

Multisite OpenStack for NFV

Bridging The Gap

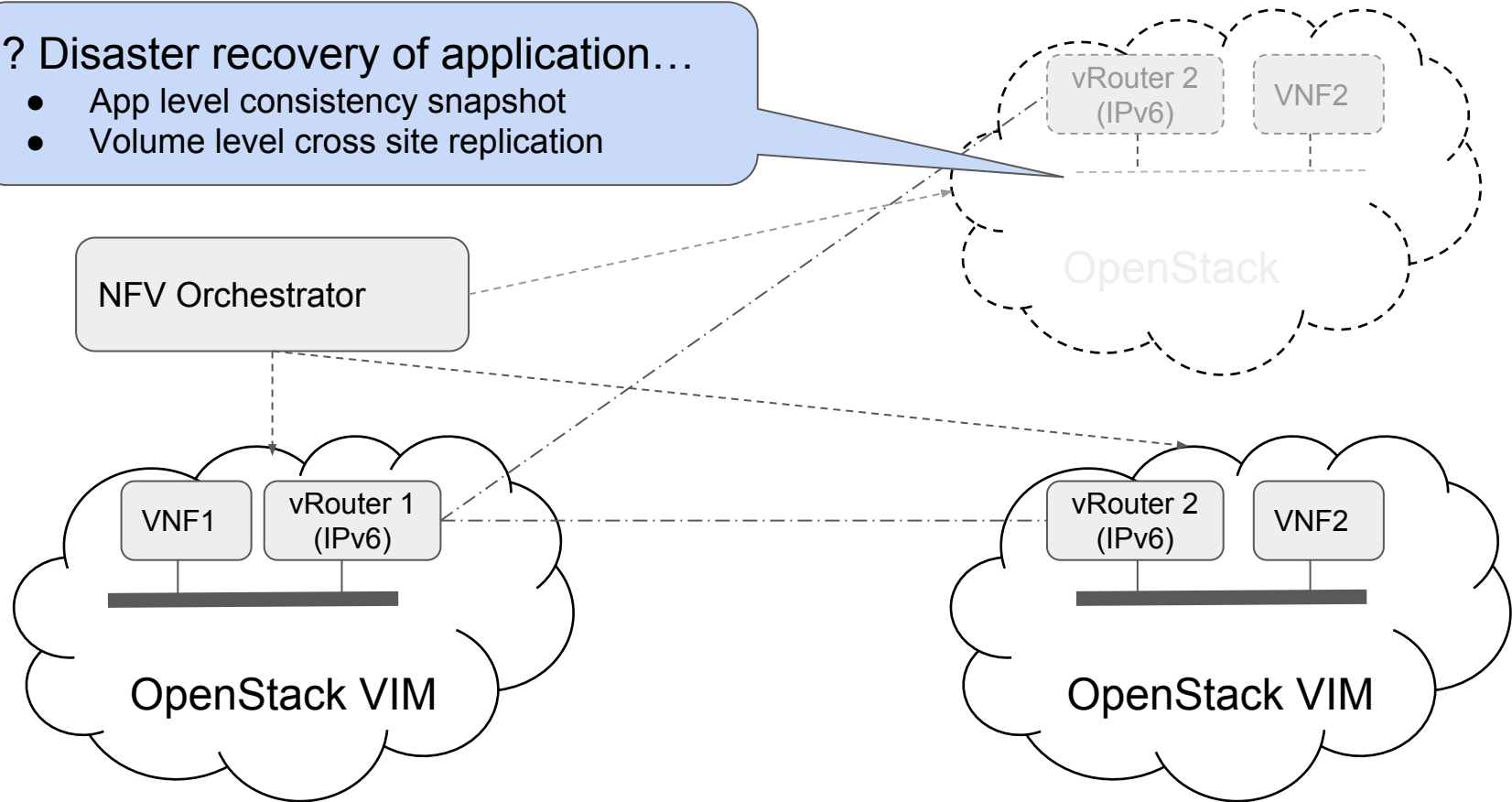
Agenda

- Gaps in multi-site OpenStack for NFV
- Tacker
- Kingbird
- Tricircle
- Putting it all together

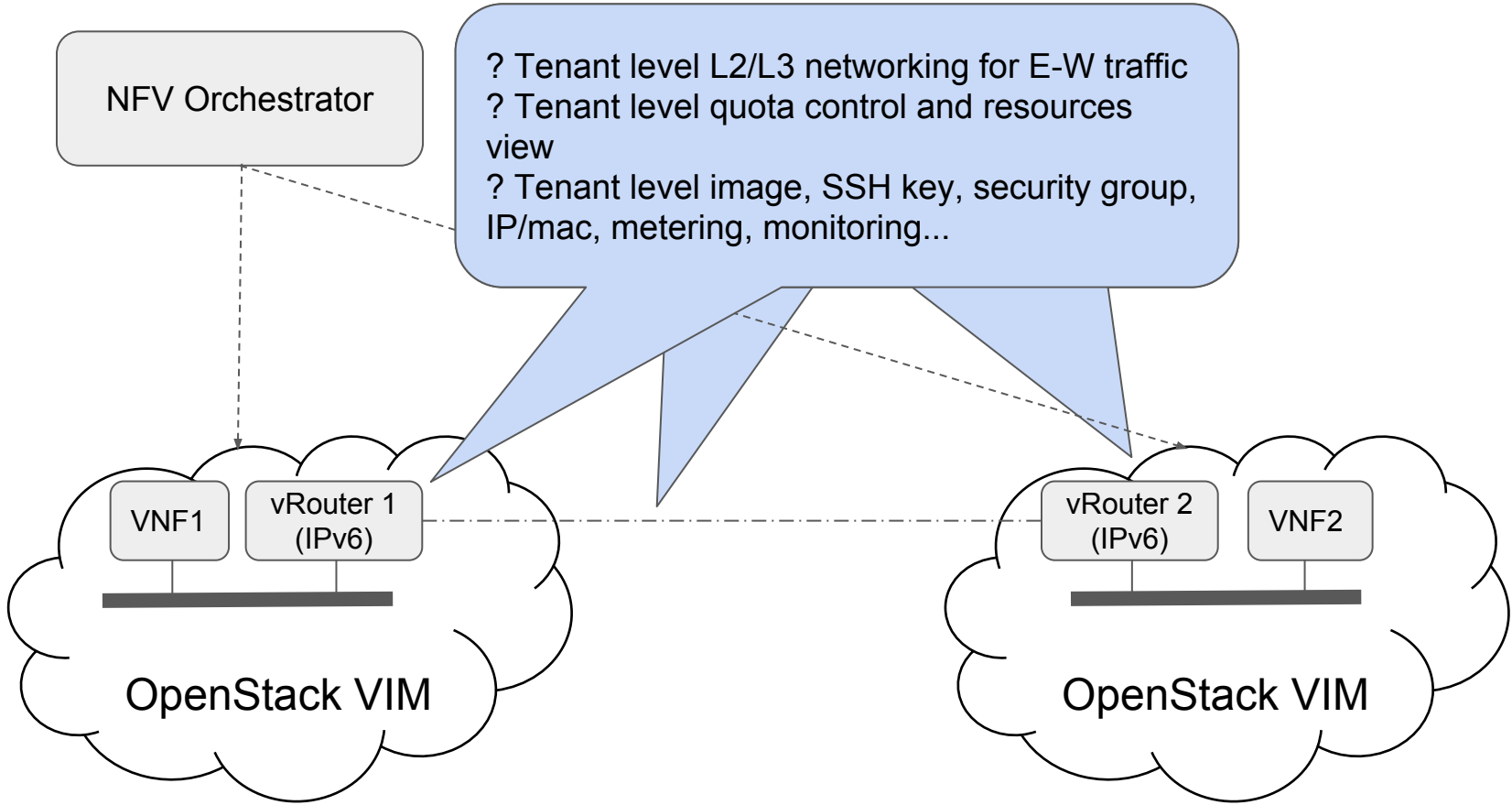
Gaps in Multi-site OpenStack for NFV

? Disaster recovery of application...

- App level consistency snapshot
- Volume level cross site replication



Gaps in Multi-site OpenStack for NFV

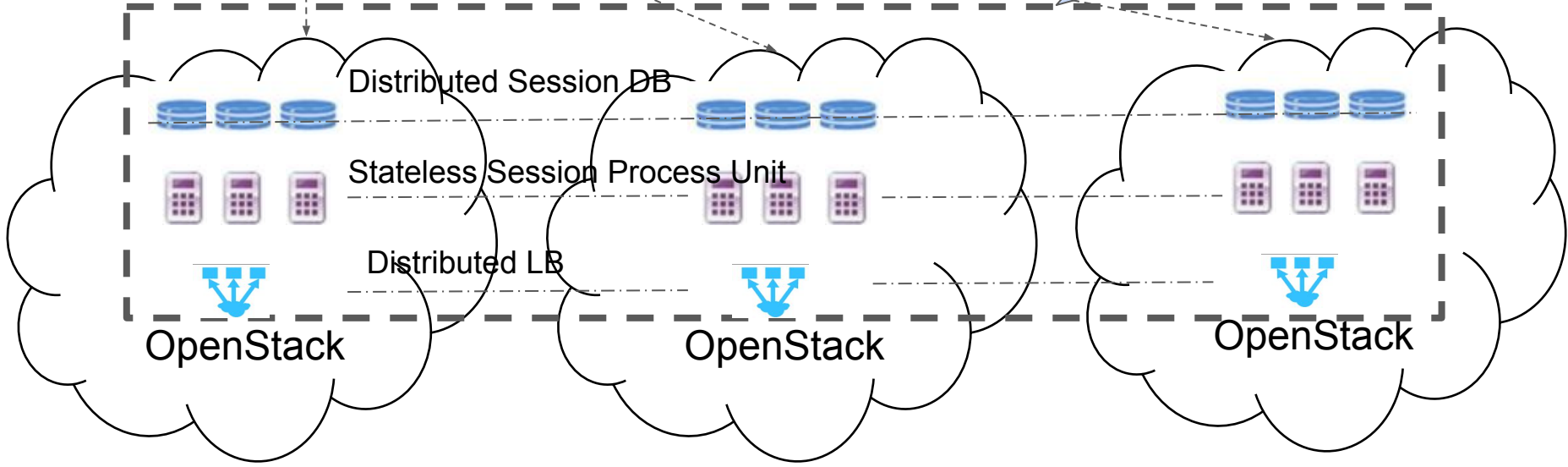


Gaps in Multi-site OpenStack for NFV

NFV Orchestrator

? Tenant level L2/L3 networking for E-W traffic
? Tenant level quota control and resources view
? Tenant level image, SSH key, security group, IP/mac, metering, monitoring...

