OpenStack in Sina

@程辉

freedomhui@gmail.com

Agenda

- OpenStack Overview
- Architecture Analysis
- Integration Changes
- Sina Contributions


amazon
web services™

AWS模式的巨大成功

- 构建了完整的云计算生态系统
- 通过Web Service(API)管理一切服务*
- 完全面向服务架构SOA(Service-Oriented Architecture)*
- 事实上的IaaS 标准
- 成功的商业模式

*https://plus.google.com/112678702228711889851/posts/eVeouesvaVX
*http://coolshell.cn/articles/5701.html

AWS Overview

Tools to access services

- Libraries & SDK's
- Management Console
- Command Line Interface

Cross Service features

Identity

IAM Add-on

Monitoring

Amazon CloudWatch

Deployment & Automation

CloudFormation

Elastic Beanstalk

Platform building blocks

Messaging

Simple Email Service (SES)

Simple Queue Service (SQS)

Content Distribution

Amazon CloudFront

Parallel Processing

Amazon Elastic MapReduce

Infrastructure building blocks

Compute

Elastic Compute Cloud (EC2)

Storage

Elastic Block Storage (EBS)

Simple Storage Service (S3)

Networking

Elastic Load Balancer

Route 53

Virtual Private Cloud (VPC)

Database

Relational Database Service

ElastiCache

SimpleDB

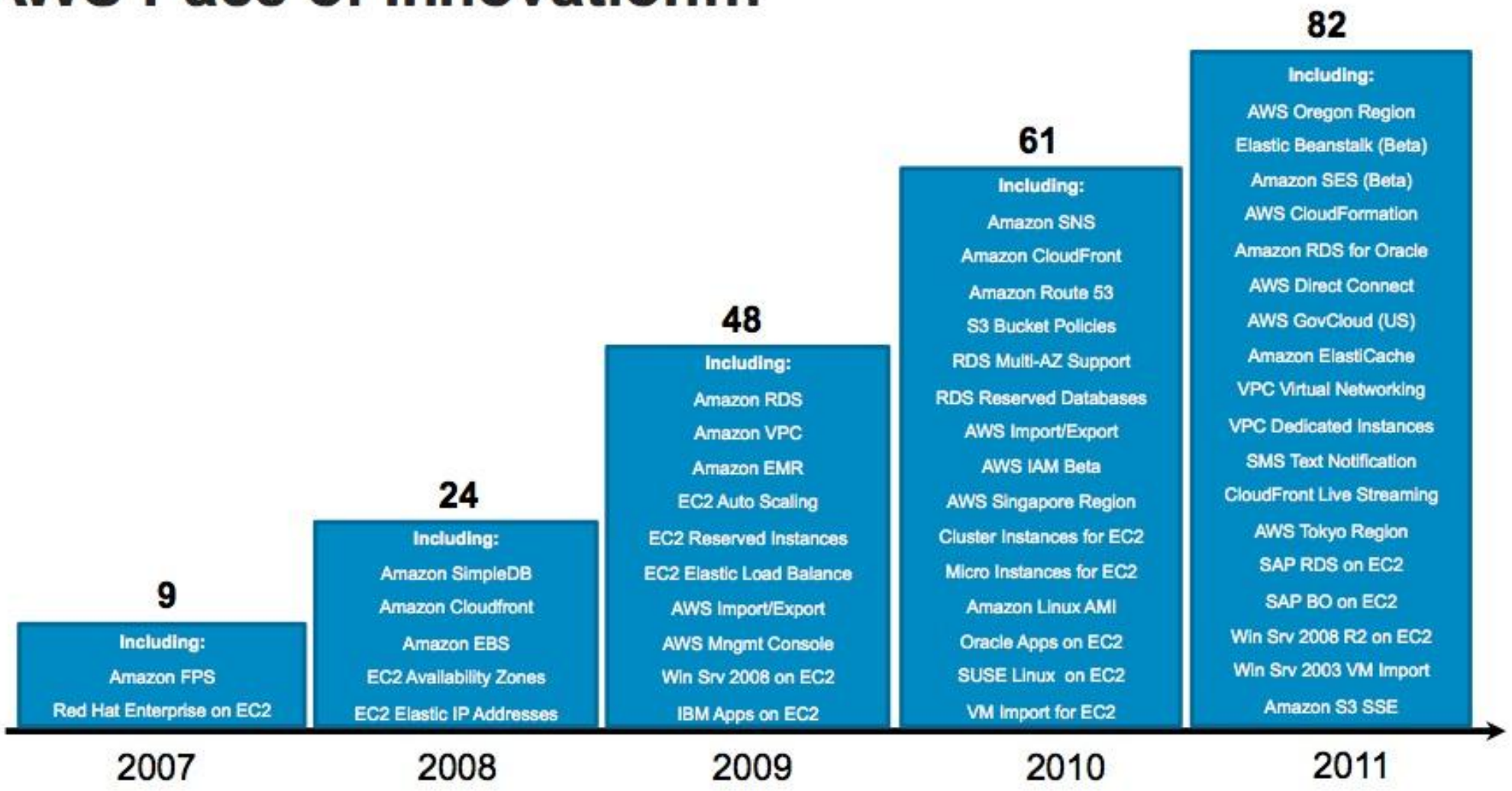
Region

Availability Zone

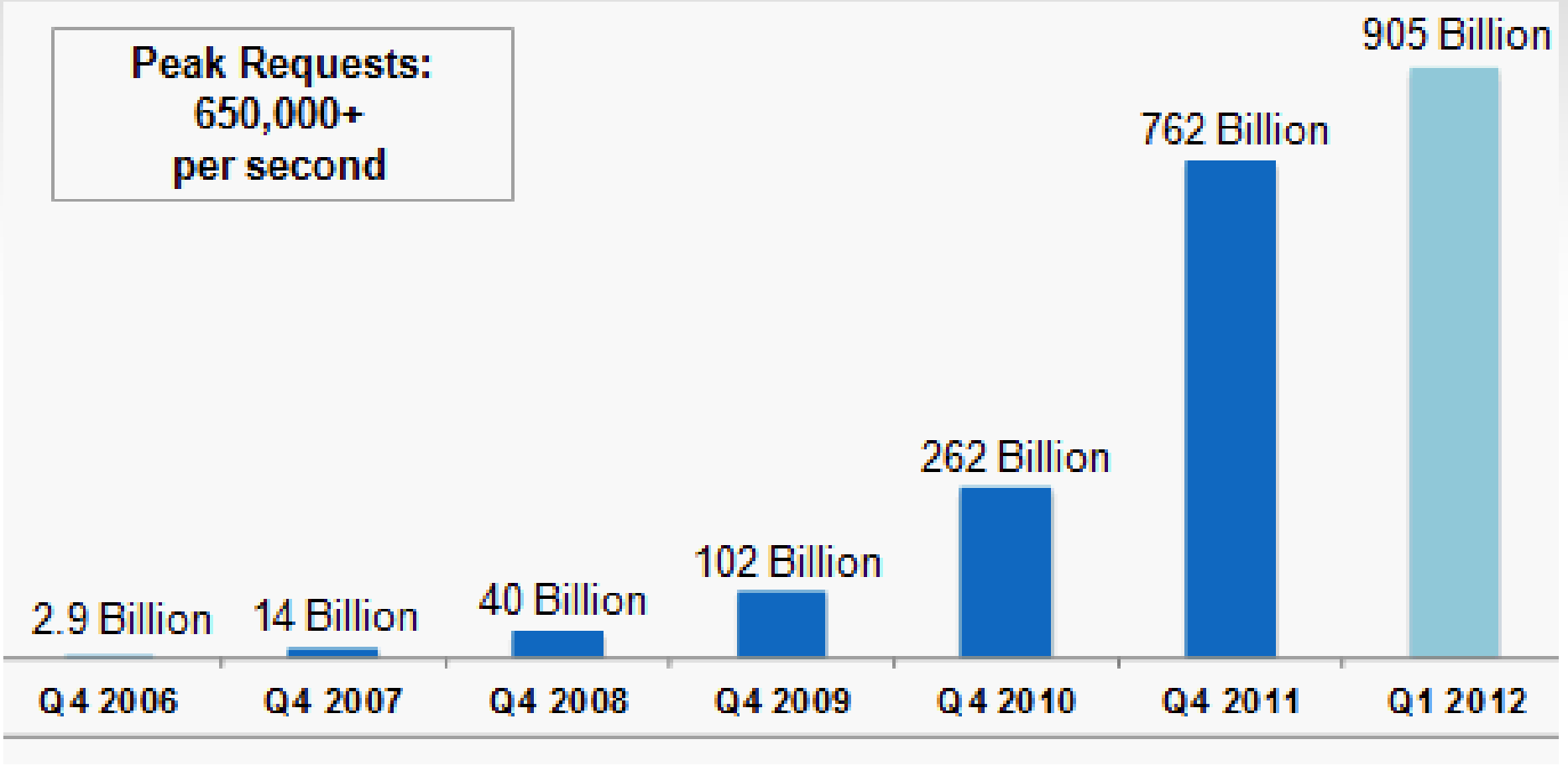
Edge Location

AWS Global Infrastructure

AWS Pace of Innovation…



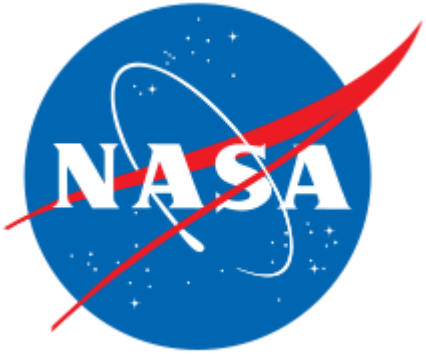
More Detail: http://aws.amazon.com/products/


