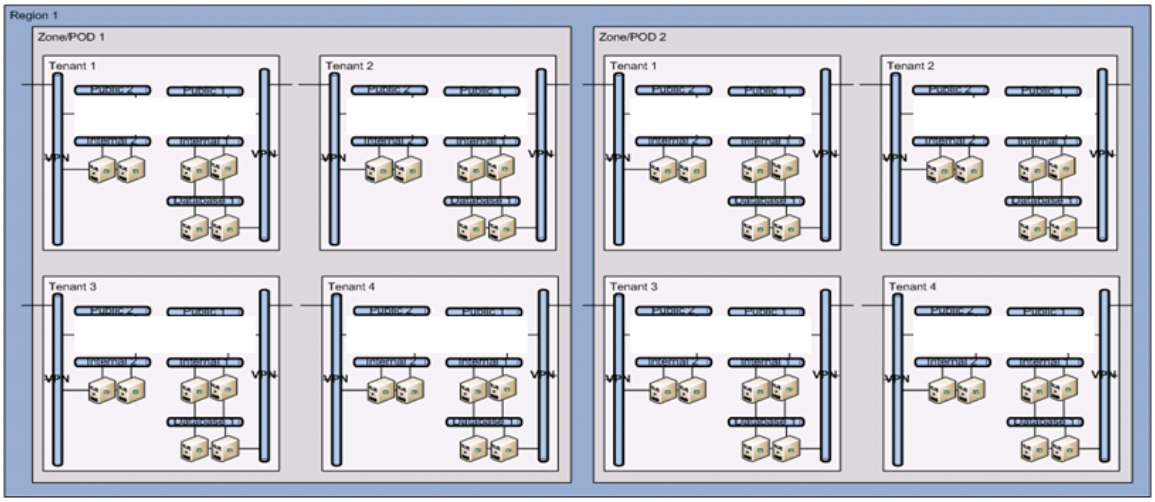


- Public flat network and per tenant (VLAN backed) flat private network.



Starting the OpenStack/Folsom release the OpenStack networking is managed by Quantum which enables explicit presentation and manipulation of advanced network topologies that potentially could be exposed to the cloud tenant. In addition to owning Layer-2 network and Layer-3 IP management, Quantum is also implemented in a generic way that support traditional

VLAN backed networks while also supporting newer technologies such as VXLAN, Nicira, Open-Flow based and more.



After the completion of the Quantum integration into Open Stack which enhances the networking functionality on layer-2 and layer-3, one of the next goals for Quantum is to support advanced networking functionality on Layer-4 to Layer-7 such as Load-balancers, Firewalls, Application firewalls, VPNs and more.

Load Balancer as a Service (LBaaS) integrated with Quantum

Currently there are two types of APIs that need to be covered by LBaaS

1. Tenant/Consumer facing APIs
2. Operator/Administrator facing APIs

This document does not cover the specifics on how each APIs is defined. Such details should be covered by the community discussions in regards to the different API proposals by Mirantis, Atlas-LB, EBay/PayPal, etc.

This document covers two subjects:

1. Exposing LBaaS in OpenStack integrated with Quantum.
2. Quantum support for hardware/software load balancing devices

Exposing LBaaS in OpenStack integrated with Quantum.

Since quantum introduces the notion of dynamically allocated layer-2 networks, the notion of how the device/load-balancer is connected to the network must be incorporated as part of the API. In addition, the ability to configure the L2/L3 connectivity on demand based on the tenants specification suggest that APIs similar to the Operator/Administrator ones might also be available for tenants.

LBaaS as a top level API

In this approach the LBaaS has its own API available as a top level API.

The following objects needs to be considered in regards to Quantum references

- Server Node – the serve node's network identity is represented by a v4/v6 IP address. We need to consider whether in addition to that, the Network ID needs to be added
    - If the API allows specifying a Nova-VM as a Server Node, than the Nova-VM might be connected to multiple networks/subnets and have multiple IPs, hence a logic to detect/specify the interface and the IP /Subnet ID on that interface is needed (Can a VM have multiple IPs on the same interface?)
- Virtual IP – the virtual IP (in Mirantis-LB) includes VLAN. In Atlas-LB it does not. The Virtual IP should include the Network ID where it needs to be connected. In addition, in case the IP is not specified and is instead requested from Qunatum, the Network ID and the subnet ID must be specified.
- Virtual IP address could be allocated as follows:
    - User defined
    - Dynamically allocate from Quantum IP sub-net
    - Allow the use to assign a Floating IP
- Load Balancer – the network IDs and subnet IDs should be specified so that the load balancer would be connected to the needed networks and subnets.
    - Consider support two/more L2 networks routed topology?
    - Consider support two/more L2 networks NAT/Proxy topology?
    - Other topologies?
- Device – Quantum supports multi-tenant network technologies which are not VLAN based. Probably all of the hardware based appliance can only support VLAN based connectivity. An option to specify how a device needs to be connected to the Quantum network is discussed in the section covering "Quantum support for hardware/software load balancing devices".