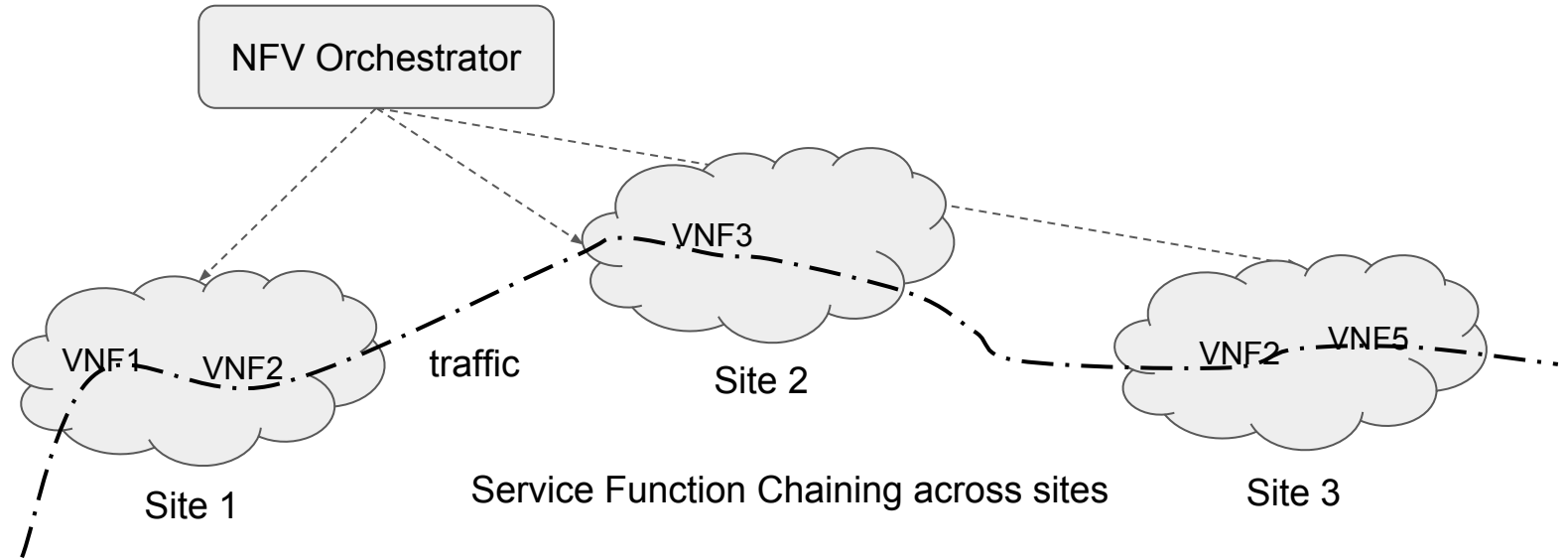
Distributed VNF



OPNFV Multisite Use Cases

| | | |
|------------|---|--|
| Use case 1 | Multisite identity service management | <ul style="list-style-type: none">● Bug reported to Keystone (Liberty) |
| Use case 2 | VNF high availability across VIM | <ul style="list-style-type: none">● Cross OpenStack L2 networking requirements to L2GW (Newton) |
| Use case 3 | Multisite VNF Geo-Site disaster recovery | <ul style="list-style-type: none">● Group VMs consistency snapshot via quiesce/unquiesce API (Newton)● Volume level replication |
| Use case 4 | Resources management and synchronization. | <ul style="list-style-type: none">● Kingbird● Tricircle (for different deployment scenario) |
| Use case 5 | Centralized monitoring service. | |

End-to-end Service Orchestration



Monitor and heal service chain in case of VNF failures
Modify/update service chains
Resilient to WAN bandwidth and delay

Tacker Project Overview

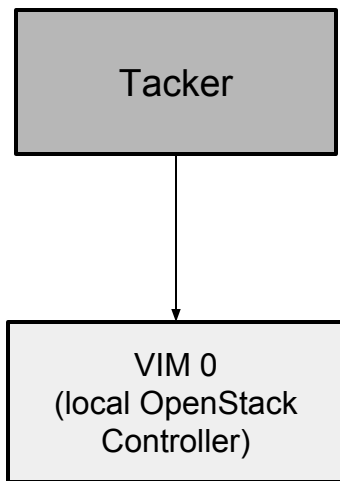
- Open NFV Orchestrator
- VNF life cycle management
 - monitoring framework
 - configuration framework
- VNF catalog management
 - TOSCA template support
- EPA support
 - CPU-Pinning, Huge Page, NUMA awareness, SR-IOV
- Auto resource creation
 - Flavor, Network and Image creation

Multisite VIM Support

- Unified view of VIM management using a single dashboard
- Ease of deployment for operators to integrate Tacker with their existing OpenStack installations with minimal overhead
- Fulfillment of a certain level of RO functionality, by combining necessary existing OpenStack modules such as heat and keystone

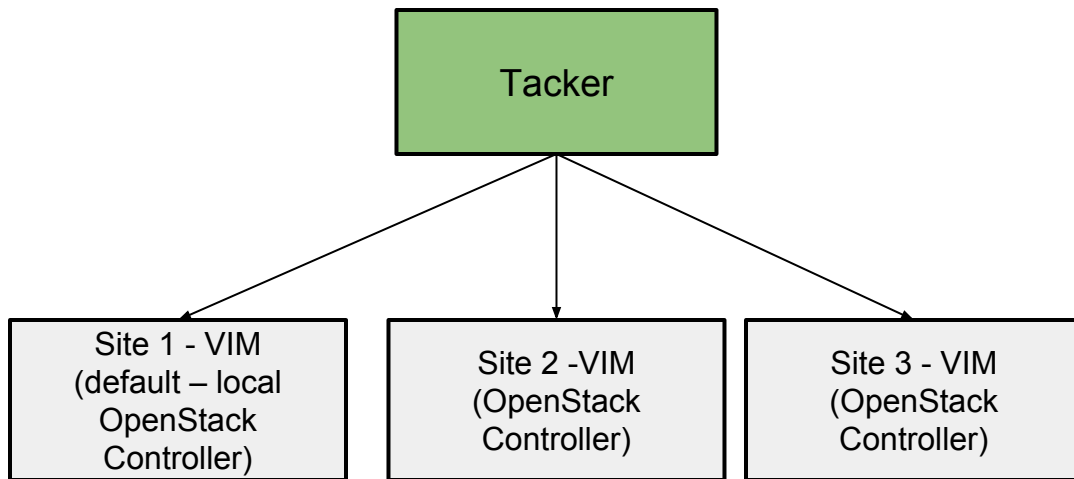
Unified View of VIM Management

Liberty



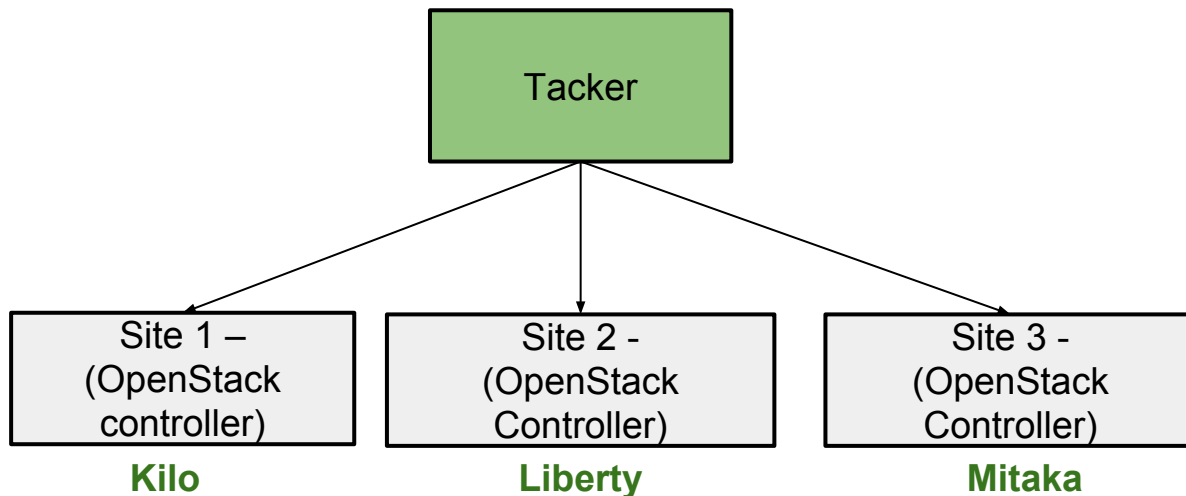
Tacker can instantiate VNFs only on same OpenStack controller

Mitaka



VNF placement on specific target OpenStack VIM
Explicit Region support

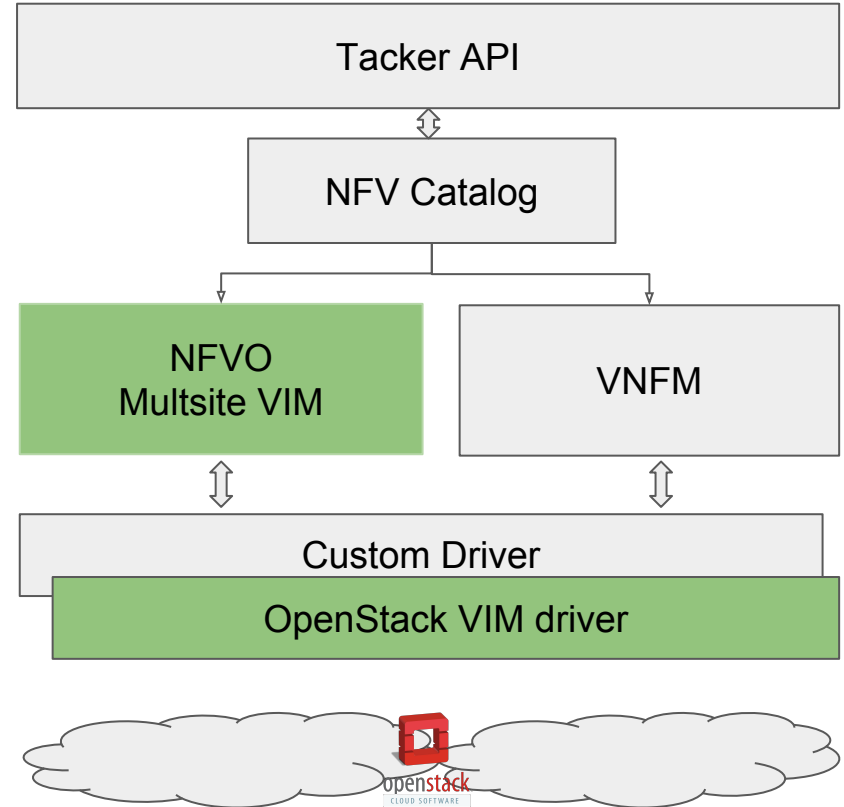
Multiple OpenStack Versions



- Operators have multiple OpenStack VIMs with different OpenStack versions
- Detect KeyStone and HOT template version and translate accordingly
- Gracefully downgrade resource requests on older VIMs

Tacker Multisite Architecture

- Multisite VIM module part of orchestrator component
- Pluggable driver framework
 - (aka Bring your own VIM)
- VIMs shared across tenants
- Horizon dashboard and CLI support



Tacker Resources

- <https://wiki.openstack.org/wiki/Tacker>
- <https://github.com/openstack/tacker>
- <https://github.com/openstack/tacker-specs/blob/master/specs/mitaka/multi-site-feature.rst>

Kingbird Project

Resource synchronization and management for multi-region OpenStack deployments.

Based on OPNFV Multisite UC5, UC6.

Aggregated view of distributed resources.

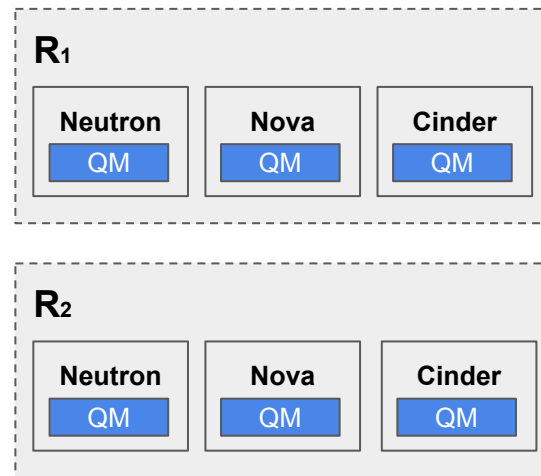
Synchronization (ssh-keys, security groups, images, flavors, etc.)

Centralized quota management.