Peak Requests:
650,000+
per second
2.9 Billion
14 Billion
40 Billion
102 Billion
262 Billion
762 Billion
905 Billion
Q4 2006
Q4 2007
Q4 2008
Q4 2009
Q4 2010
Q4 2011
Q1 2012

OpenStack横空出世

- 目标：AWS开源实现
- Rackspace & NASA联合成立

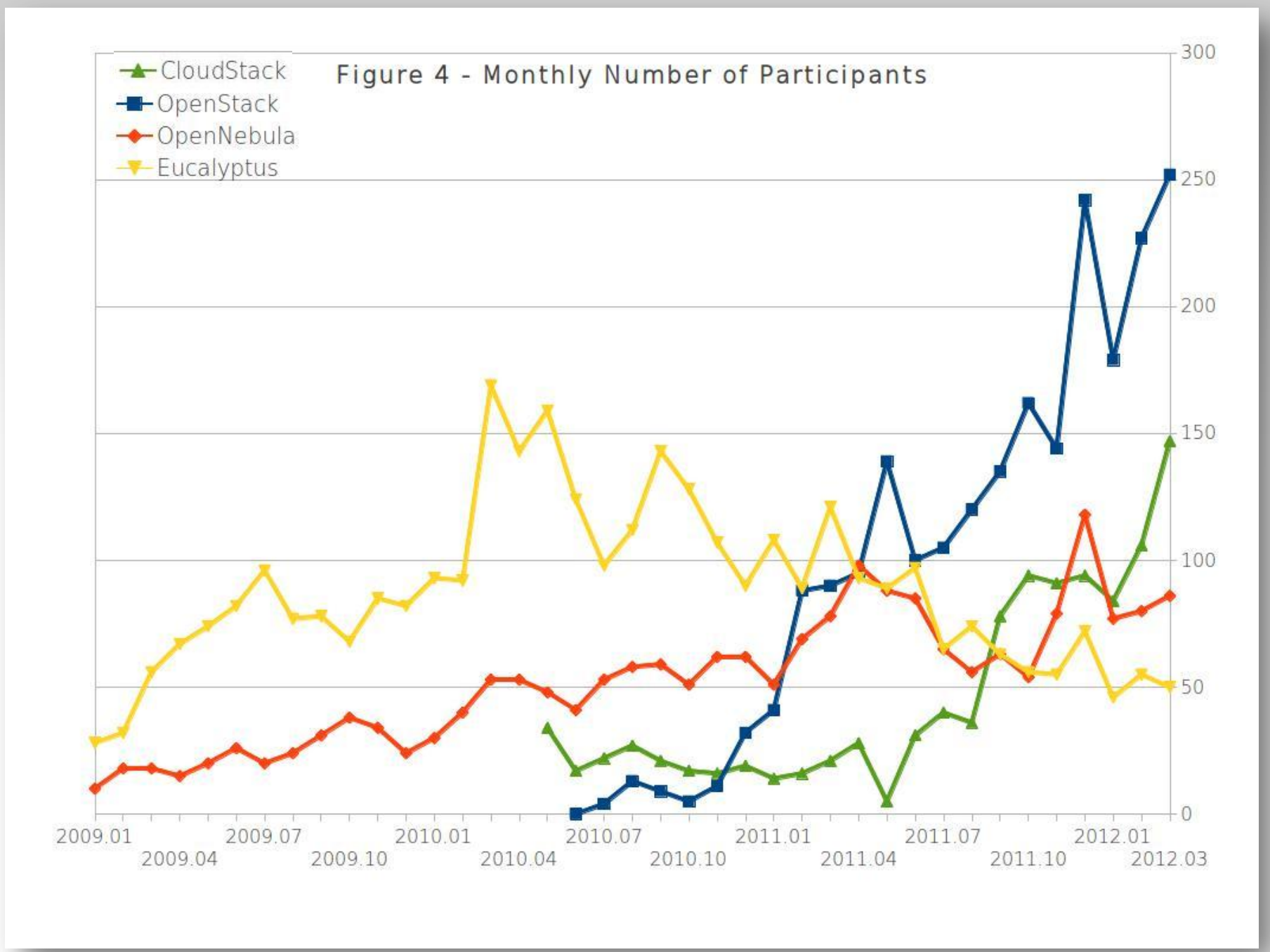
Swift



Nova

OpenStack Companies

More detail: http://www.openstack.org/community/companies/



来源：《OpenStack, OpenNebula，Eucalyptus，CloudStack社区活跃度比较》http://www.qyjohn.net/?p=1856

Open Source

Apache 2.0 license, NO ‘enterprise’ version

Open Design

Open Design Summit

Open Development

Anyone can involve development process Open development management via Launchpad & Github

Open Community

OpenStack Foundation in 2012

OpenStack Mission

"To produce the ubiquitous **Open Source** cloud computing platform that will meet the needs of **public and private** cloud providers regardless of size, by being **simple** to implement and **massively scalable**."

OpenStack Projects

Core Projects

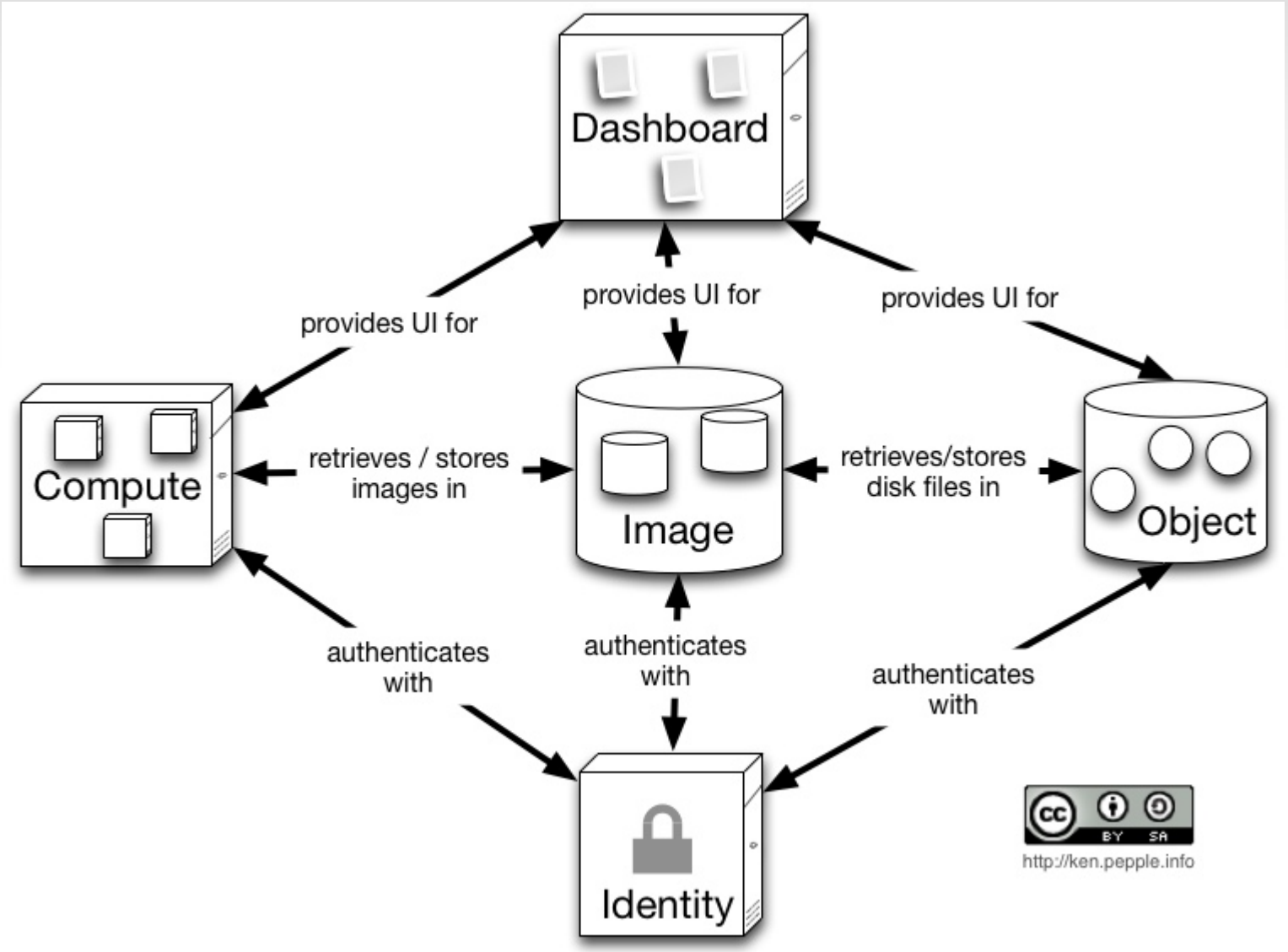
- OpenStack Compute(**Nova**)
- OpenStack Object Storage(**Swift**)
- Image Service (Glance)
- Identity (**Keystone**)
- Dashboard (Horizon)
- Network Connectivity (**Quantum**)