LBaaS as a Network Service API implementer out of Quantum

In this approach a generic network-service entity will be added to quantum. The network-service will have the following properties:

- Service Type – ex: load-balancer
- Service Profile – ex: SSL Termination=true, HA=true, etc…
- Service Capacity/Quotas – ex: band width limits
- Service Operation Model – ex: Proxy or Routed
- Connected Networks and Subnets – list of Quantum Network IDs and subnets IDs. In the case of load balancers, the list might be limited to two network ID + subnet ID one specifying the server's network and an optionally second network ID specifying the client facing network.

The network-service supports CRUD operations.

- Create a network-service will return a network service instance that has the following

properties in addition to the ones defined in the creation request.

- Service ID – the instance identifier
- Service Management URL – a URL/URI pointing to a tenant API that exposes the specific LBaaS API. Alternatively, the LBaaS could be a URI path concatenated to the base service path: ex:

  GET v2.0/network-service/fc68ea2cb60b4b4fbd8294ec81110766/loadbalancers

  Returns the list of all the load balancers hosted in the service.

Since working under the network-service context defines the network connectivity, most of the Tenant facing LBaaS APIs could remain unchanged.

- Server Node – the serve node's network identity is represented by a v4/v6 IP address. The network service adds the context of the server's network hence no need to repeatedly specify it.
  - If the API allows specifying a Nova-VM as a Server Node, than the Nova-VM will be searched for the interface connected on the server's network and if only one IP address exists no need to specify anything in addition. In addition, the subnet-id can assist in selecting the right IP address of the VM.
- Virtual IP – does not need to specify network ID or VLAN as the network-service client facing network creates this association.
- Load Balancer – there is no need to specify network IDs as they come out of the context of the network-service

Operator facing APIs will need to change so that hardware based load balancers could be defined and connected to a quantum network. The change should be impact by the agreement on "Quantum support for hardware/software load balancing devices" discussed further on.

Quantum support for hardware/software load balancing devices

Background

The following steps might need to take place so that the load balancing device will operate correctly:

- Layer 2
  - Connect the device to the switch
  - Configure the switch and the device on to use the appropriate VLANs
- Layer 3
  - Based on the device role in the network
    - Gateway
    - Proxy
  - Define the IP interfaces addresses
  - Define the IP proxies addresses
  - Define the different Gateways

    - Define static routes
- HA
    - Configure the HA IP addresses / VRRP
    - Configure the HA memberships
    - Configuration Synchronization
    - Session Synchronization

The part which needs special reference in a Quantum environment is managing the Layer 2 connectivity.

Layer 3 configurations needs to consume the information provided by quantum in regards to L3 and also use Quantum for IPAM (IP Address Management).

HA configuration is probably vendor specific and will need to be specially defined and tailored by each load balancing provider.

L2 Support for connecting load balancers

There are three use cases:

1. Connecting software based load balancers – in this case the software is being connected as a virtual machine hence there is no special requirement for Quantum
2. Connecting hardware based load balancers
    a. A device that can natively supports the type of used quantum network – for example, hardware device could probably connected to a VLAN based quantum network
    b. Ad device that can't natively support the type of used quantum network – for example the hardware device can connect to a VLAN based network while the Quantum network is using VXLAN. To support such a configuration, a network-gateway-service is proposed. A network-gateway-service enables translation between the quantum network to another network technology (ex: VXLAN->VLAN). In this model, the hardware load balancer connected to a specific network-gateway-service can ask the service to translate between the Quantum Network ID and a specific VLAN. There could be more than one network-gateway-service. This will allow mapping more than 4096 VLANs for the quantum network as each network-gateway-service can support translation for up to 4096 VLANs but multiple network-gateway-service can extend such support.

To support connecting hardware load balancers to Quantum follows two requirements:

1. An API that returns the implementation Meta data of the Network ID. For example, the technology type, the technology identifier, etc.
2. Support for network-gateway-service(s). When a hardware load balancer is registered into Quantum, the network-gateway-service that will provide the service for the specific

device needs to be assigned to it.