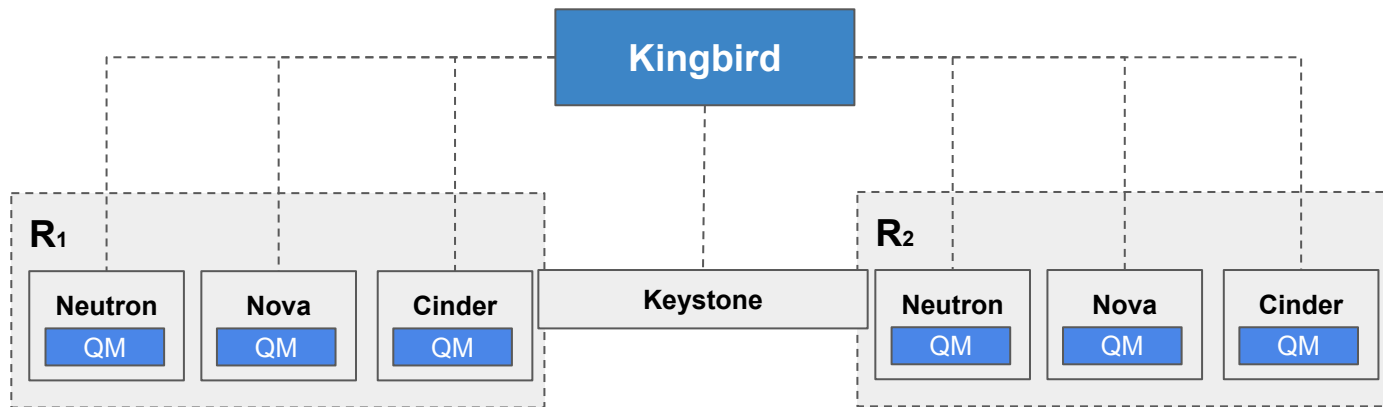
Quotas in OpenStack

- Defined on a per-region basis.
- Fragmented quota management in nova, neutron, cinder.
- Assigned per region.
 - E.g. max 50 instances in R1, max 30 instances in R2
- No process for synchronizing allocated quotas across regions.



Kingbird Quota Management

- New centralized quota management function.
- Global quota limits across multiple regions.
- Minimal impact on the existing OpenStack services.



Kingbird Quota Management

- Use existing APIs to dynamically balance quota values.
- Calculate resource usage upon synchronization.
- Store the default/tenant quota limits in Kingbird.
 - Provide CRUD operations for the known quota limits
 - Expose similar APIs (os-quota-set, os-quota-class-sets)

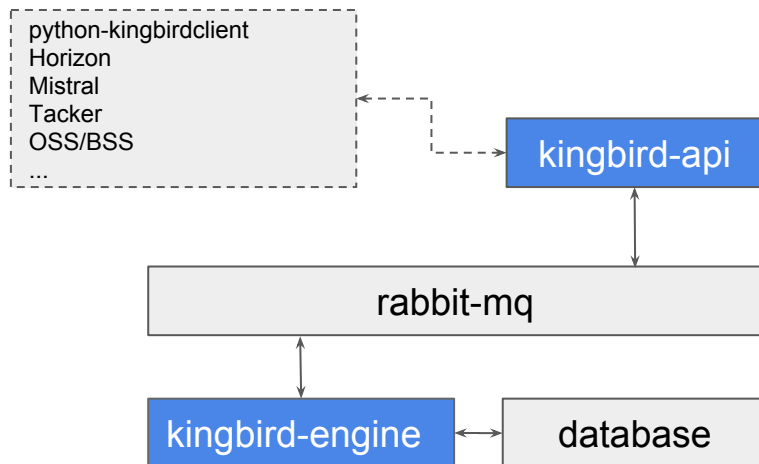
Kingbird Architecture

kingbird-api

Provides an API for managing global quota limits and on-demand quota synchronization.

kingbird-engine

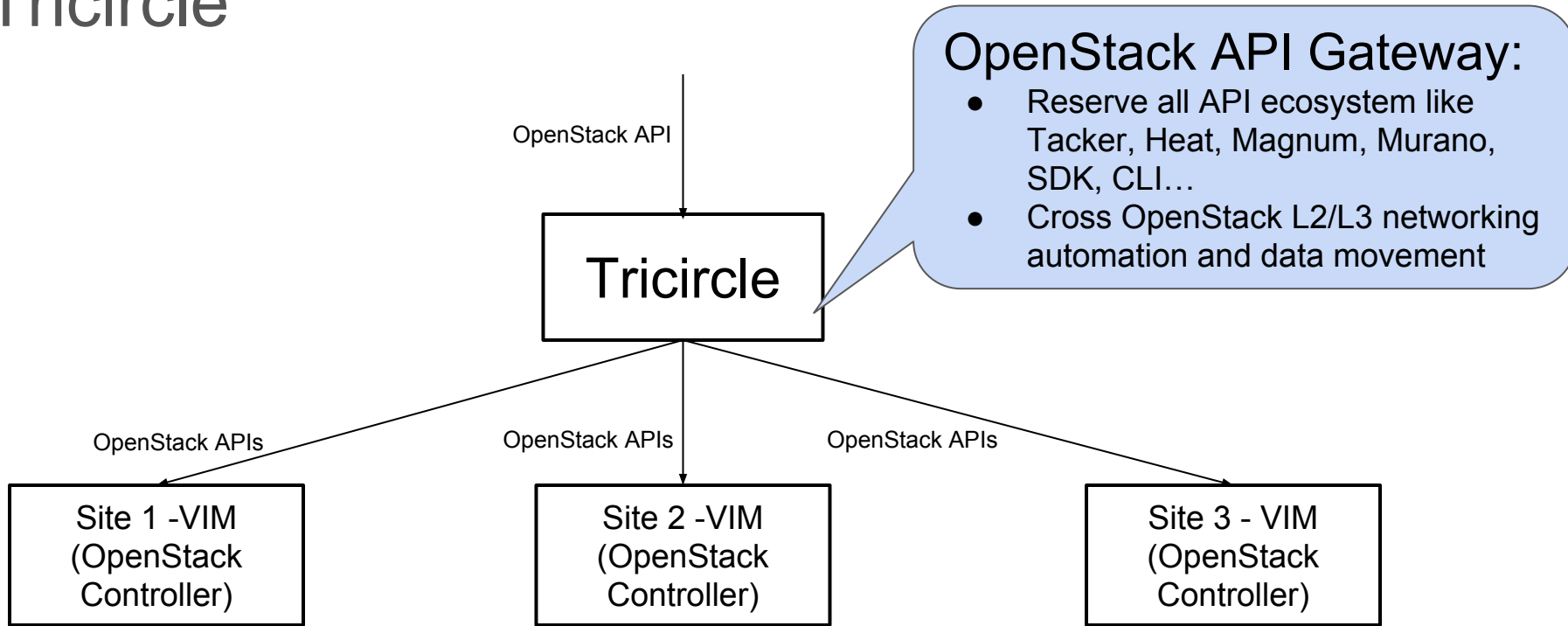
Responsible for communicating with OpenStack services in regions, fetching tenant resource usages, periodically rebalancing quota limits.



Kingbird Resources

- Project status, bugs, and blueprints are tracked on Launchpad
<https://launchpad.net/kingbird>
- Source code on Github
<https://github.com/openstack/kingbird/>

Tricircle



OpenStack API Gateway:

- Reserve all API ecosystem like Tacker, Heat, Magnum, Murano, SDK, CLI...
- Cross OpenStack L2/L3 networking automation and data movement

Tricircle

OpenStack APIs

Site 1 -VIM
(OpenStack
Controller)

OpenStack APIs

Site 2 -VIM
(OpenStack
Controller)

OpenStack APIs

Site 3 - VIM
(OpenStack
Controller)

Tricircle is OpenStack API gateway with added value like cross OpenStack L2/L3 networking, volume/VM movement, image distribution, global resource view, distributed quota management ...

This makes massive distributed edge clouds work like one inter-connected cloud, one OpenStack

Tricircle

OpenStack API

Tricircle

OpenStack API Gateway:

- Tenant level L2/L3 networking automation for E-W traffic isolation

Site 2 - VIM
(OpenStack
Controller)

Site 2 - VIM
(OpenStack
Controller)

Site 3 - VIM
(OpenStack
Controller)