Community Projects

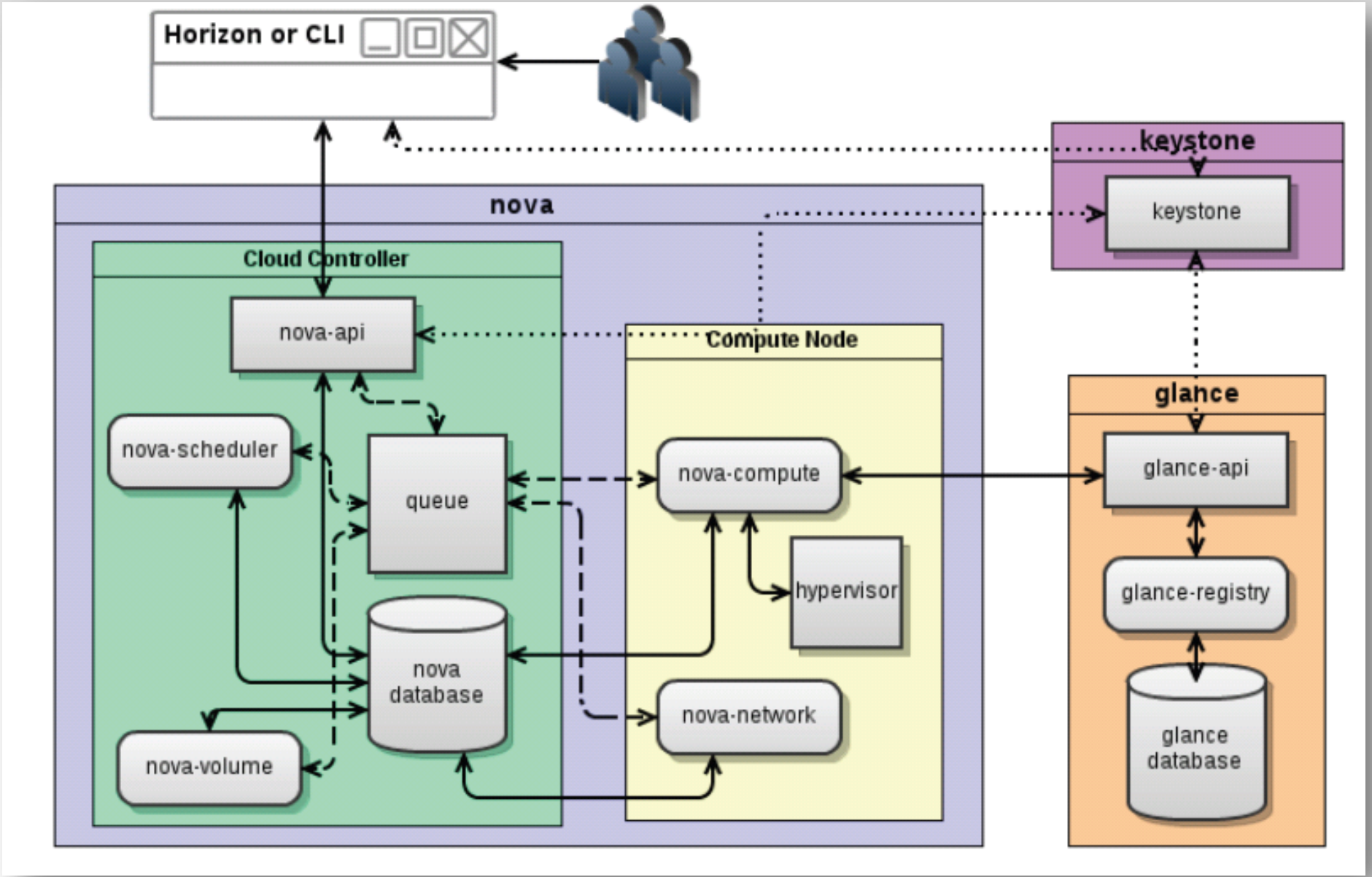
- Melange
- Altas-LB
- Crowbar
- Juju
- RedDwarf
- Burrow

AWS	OpenStack
EC2	nova
S3	swift
EBS	nova-volume
ELB	Atlas-LB
SQS	Burrow
Console	Dashboard
IAM	Keystone
VPC	Quantum
RDS	RedDwarf

Architecture Overview



Detail Overview



Where to Get Started?

Ubuntu 12.04 server 集成OpenStack

Trystack.org申请测试账号

devstack.sh一键安装

OpenStack Development

Authorization
(group membership)

Bug tracking

Hosting code &
formal docs

github

Feature planning
(Blueprints)

Mailing lists

User support
(Answers)

Wiki

Informal docs

gerrit
Gerrit Code Review

Jenkins

Continuous
integration

Nova Key Features

- ReST-base API
- Asynchronous communication
- Horizontally scalable
- Shared nothing architecture*
- Distribute everything
- Test everything
- 100% Python Based

* http://en.wikipedia.org/wiki/Shared_nothing_architecture
* http://wiki.openstack.org/BasicDesignTenets

OpenStack Compute: Nova

nova-api

Compute API Server

OpenStack API, EC2 compatibility API

nova-compute

Compute worker

Manage compute host and VMs

Libvirt(QEMU,KVM,LXR), XenServer and XCP, ESX(i)*

nova-network

Network controller

Manage network resources: IPAM, VLAN, NAT

*http://wiki.openstack.org/HypervisorSupportMatrix

OpenStack Compute: Nova(cont.)

nova-scheduler

Determines the placement of new resources

nova-volume

Block storage, remote attach a LVM volume using iSCISI protocol

Like Amazon EBS, but far way from mature

RabbitMQ

Message Queue

Cast and RPC Call for services

Keystone: Concept

User/Tenant

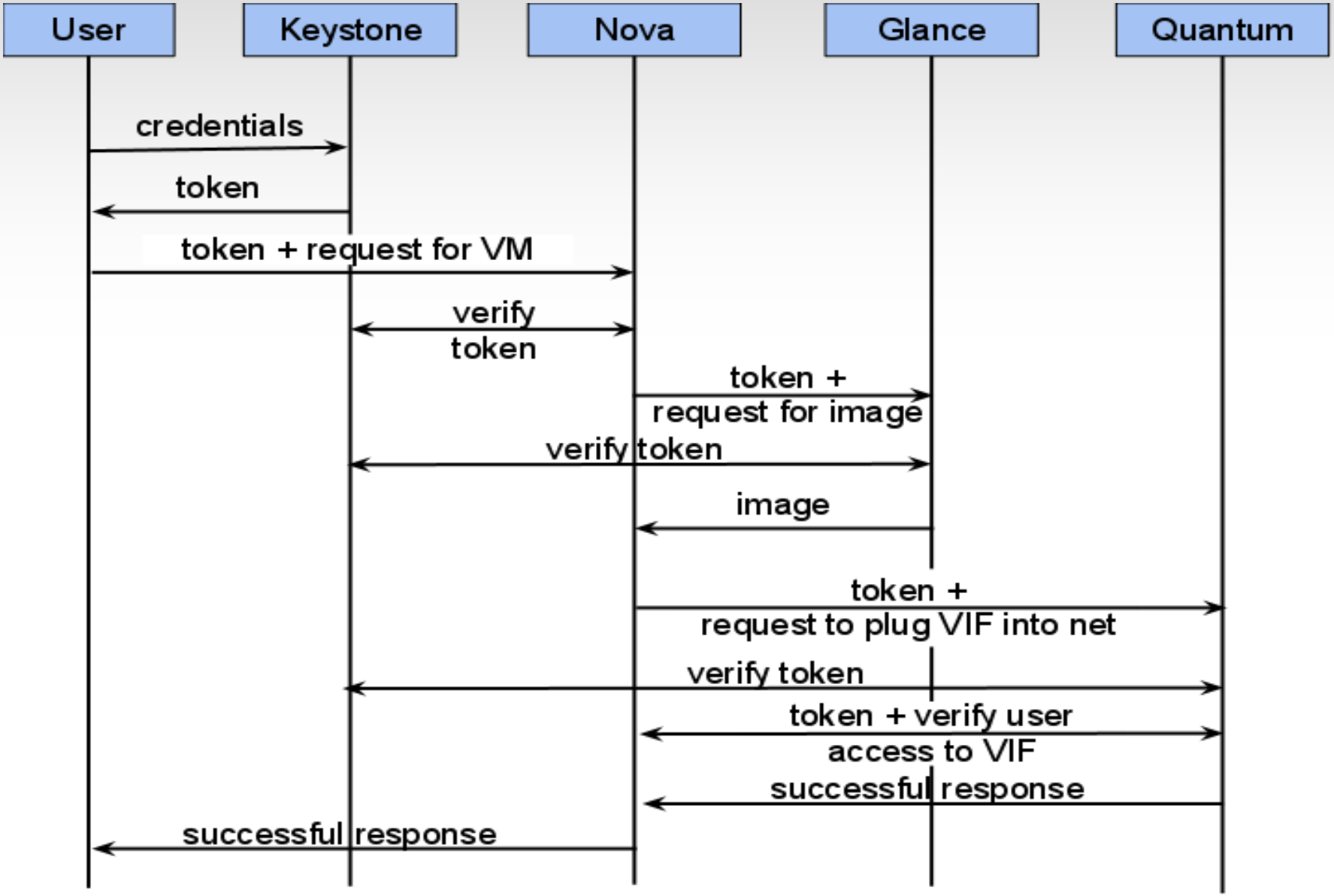
Authentication/Authorization

Token

Service/Endpoint

Role

Keystone: User Case



Nova Network

L2

FLAT, FLATDHCP, VLAN

L3

IPAM(IP Address Management)

Fixed IP, Floating IP

Gateway, NAT, VPN

Quantum

Quantum Basics

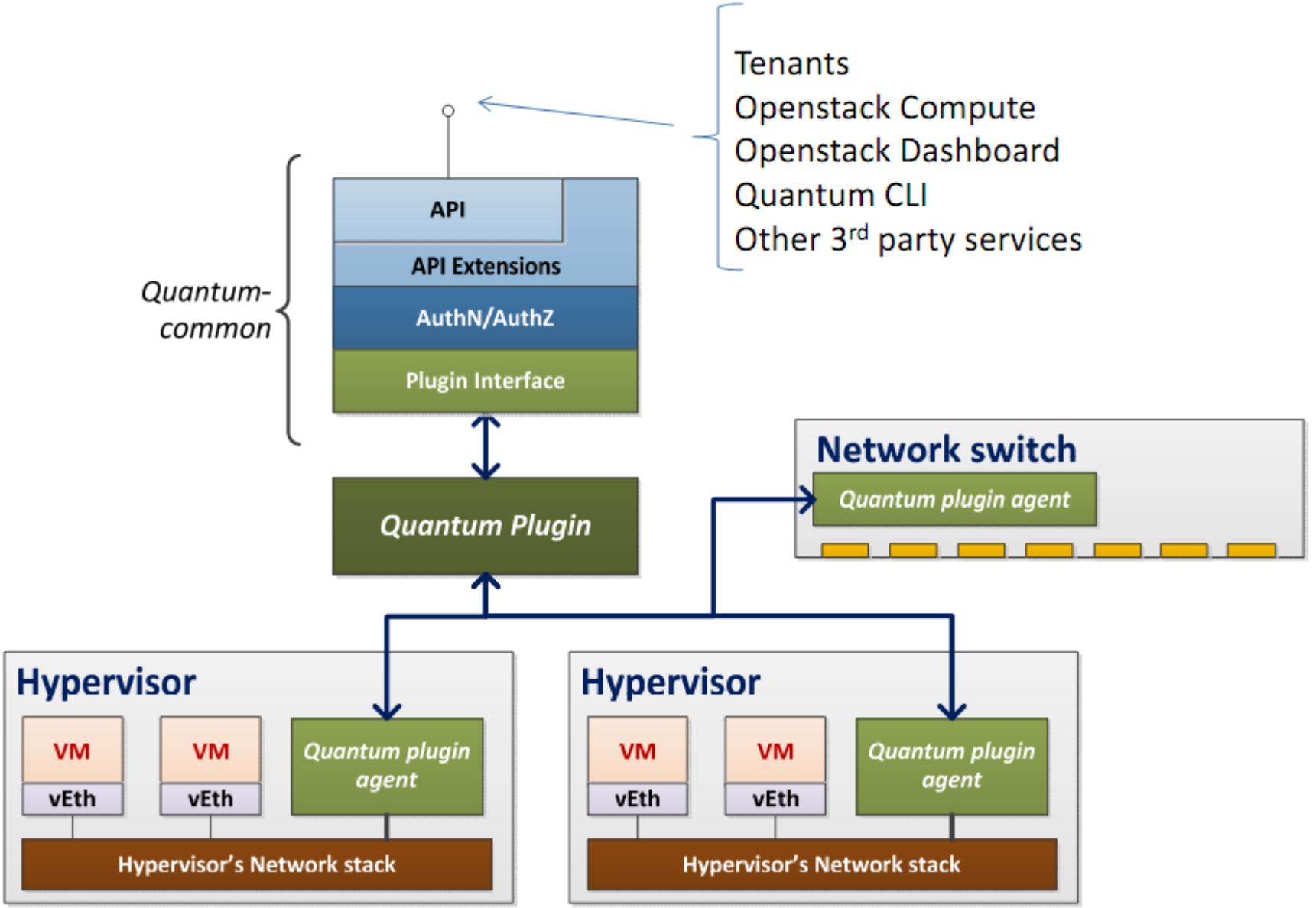
Nova: virtual server

Quantum: virtual network

Quantum :

- Expose a API for creating virtual networks and attaching instances(e.g.,novaservers) to those networks
- Manage switches(virtual or physical) in the data center to implement connectivity described via API
- Provide a “plugin” architecture to leverage support using different back-end technologies

The Quantum Service



Quantum: available plugins

Open vSwitch

- Builds isolated networks with OVS and L2-in-L3 tunnel

Cisco UCS

- Isolation based on VLAN and net-profiles applied to Cisco UCS converged network adapters

Linux Bridge

- Build isolated networks with VLAN interfaces and linux bridges

- Works with every Linux Distro

NTT-Data Ryu

- Acts as a proxy for the NTT Ryu platform

Nicira NVP

- Acts as a proxy for the Nicira NVP platform

Swift: Storage Types

Types	Protocol	Application
Block Storage	SATA, SCISI, iSCISI	SAN, NAS, EBS
File Storage	Ext3/4, XFS, NTFS	PC, Servers, NFS
Object Storage	HTTP, REST	Amazon S3, Google Cloud Storage, Rackspace Cloud Files
Specific Storage	Specific protocol based on tcp	MySQL, MongoDB, HDFS

We want a Object Storage like Amazon S3.