VM1

vRouter 1

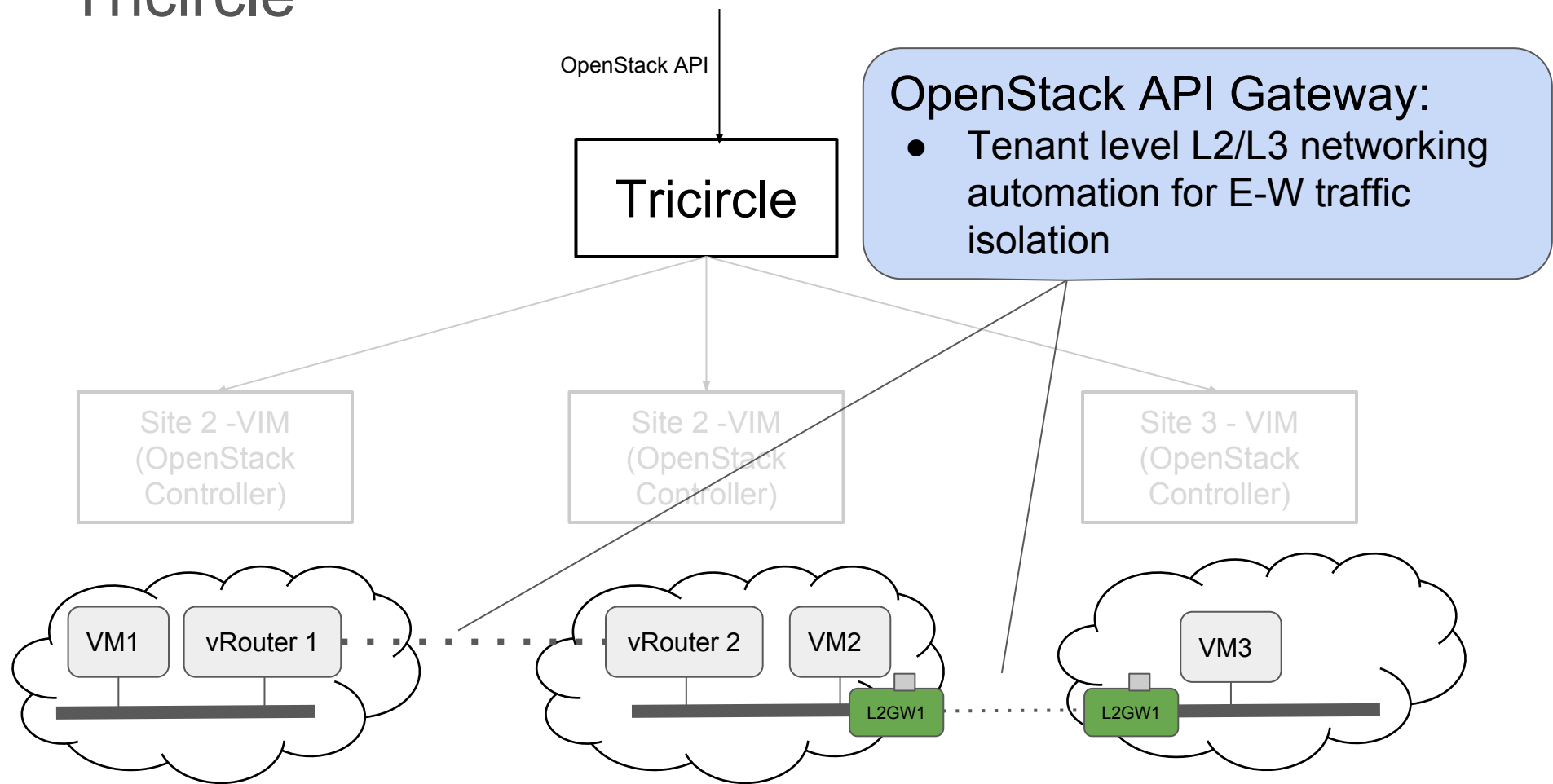
vRouter 2

VM2

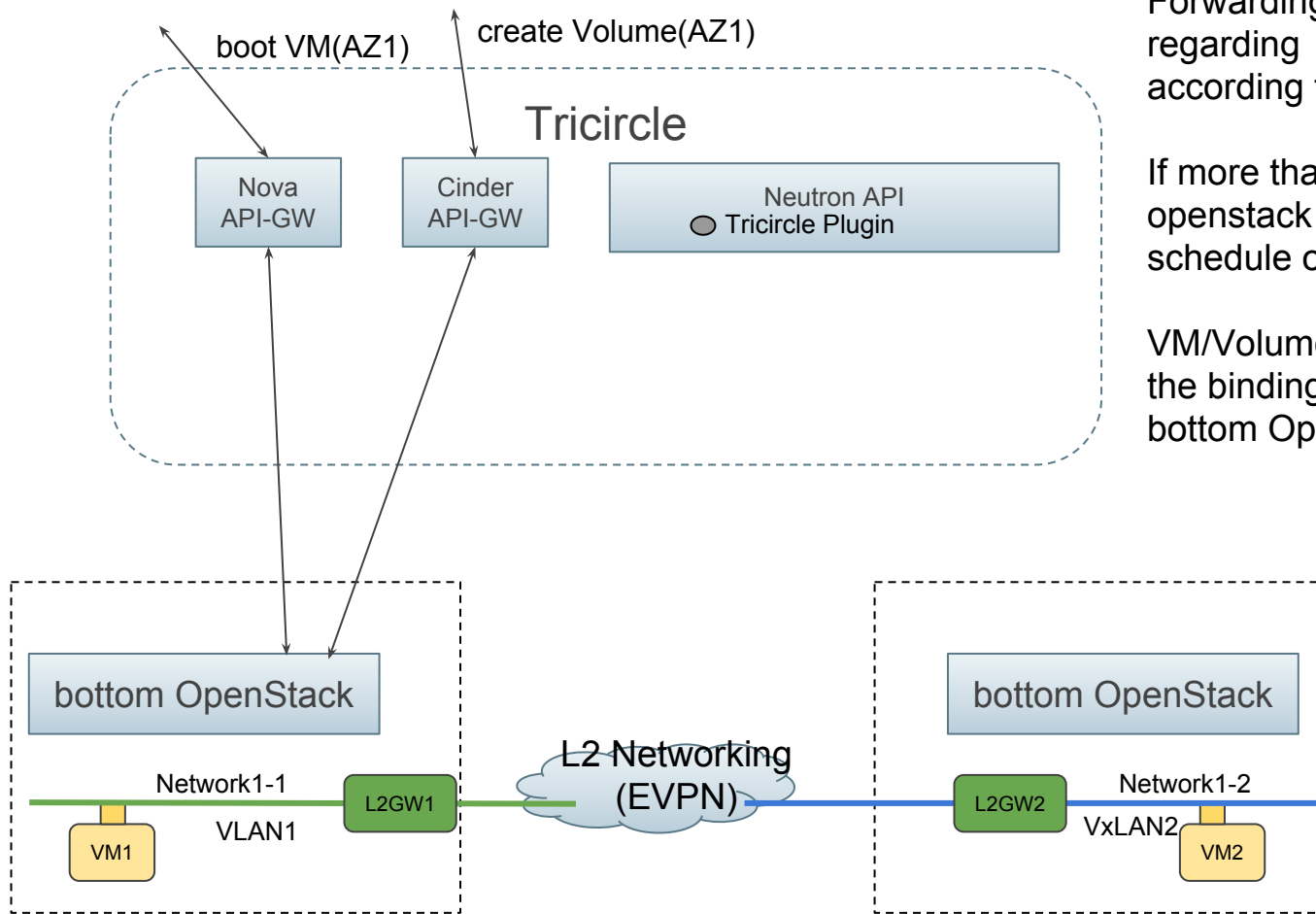
L2GW1

VM3

L2GW1



Tricircle-request forwarding

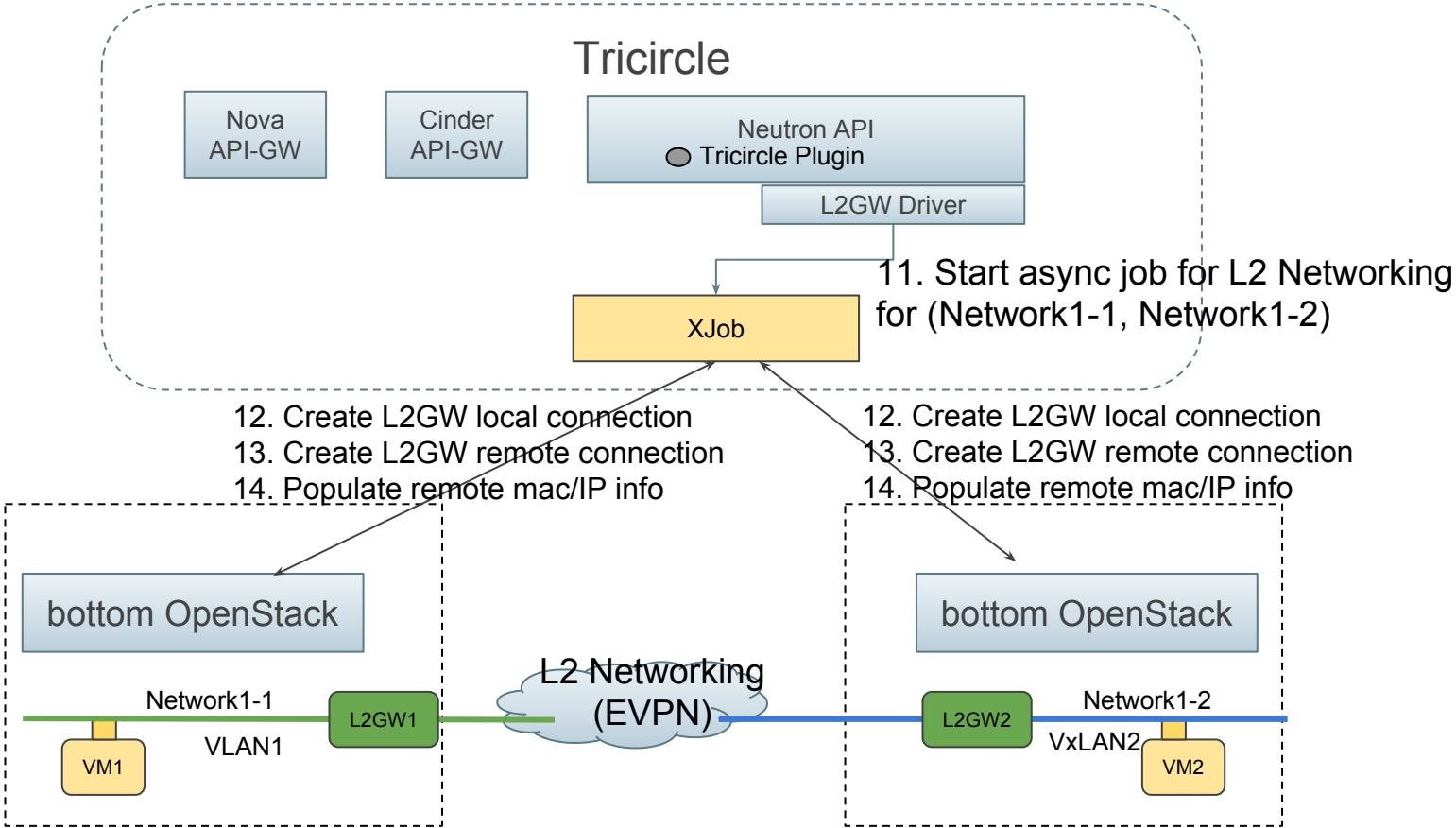


Forwarding request to regarding bottom OpenStack according to AZ.

If more than one bottom openstack in one AZ, then schedule one.

VM/Volume co-location through the binding of tenant-id and bottom OpenStack

Tricircle-extend the network to where the VM is, automatically



Tricircle

OpenStack APIs

Tricircle

OpenStack API Gateway:

- Move tenant's data (VM, Volume, Image, etc) across site leverage the cross site tenant L2/L3 networking

Site 2 - VIM
(OpenStack
Controller)

Site 2 - VIM
(OpenStack
Controller)

Site 3 - VIM
(OpenStack
Controller)

VM1
(Trans
Tool)

volume

VM2
(Trans
Tool)

volume

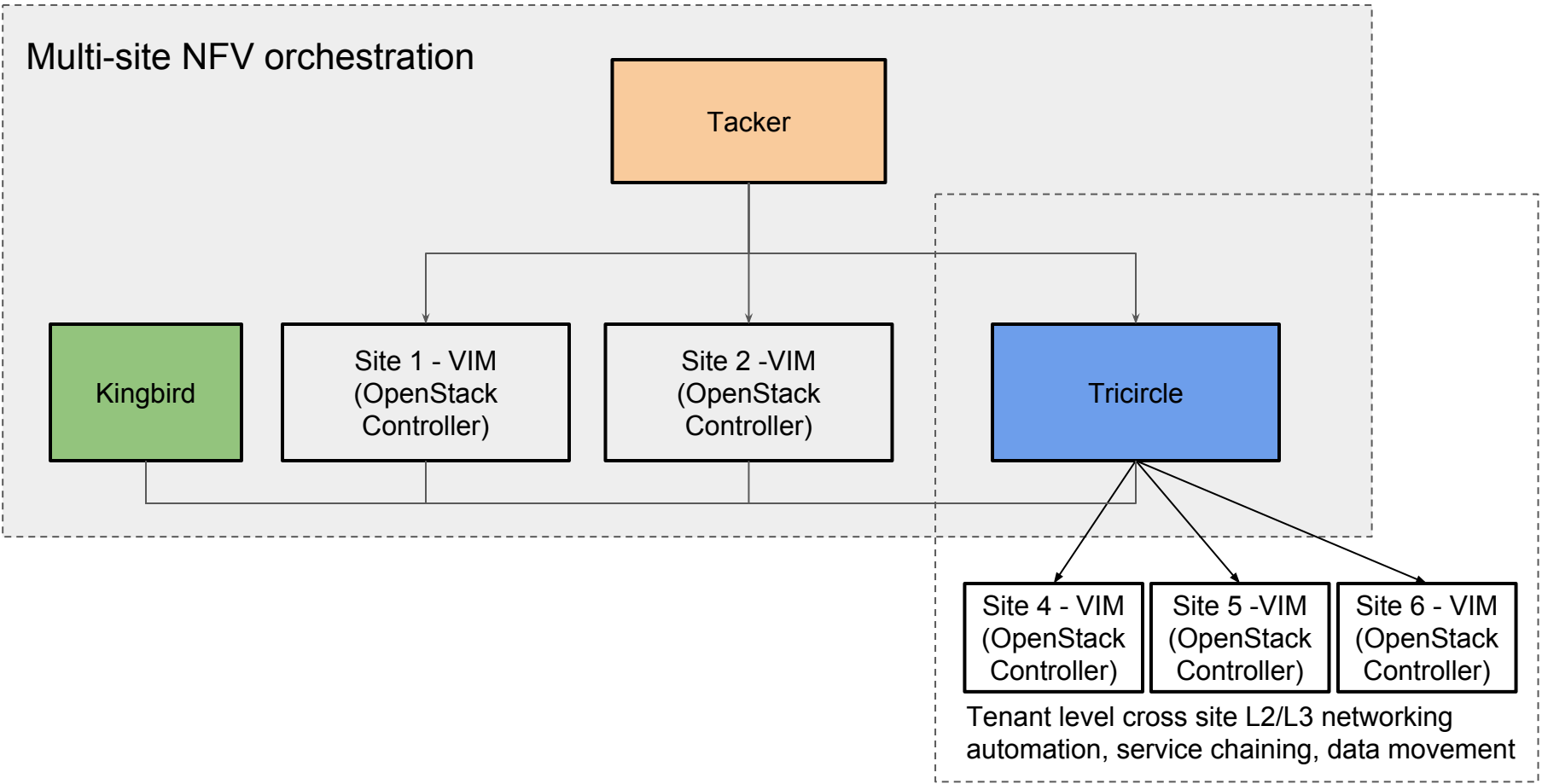
Create VM with transportation tool, and attach the volume(data to be moved) to the VM, move the data across OpenStack through tenant level L2/L3 networking.