Swift vs Amazon S3

Features	Swift	Amazon S3
object/bucket CRUD	√	√
account/bucket/object ACL	√	√
object metadate	√	√
large object	√	√
rate limit	√	√
expiring object	√	√
static web	√	√
REST API	√	√
Account support	√	X
Account metadata	√	X
Bucket metadata	√	X
Bucket sync across cluster	√	X
Object versioning	X	√
Log to bucket	X	√
Notification	X	√
Reduced Redundancy Storage	X	√
SOAP API	X	√
Server Side Encryption	X	√
BitTorrent protocol	X	√

Swift Evaluation

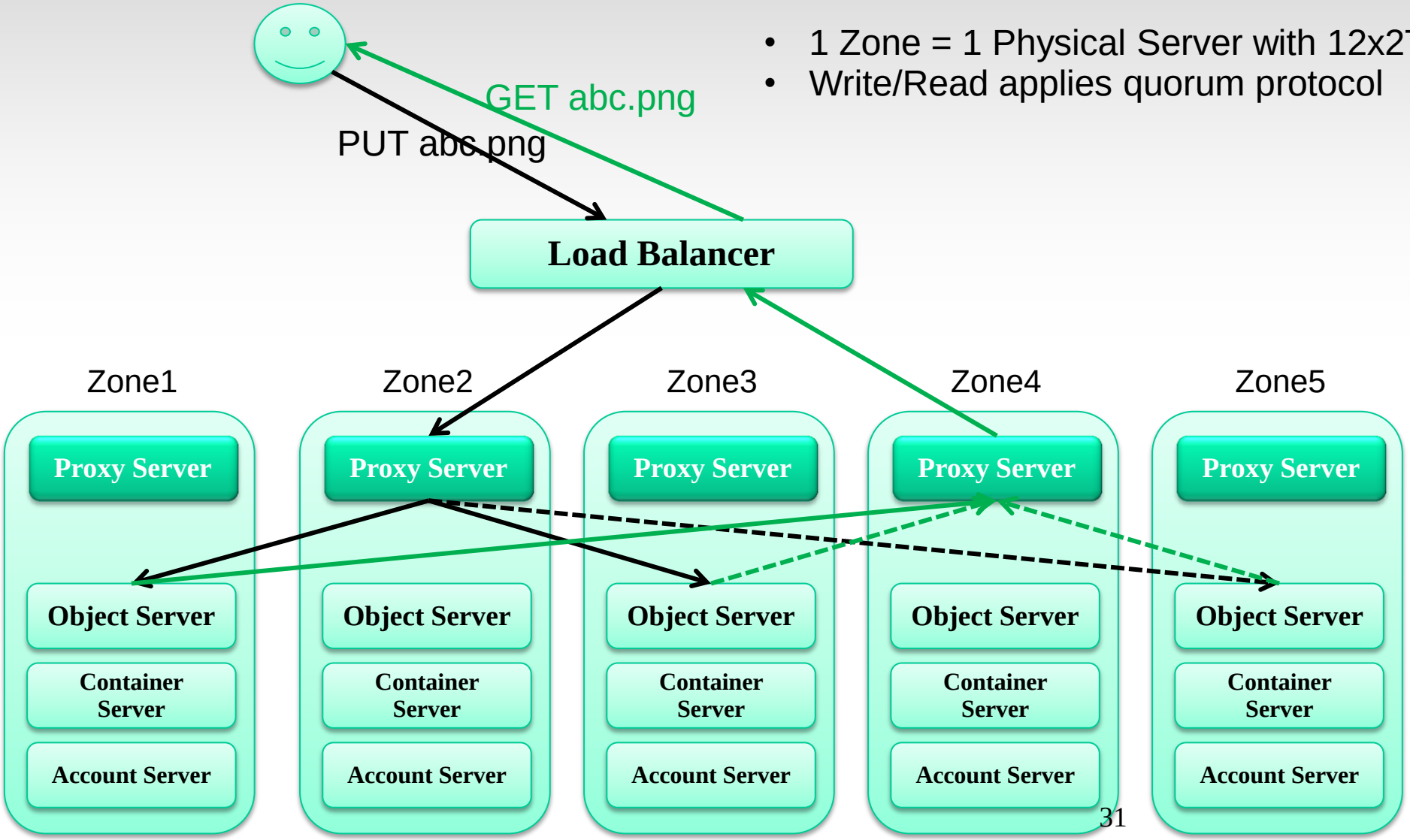
- Extremely Durable and Highly Available
- Superior Scalability
- Linear Growth of Performance
- Symmetric Architecture
- No Single-failure
- Simple & Reliable

Swift Components

- **The Ring**: Mapping of names to entities (accounts,containers, objects) on disk.
  - Stores data based on zones, devices, partitions, and replicas
  - Weights can be used to balance the distribution of partitions
  - Used by the Proxy Server for many background processes
- **Proxy Server**: Request routing, exposes the public API
- **Replication**: Keep the system consistent, handle failures
- **Updaters**: Process failed or queued updates
- **Auditors**: Verify integrity of objects, containers, and account

Swift Architecture

- 1 Zone = 1 Physical Server with 12x2T disk
- Write/Read applies quorum protocol



Swift Installation

Swift packages
Proxy Server
Account Server
Container Server
Object Server

```
root@sws-swift-2:~# ps aux | grep swif[t] | awk '{print $12" "$13}' | uniq -u | sort
/usr/local/bin/swift-account-auditor /etc/swift/account-server.conf
/usr/local/bin/swift-account-reaper /etc/swift/account-server.conf
/usr/local/bin/swift-account-replicator /etc/swift/account-server.conf
/usr/local/bin/swift-account-server /etc/swift/account-server.conf
/usr/local/bin/swift-container-auditor /etc/swift/container-server.conf
/usr/local/bin/swift-container-replicator /etc/swift/container-server.conf
/usr/local/bin/swift-container-server /etc/swift/container-server.conf
/usr/local/bin/swift-container-updater /etc/swift/container-server.conf
/usr/local/bin/swift-object-auditor /etc/swift/object-server.conf
/usr/local/bin/swift-object-auditor /etc/swift/object-server.conf
/usr/local/bin/swift-object-replicator /etc/swift/object-server.conf
/usr/local/bin/swift-object-server /etc/swift/object-server.conf
/usr/local/bin/swift-object-updater /etc/swift/object-server.conf
root@sws-swift-2:~#
0 bash  1 bash  2 bash  3 bash  4 bash
```

OS installation

sda

raid 1

disk1 **disk2**

Storage Nodes

sdb **sdc** **sdd** ..... **sdk**

disk3 **disk4** **disk5** **disk12**

Conclusion

- 核心功能基本可用，但稳定性需要加强
- 云服务(web service)比较丰富
- 起步虽晚，但发展飞快，OpenStack生态系统正在形成
- 逻辑结构清晰、文档丰富、源码规范易懂，便于二次开发
- **Open** [Source | Desgin | Development | Community]

Integration Challenges

- Best Network Topology
- Security Enhancement
- Load Balancer
- CDN Services
- Metering & Billing

Infrastructure & Platform

Physical Servers

Traditional Operation

Virtualization Platform(IaaS)

- VM Management System(VMMS) → Sina Web Service(SWS)
- VMMS is private solution developed in-house
- SWS is based on OpenStack

新浪云计算

Application Platform(PaaS)

- Virtual Host → Sina App Engine(SAE)
- SAE provides both Public and Private Service.
- Proved to be Efficient and Robust

Nova Network

Networking is the biggest challenges for IaaS

Network Topology:

- VLAN
- FlatDHCP
- FlatDHCP & Multihost

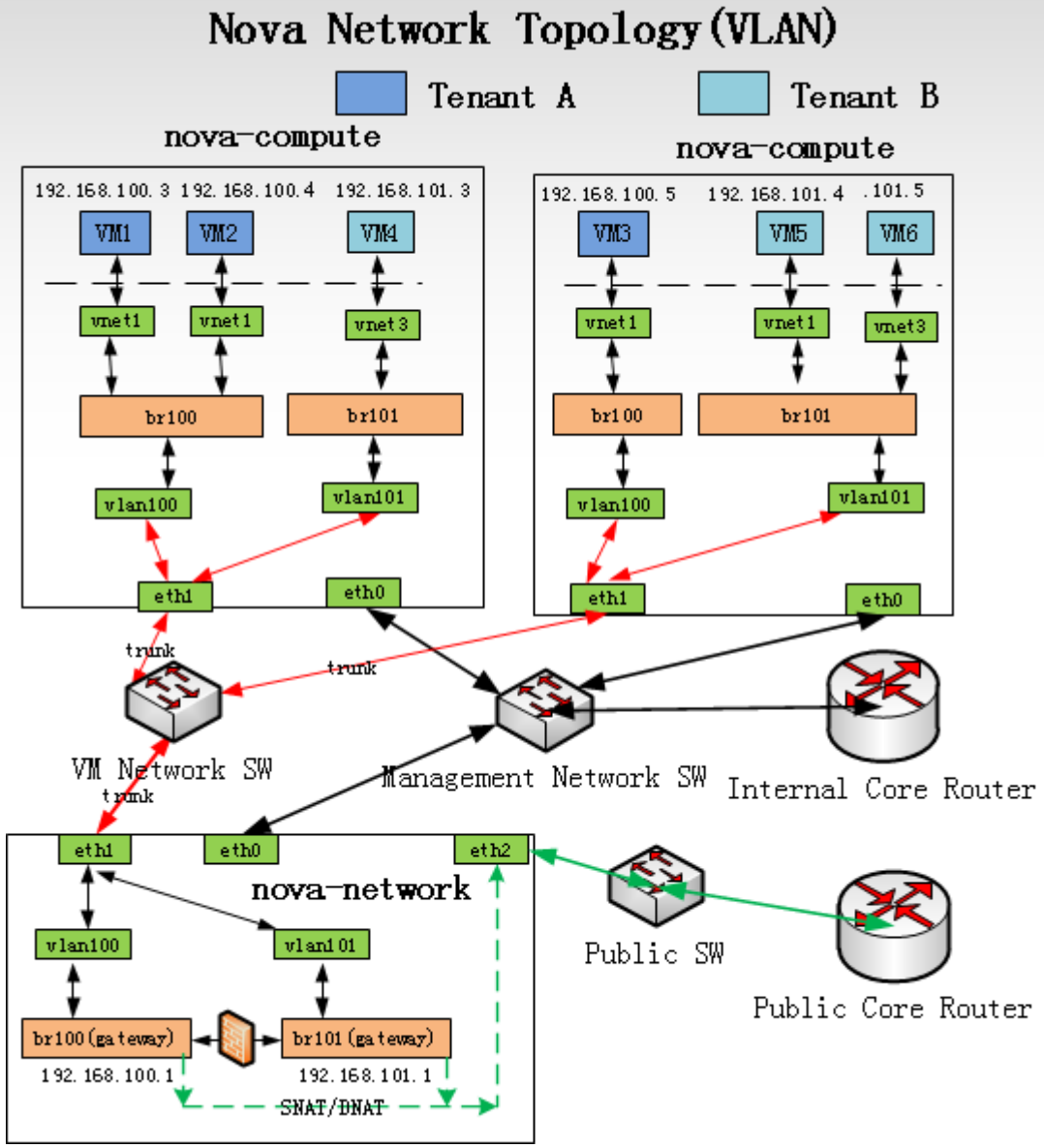
Network Topology (VLAN)

Capability:

- Accessibility of VMs within one tenant
- Isolation of VMs from different tenants
- VM is able to access public network
- VM can be accessible from public network
- Isolation between virtual network and internal network

Drawback:

- Pre-allocate network for future projects
- Hard-limit of vlan 4096
- Traffic bottleneck in the gateway/NAT



Network Topology(Flat)

Capability:

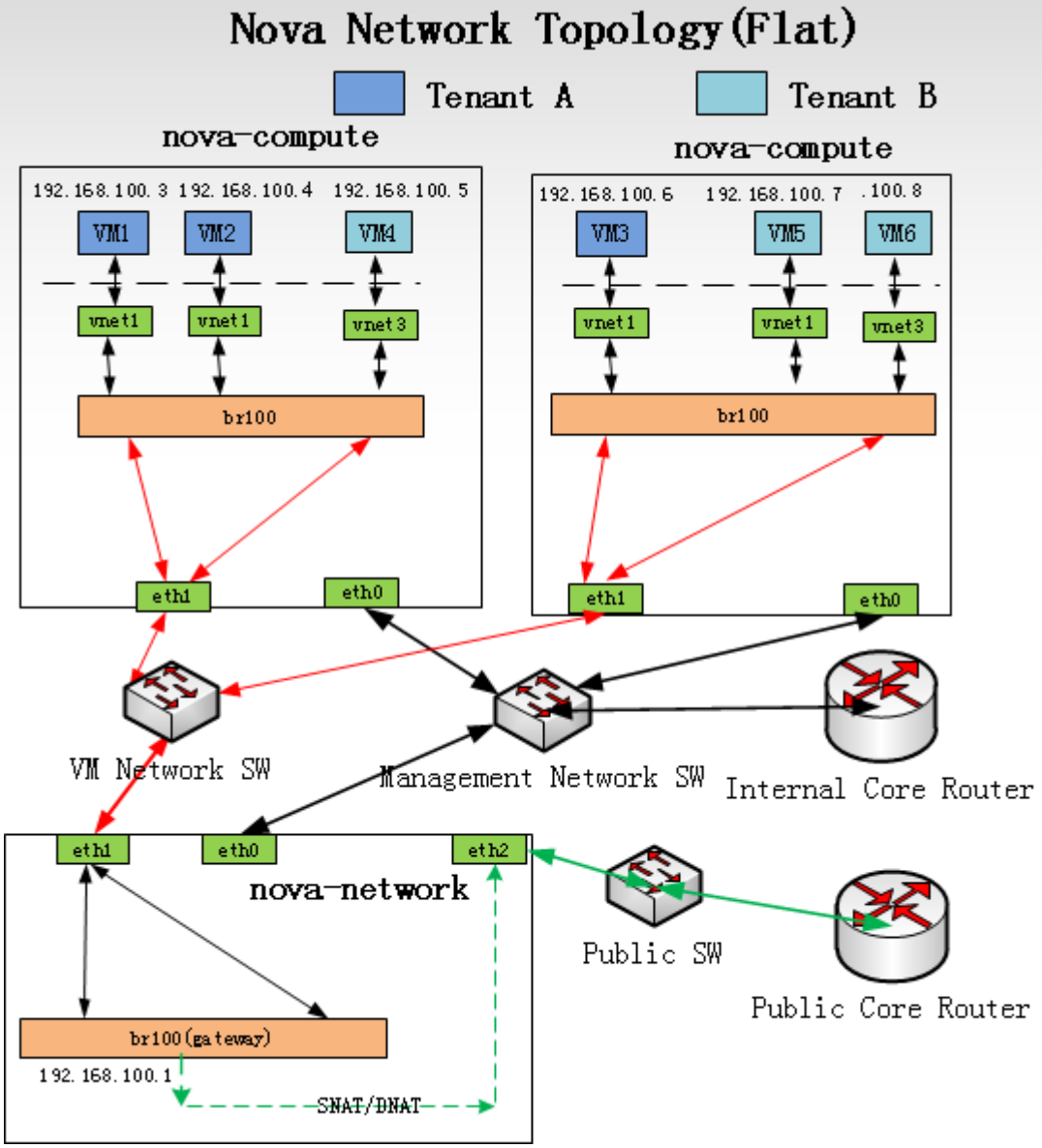
- Accessibility of all VMs in the fixed IP range
- VM is able to access public network
- VM can be accessible from public network
- Full isolation between virtual network and internal network

Bonus:

- Do not need pre-allocate for new projects
- Eliminating bottleneck between tenants

Drawback:

- Tenant isolation has gone
- Traffic bottleneck still exists in NAT



Network Topology(Flat & Multihost)

Capability:

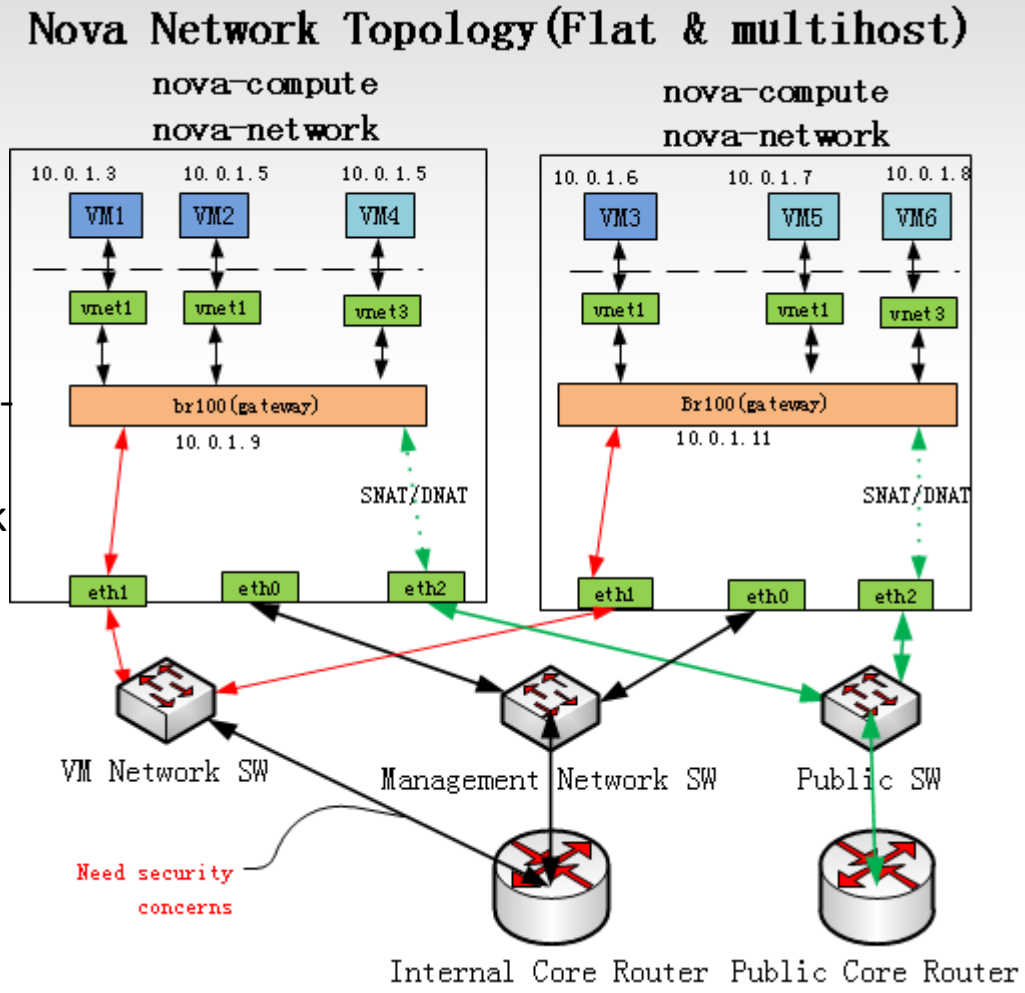
- Accessibility of all VMs in the fixed IP range
- VM is able to access public network
- VM can be accessible from public network

Bonus:

- Totally distributed architecture avoid single-point failure.
- Multiple gateway eliminates NAT bottleneck
- **High speed between OS regions**

Drawback:

- Tenant isolation lessens
- Need security facility(*SWS-filter*) to protect intranet



If security problems were solved, this would be our **best choice!**

Security in OpenStack

Security Group --- L3 Filter

Role-based firewall
- One security group is a Role

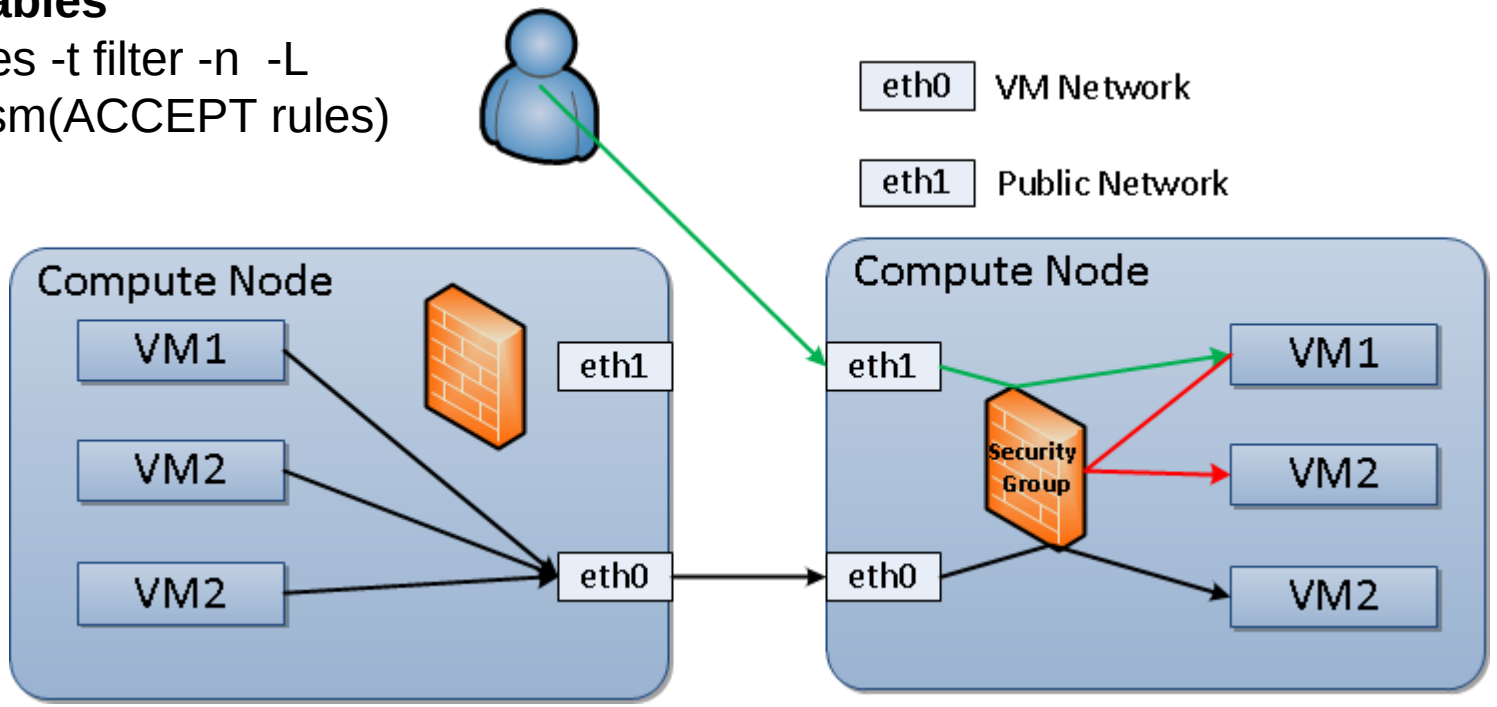
Ingress filtering
- Target is the instance
- Source can be CIDR or another group

Implemented by iptables
- See details: iptables -t filter -n -L
- Whitelist mechanism(ACCEPT rules)

Static filters --- L2 Filter

MAC, IP, and ARP spoofing protection
- Not configurable
- Defined in /etc/libvirt/nwfilter/*.xml

Implemented by ebtables
- ebtables -t nat --list



Security Enhancement

SWS Filter

Prevent Intranet Penetration

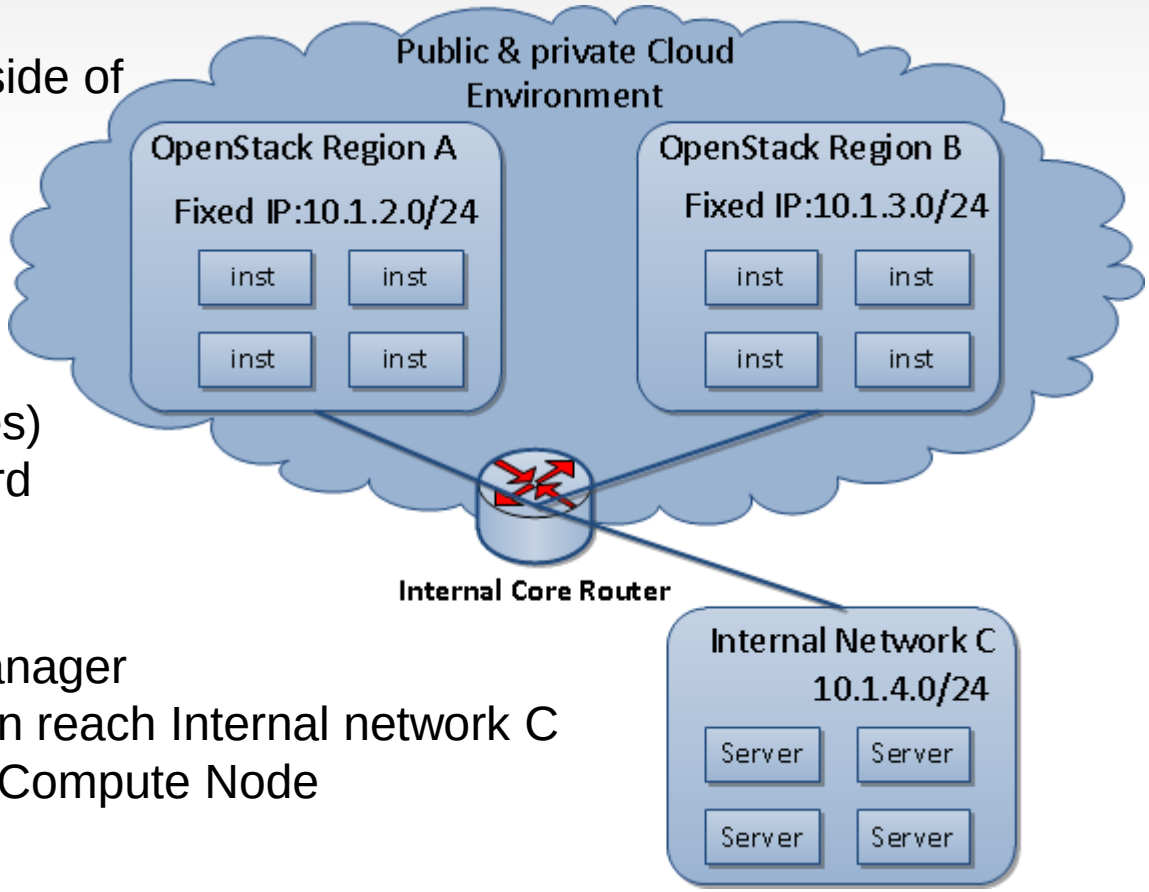
- Intranet is the internal network outside of OpenStack