*Conveyor, a project built above Tricircle will help to do this:<https://launchpad.net/conveyor>

Tricircle Resources

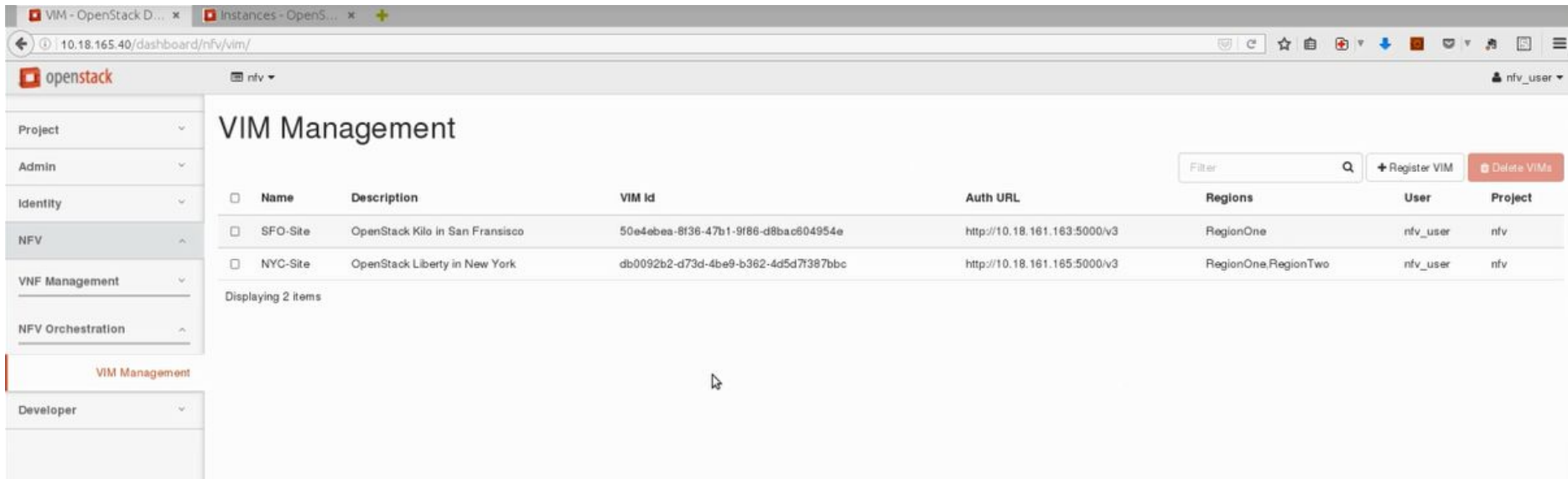
- Project status, bugs, and blueprints are tracked on Launchpad
<https://launchpad.net/tricircle>
- Source code on Github
<https://github.com/openstack/tricircle>

Putting it all together



Thanks, Q&A

Tacker Multisite Horizon Workflow



The screenshot displays the OpenStack Horizon interface for VIM Management. The browser address bar shows the URL `10.18.165.40/dashboard/nfv/vim/`. The page title is "VIM Management". On the left, a navigation sidebar includes "Project", "Admin", "Identity", "NFV", "VNF Management", "NFV Orchestration", "VIM Management" (highlighted), and "Developer". The main content area features a table with columns: Name, Description, VIM Id, Auth URL, Regions, User, and Project. Two VIMs are listed: "SFO-Site" and "NYC-Site". Above the table, there is a search filter, a "+ Register VIM" button, and a "Delete VIMs" button. Below the table, it indicates "Displaying 2 items".

| <input type="checkbox"/> | Name | Description | VIM Id | Auth URL | Regions | User | Project |
|--------------------------|----------|---------------------------------|--------------------------------------|------------------------------|---------------------|----------|---------|
| <input type="checkbox"/> | SFO-Site | OpenStack Kilo in San Francisco | 50e4ebea-8f36-47b1-9f86-d8bac804954e | http://10.18.161.163:5000/v3 | RegionOne | nfv_user | nfv |
| <input type="checkbox"/> | NYC-Site | OpenStack Liberty in New York | db0092b2-d73d-4be9-b362-4d5d7f387bbc | http://10.18.161.165:5000/v3 | RegionOne,RegionTwo | nfv_user | nfv |

Tacker Multisite Horizon Workflow

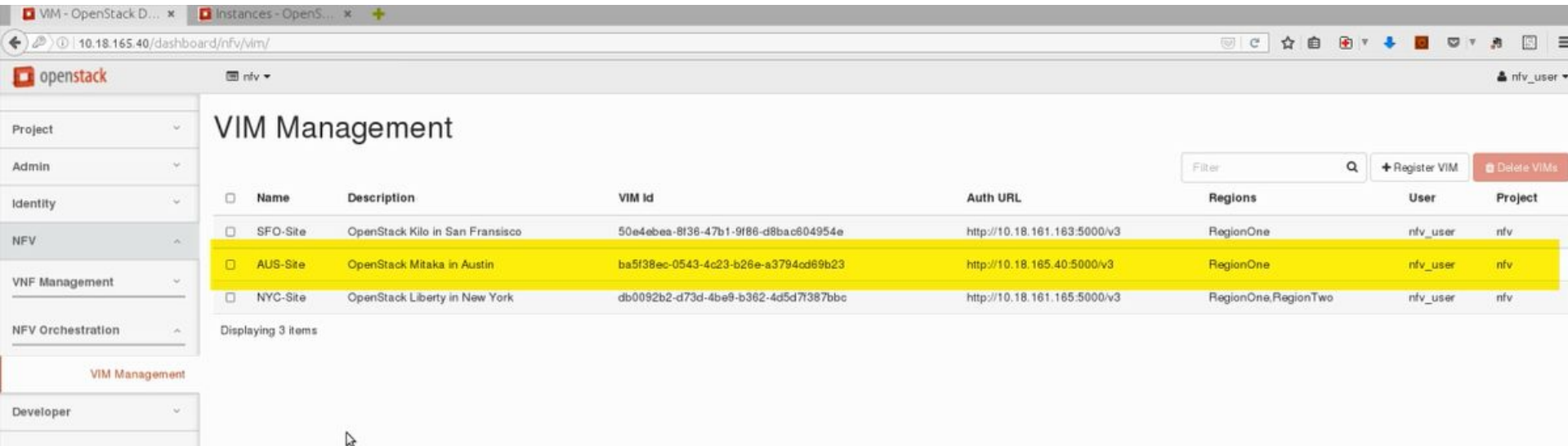
The screenshot displays the OpenStack Horizon interface for VIM Management. A modal dialog titled "Register VIM" is open, allowing the user to register a new VIM. The dialog contains the following fields:

- Name:** AUS-Site
- Description:** OpenStack Mitaka in Austin
- Auth URL:** http://10.18.165.40:5000
- Username:** nfv_user
- Password:** (masked with asterisks)
- Project Name:** nfv

The background interface shows the "VIM Management" section with a table of existing VIMs:

| Regions | User | Project |
|---------------------|----------|---------|
| RegionOne | nfv_user | nfv |
| RegionOne,RegionTwo | nfv_user | nfv |

Tacker Multisite Horizon Workflow



The screenshot shows the OpenStack Horizon VIM Management interface. The browser address bar indicates the URL is `10.18.165.40/dashboard/nfv/vim/`. The page title is "VIM Management". On the left, there is a navigation sidebar with categories like Project, Admin, Identity, NFV, VNF Management, and NFV Orchestration. The main content area displays a table of VIMs with columns for Name, Description, VIM Id, Auth URL, Regions, User, and Project. Three VIMs are listed: SFO-Site, AUS-Site, and NYC-Site. The AUS-Site row is highlighted in yellow. Below the table, it says "Displaying 3 items".

| <input type="checkbox"/> | Name | Description | VIM Id | Auth URL | Regions | User | Project |
|--------------------------|----------|---------------------------------|--------------------------------------|------------------------------|---------------------|----------|---------|
| <input type="checkbox"/> | SFO-Site | OpenStack Kilo in San Francisco | 50e4ebee-8f36-47b1-9f86-d8bac604954e | http://10.18.161.163:5000/v3 | RegionOne | nfv_user | nfv |
| <input type="checkbox"/> | AUS-Site | OpenStack Mitaka in Austin | ba5f38ec-0543-4c23-b26e-a37940d69b23 | http://10.18.165.40:5000/v3 | RegionOne | nfv_user | nfv |
| <input type="checkbox"/> | NYC-Site | OpenStack Liberty in New York | db0092b2-d73d-4be9-b362-4d5d7387bbc | http://10.18.161.165:5000/v3 | RegionOne,RegionTwo | nfv_user | nfv |

Displaying 3 items

Tacker Multisite Horizon Workflow

The screenshot displays the OpenStack VNF Manager web interface. A modal dialog titled "Deploy VNF" is open, allowing for the configuration of a new VNF deployment. The dialog includes fields for VNF Name, VNF Catalog Name, VIM Name, Region Name, Parameter Value Source, Configuration Value Source, and their respective files. A description on the right explains the purpose of the VNF and the effect of uploading a yamI file. The background shows the VNF Manager dashboard with a list of VNFs.

Deploy VNF

VNF Name:

VNF Catalog Name:

VIM Name:

Region Name:

Parameter Value Source:

Parameter Value File: No file selected.

Configuration Value Source:

Configuration Value File: tosca-config-openwrt-with-firewall.yamI

Description:
Deploys a VNF.
If the VNFD template is parameterized, upload a yamI file with values for those parameters.
If the VNFD template is not parameterized, any yamI file uploaded will be ignored.
If a configuration yamI file is uploaded, it will be applied to the VNF post its successful creation.

VNF Manager

Project: Project
Admin: Admin
Identity: Identity
NFV: NFV
VNF Management: VNF Management
VNF Catalog: VNF Catalog
VNF Manager
NFV Orchestration: NFV Orchestration
Developer: Developer

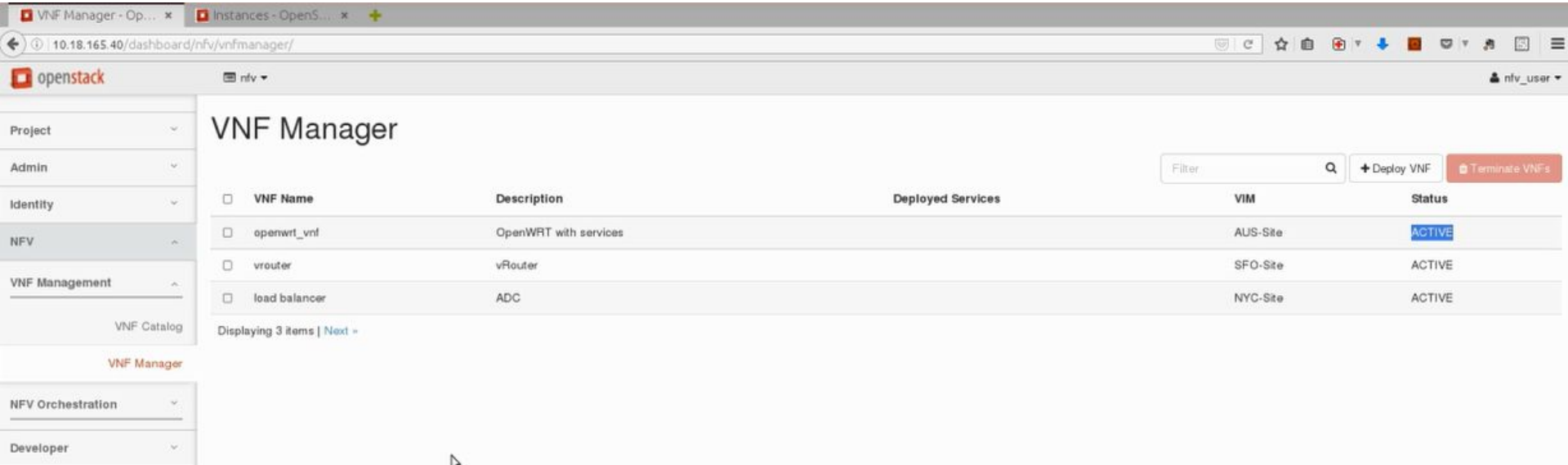
VNF Name: VNF Name
 router
 load balancer

Displaying 2 items | [Next](#)

Filter:

| VIM | Status |
|----------|--------|
| SFO-Site | ACTIVE |
| NYC-Site | ACTIVE |

Tacker Multisite Horizon Workflow



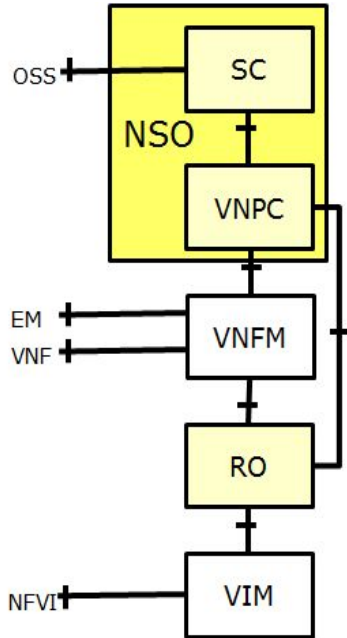
The screenshot shows the OpenStack VNF Manager interface. The browser address bar indicates the URL is `10.18.165.40/dashboard/nfv/vnfmanager/`. The OpenStack logo is visible in the top left, and the user is logged in as `nfv_user`. The main content area is titled "VNF Manager" and contains a table of VNFs. The table has columns for VNF Name, Description, Deployed Services, VIM, and Status. Three VNFs are listed: `openwrt_vnf` (OpenWRT with services, AUS-Site, ACTIVE), `vrouter` (vRouter, SFO-Site, ACTIVE), and `load balancer` (ADC, NYC-Site, ACTIVE). A search filter and buttons for "Deploy VNF" and "Terminate VNFs" are also present.

| <input type="checkbox"/> | VNF Name | Description | Deployed Services | VIM | Status |
|--------------------------|---------------|-----------------------|-------------------|----------|--------|
| <input type="checkbox"/> | openwrt_vnf | OpenWRT with services | | AUS-Site | ACTIVE |
| <input type="checkbox"/> | vrouter | vRouter | | SFO-Site | ACTIVE |
| <input type="checkbox"/> | load balancer | ADC | | NYC-Site | ACTIVE |

Displaying 3 items | Next >

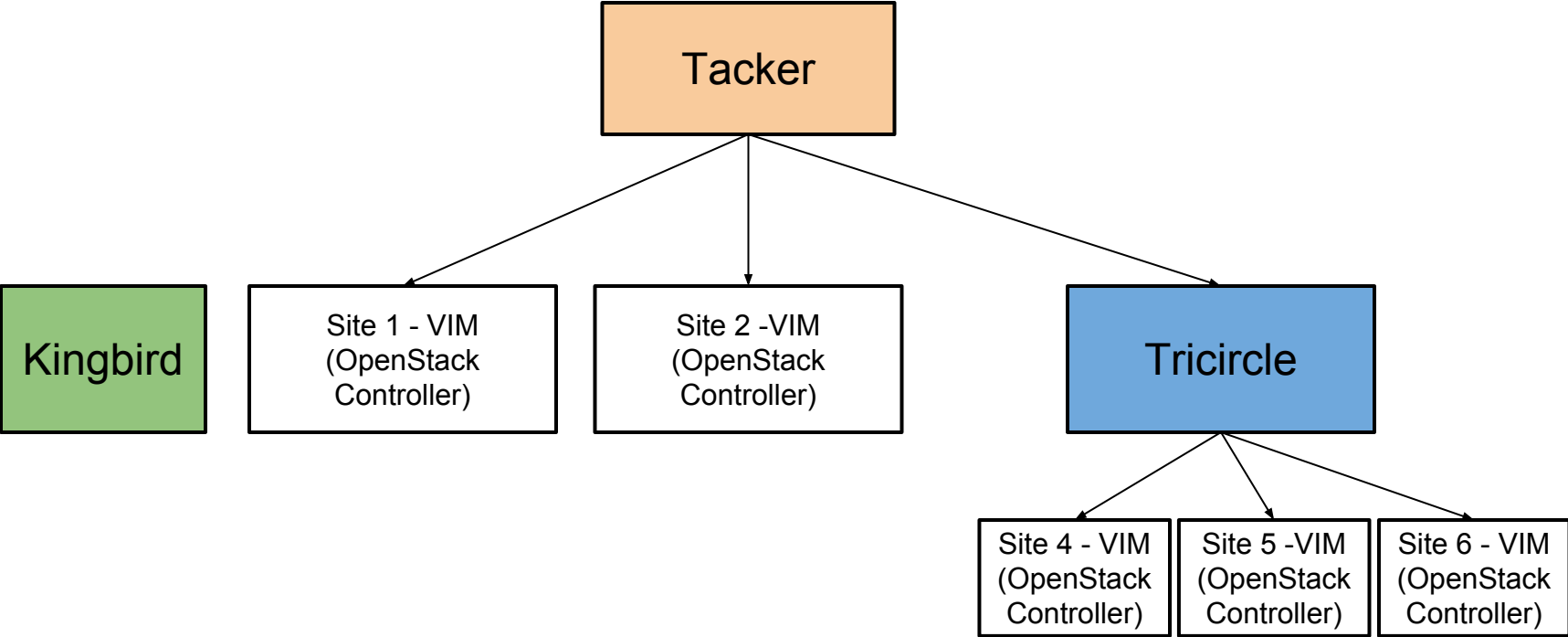
Need and Gap for Multi-site OpenStack in NFV (3)

A new work item ETSI NFV GS IFA-020 was approved on Apr 7th to provide a report on the NFVO Split, where a first time ever try for a holistic view on multiple site NFVO architecture will be discussed and introduced



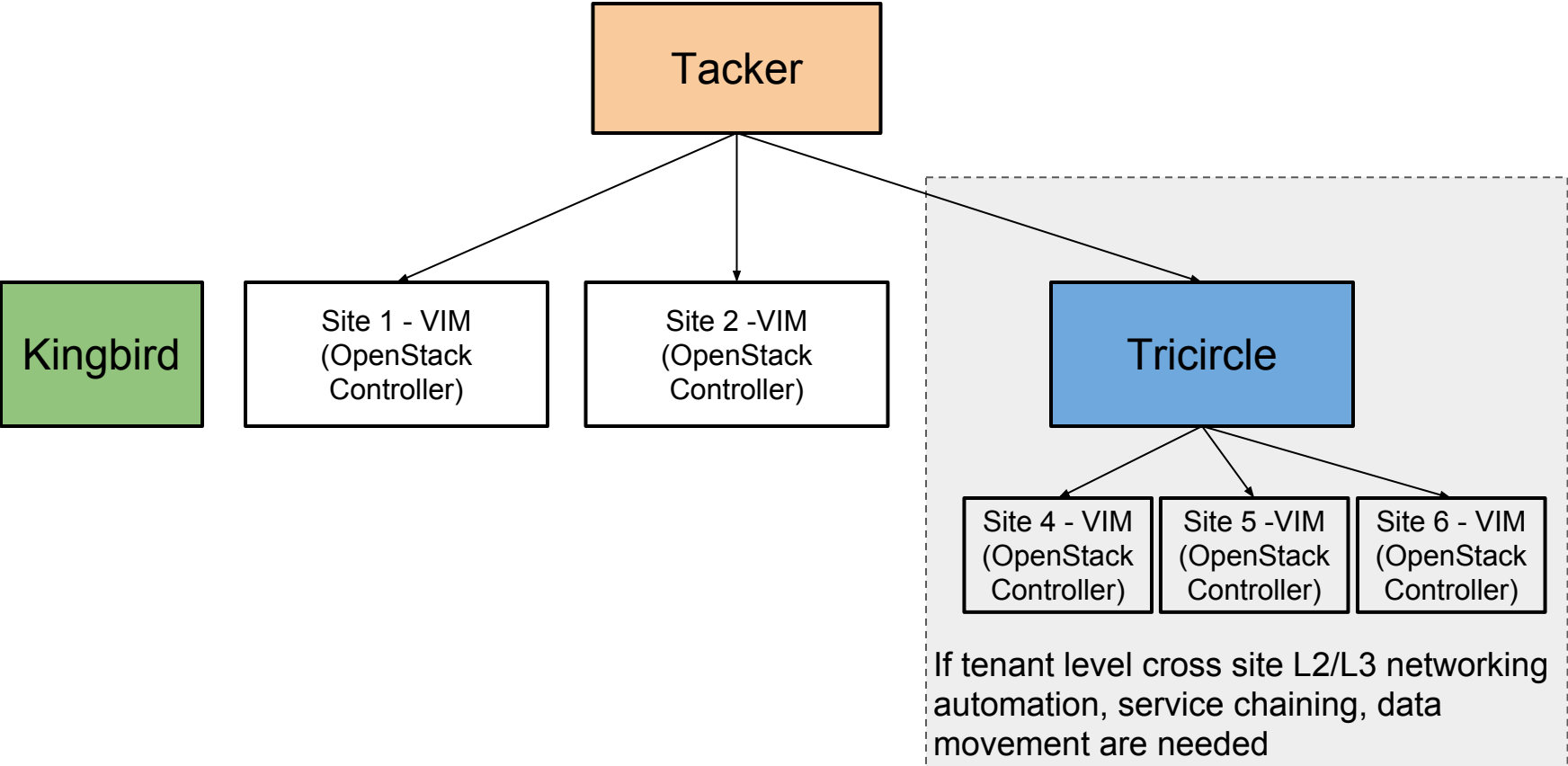
- ❑ The NFVO split architecture options should be designed to be complementary and not contradict each other
- ❑ Reference points should be reused as much as possible
- ❑ A potential option is shown on the left
- ❑ **In this context, OpenStack would be expected to provide certain level of RO functionality**
 - ❑ **Could be a combination of modules**
 - ❑ **Could be a not full-fledged RO entity**

Gaps in these solutions and what different teams are doing to fix it

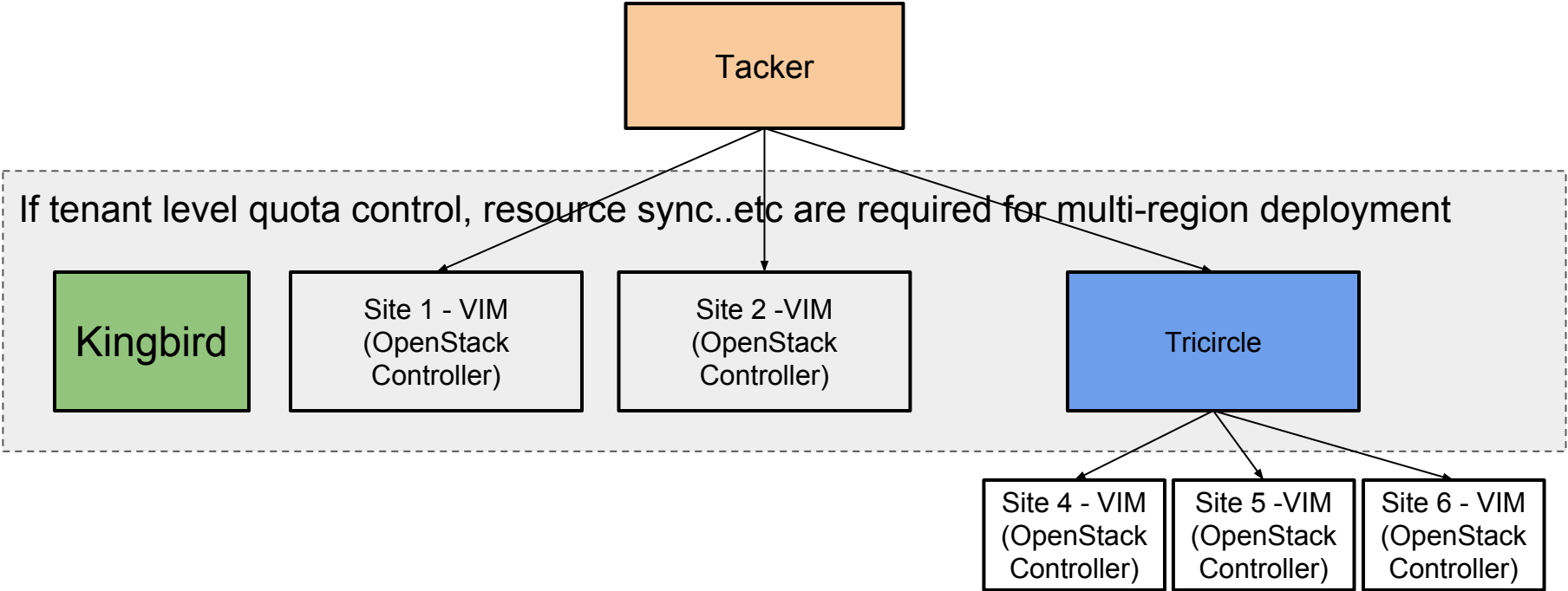


Landscape for projects working together

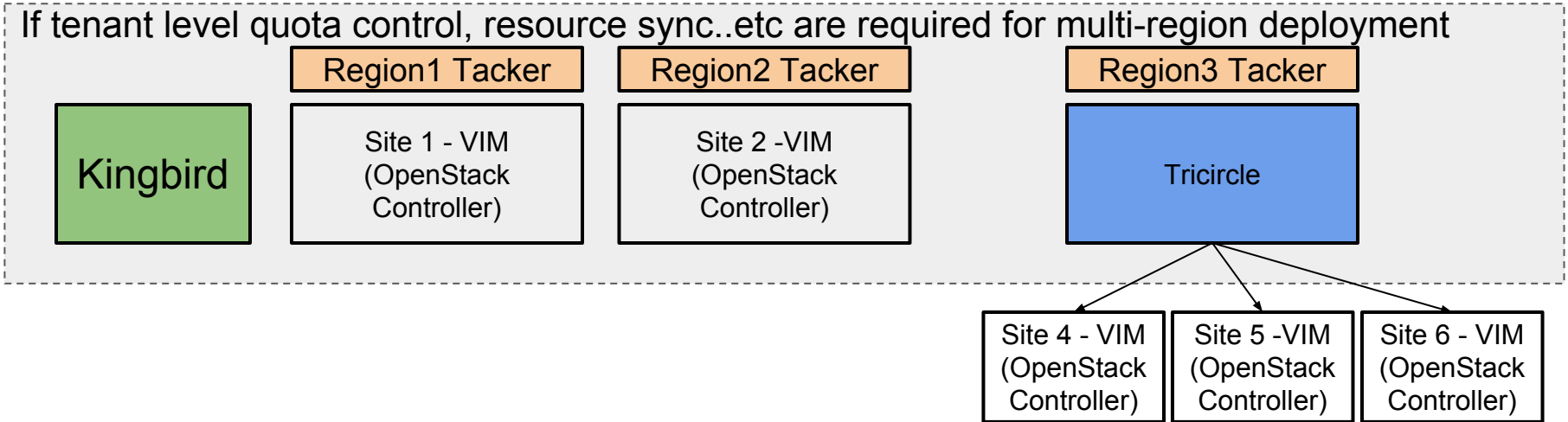
Gaps in these solutions and what different teams are doing to fix it



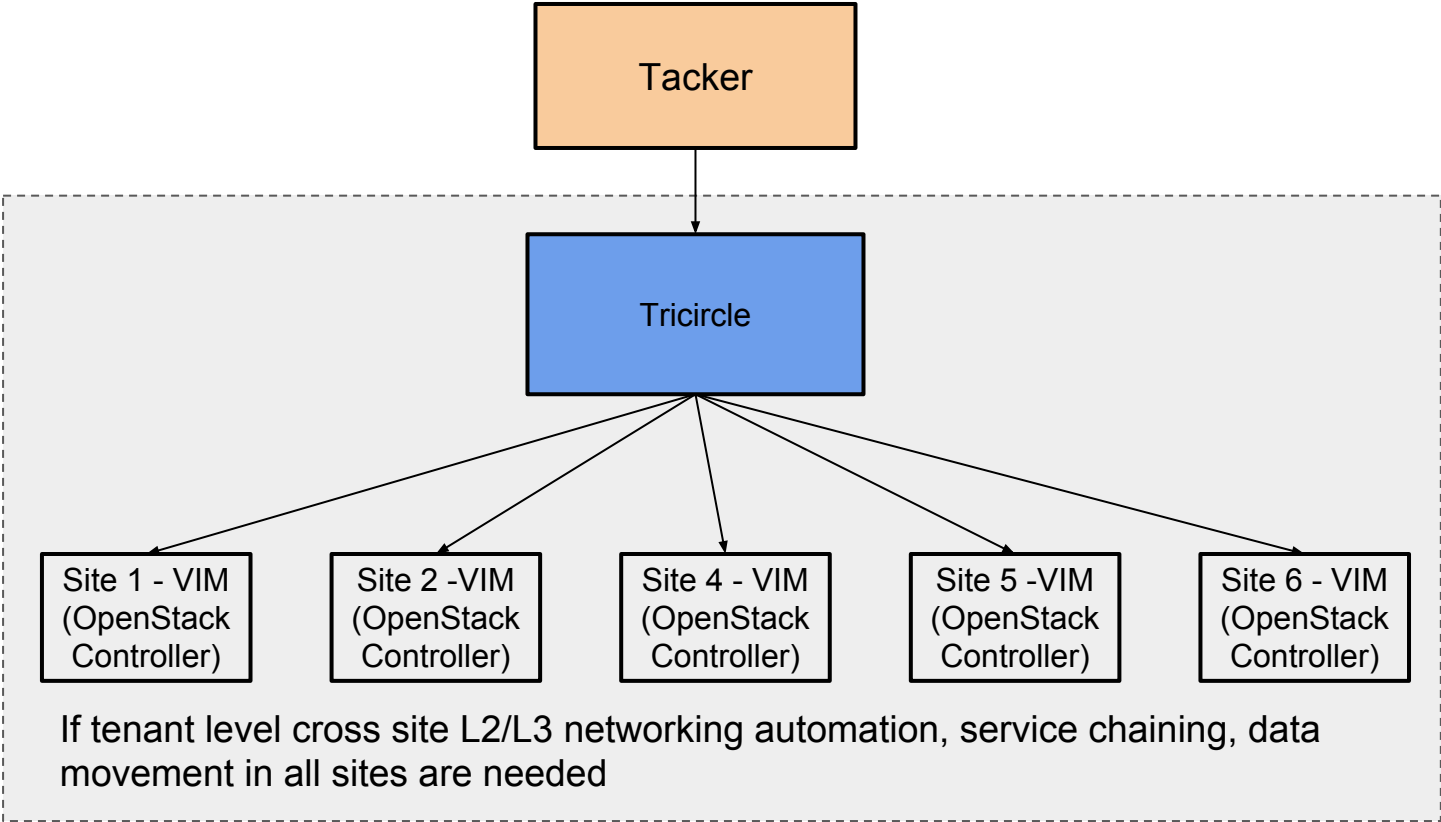
Gaps in these solutions and what different teams are doing to fix it



Gaps in these solutions and what different teams are doing to fix it



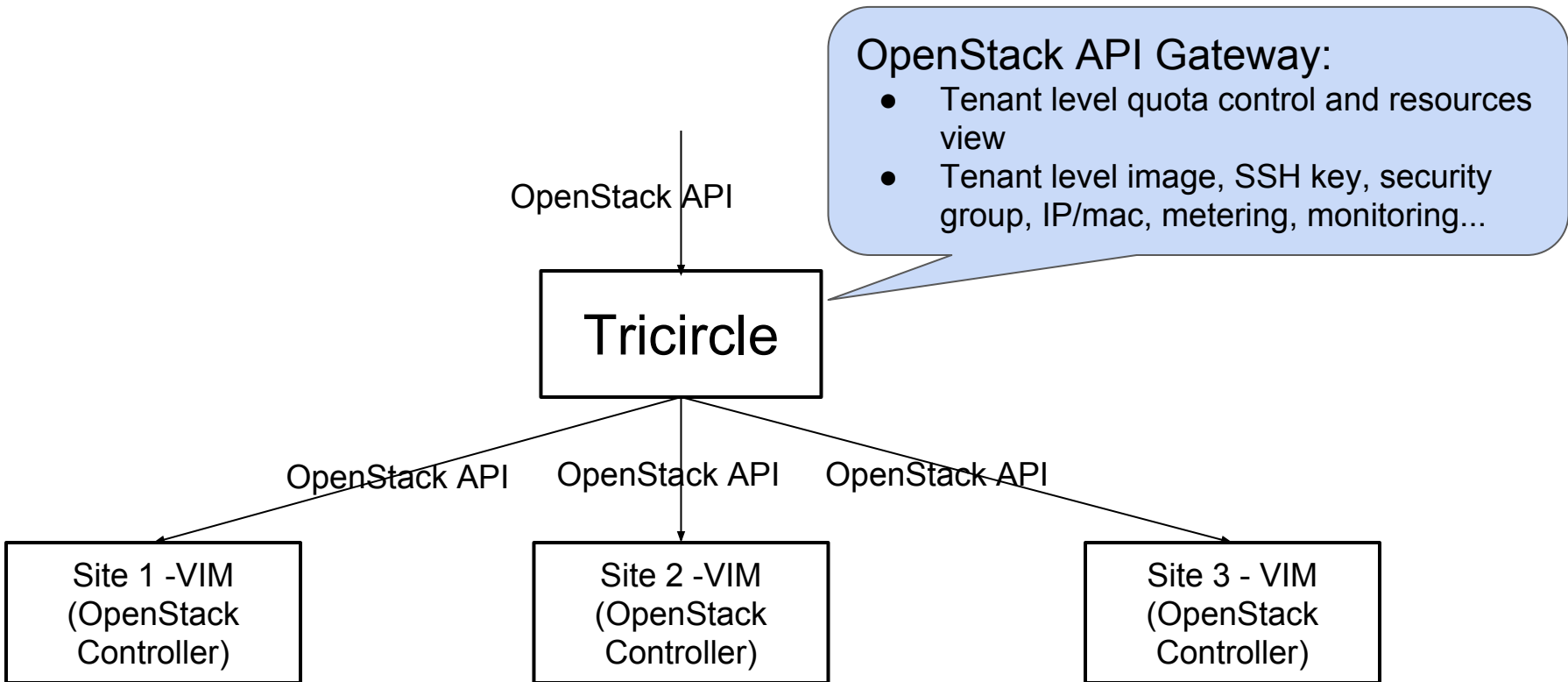
Gaps in these solutions and what different teams are doing to fix it



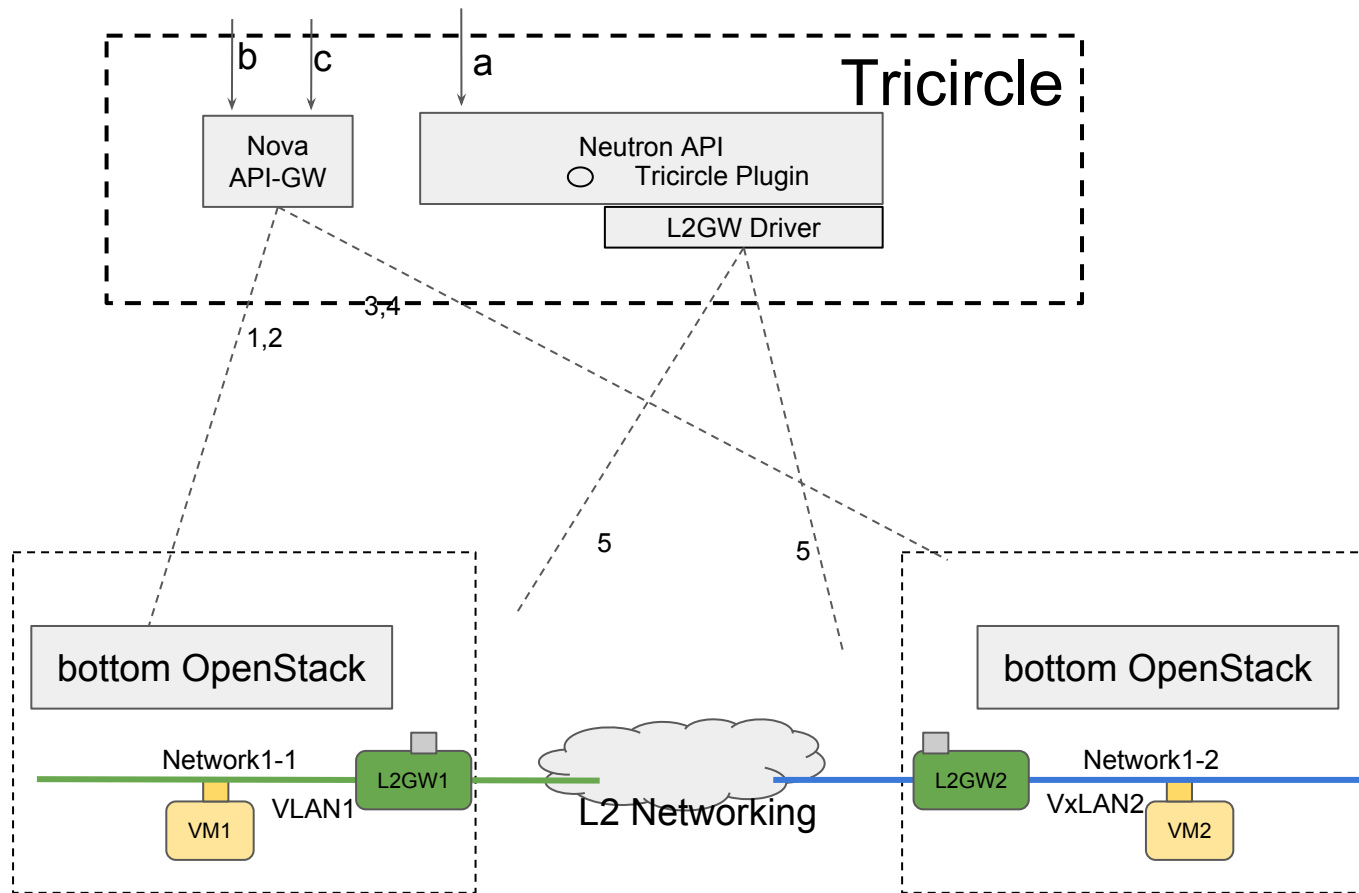
Existing projects and solutions

- Multisite
- Tacker
- Kingbird
- Tricircle

Tricircle



Tricircle



Quota sync algorithm

Three regions **A**, **B**, **C** and the global quota limit is **vcpus=10** set in *Kingbird*

S0: Initial state.

```
>+-----+-----+-----+-----+
>|      | A | B | C | Kb |
>+-----+-----+-----+-----+
>| s0  |   |   |   | 10 |
>+-----+-----+-----+-----+
```

S1: Kingbird syncs the limit in all regions:

```
>+-----+-----+-----+-----+
>|      | A | B | C | Kb |
>+-----+-----+-----+-----+
>| s0  |   |   |   | 10 |
>+-----+-----+-----+-----+
>| s1  | 10 | 10 | 10 | 10 |
>+-----+-----+-----+-----+
```

Quota sync algorithm

S2: Allocate 2 vcpus in A and 1 in C.

```
>+-----+-----+-----+-----+-----+-----+-----+-----+
>|      | A | B | C | Kb | Au | Bu | Cu | Su |
>+-----+-----+-----+-----+-----+-----+-----+-----+
>| S0  |   |   |   | 10 | 0 | 0 | 0 | 0 |
>+-----+-----+-----+-----+-----+-----+-----+
>| S1  | 10 | 10 | 10 | 10 | 0 | 0 | 0 | 0 |
>+-----+-----+-----+-----+-----+-----+-----+
>| S2  | 9  | 7  | 8  | 10 | 2 | 0 | 1 | 3 |
>+-----+-----+-----+-----+-----+-----+-----+

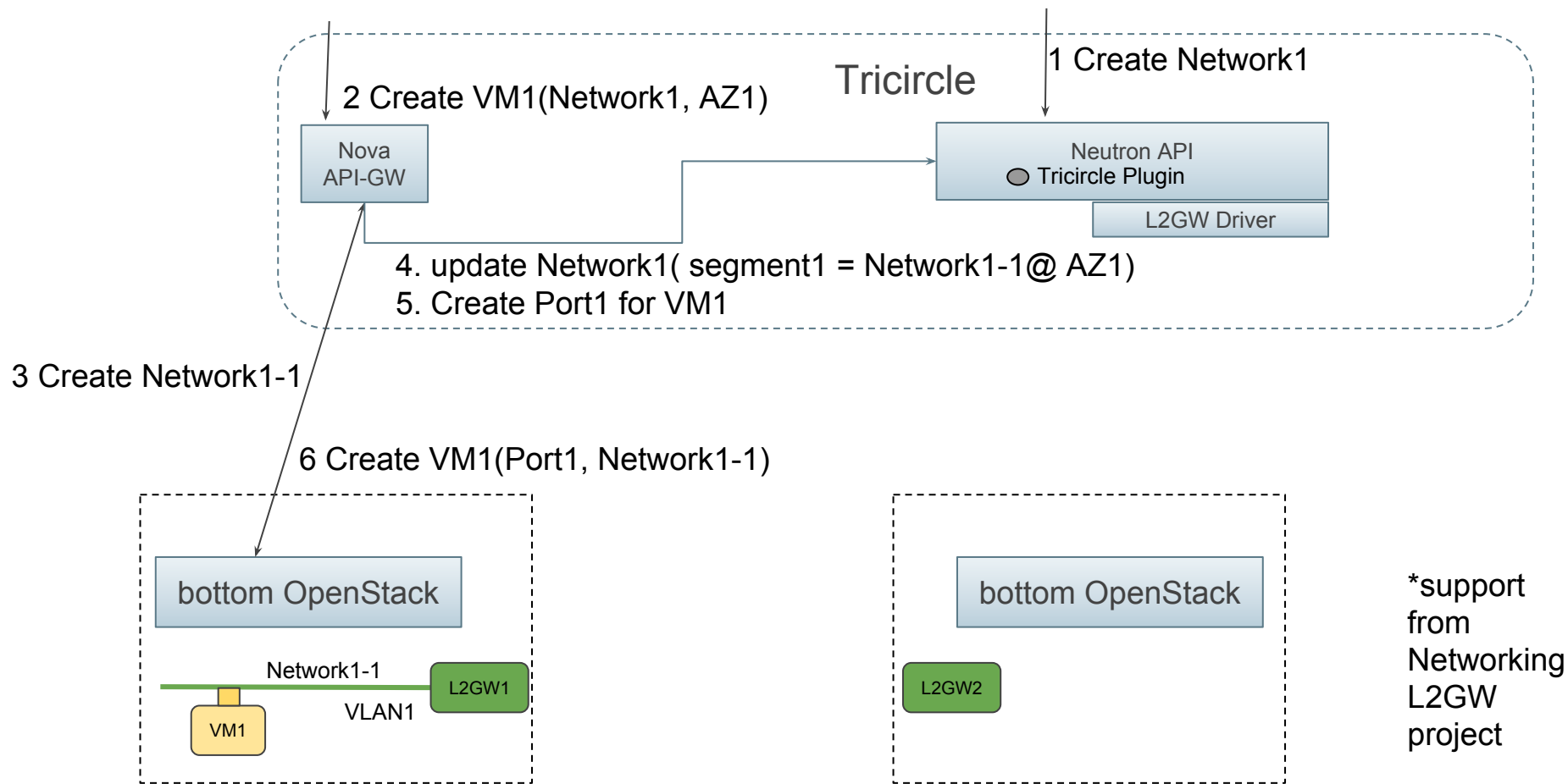
```

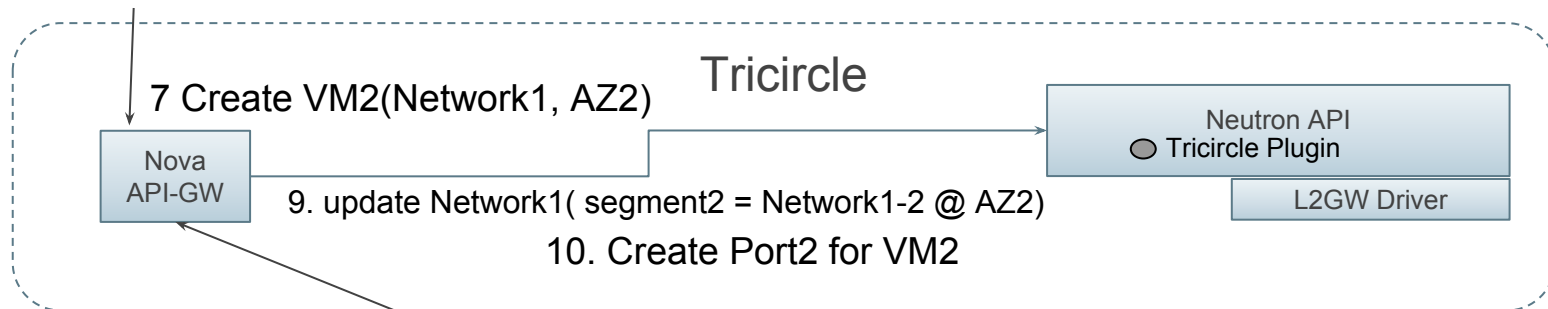
S3: Tenant allocates 2 more in A, 2 in B, and 2 more in C: so now the usage will be 4 in A, 2 in B, 3 in C (total 9).

```
>+-----+-----+-----+-----+-----+-----+-----+-----+
>|      | A | B | C | Kb | Au | Bu | Cu | Su |
>+-----+-----+-----+-----+-----+-----+-----+
>| S0  |   |   |   | 10 | 0 | 0 | 0 | 0 |
>+-----+-----+-----+-----+-----+-----+-----+
>| S1  | 10 | 10 | 10 | 10 | 0 | 0 | 0 | 0 |
>+-----+-----+-----+-----+-----+-----+-----+
>| S2  | 9  | 7  | 8  | 10 | 2 | 0 | 1 | 3 |
>+-----+-----+-----+-----+-----+-----+-----+
>| S3  | 5  | 3  | 4  | 10 | 4 | 2 | 3 | 9 |
>+-----+-----+-----+-----+-----+-----+-----+

```

`Global_remaining_limit = Kingbird_global_limit - Su(sum of all usages)`
`Region_new_limit = Global_remaining_limit + resource_usage_in_that_region.`





8 Create Network1-2

11 Create VM2(Port2, Network1-2)