Egress filtering

- Target is internal network
- Source is instances in OpenStack

Implementation

- Whitelist mechanism(ACCEPT rules)
- On the top of nova-filter-top Forward Chain

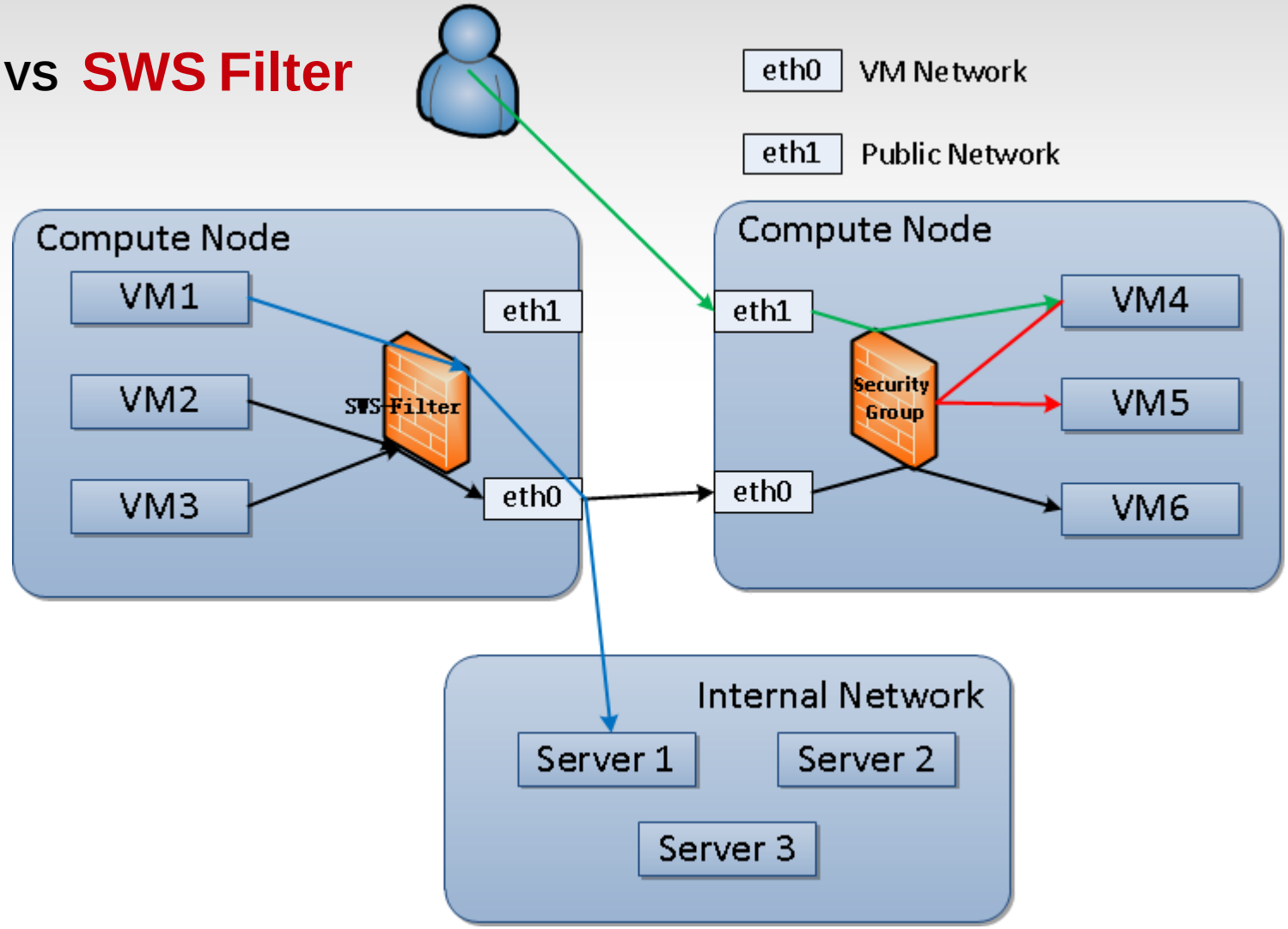


Rational

- SWS filter is managed by cloud manager
- Only explicit authorized packets can reach Internal network C
- Packet should be controlled within Compute Node

Security Enhancement

Security Group vs **SWS Filter**



Load Balancer

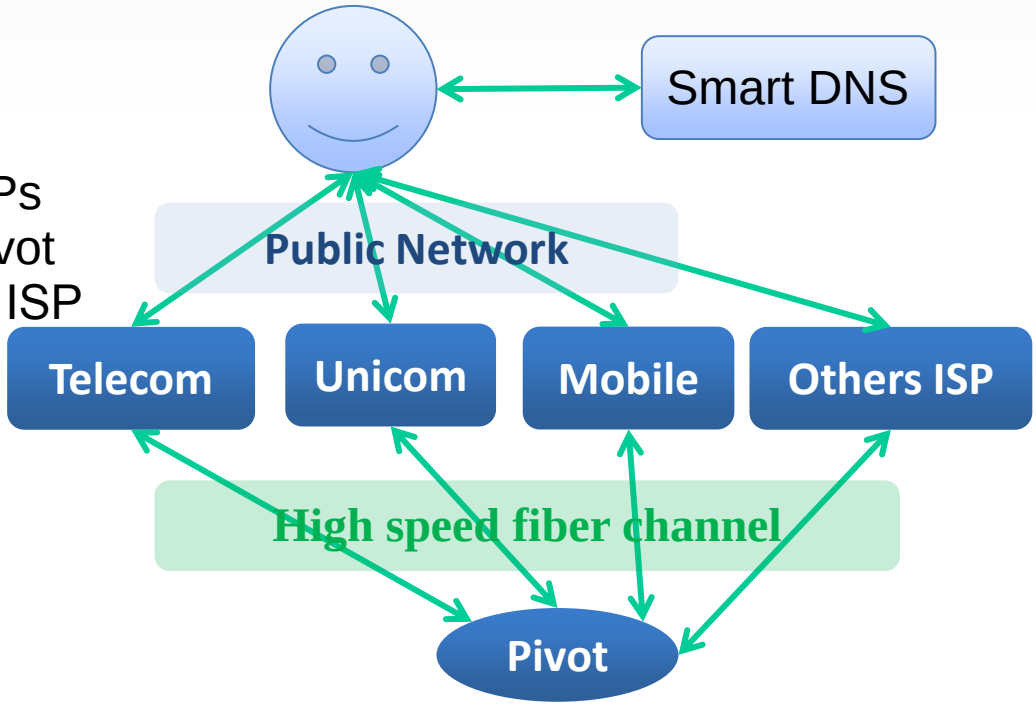
Goals

Load Balance
- Dispatch request
- Support multiple routing algorithm
- Health check

Acceleration
- Reality: narrow bandwidth between ISPs
- Building fiber channels from ISPs to pivot
- Given the same endpoint within user's ISP

IPv4 Shortage
- Reality: dozens of public IPs support hundreds of VMs
- IPv4 has been exhausted
- IPv6 is not realistic yet in China

DNS Acceleration Design

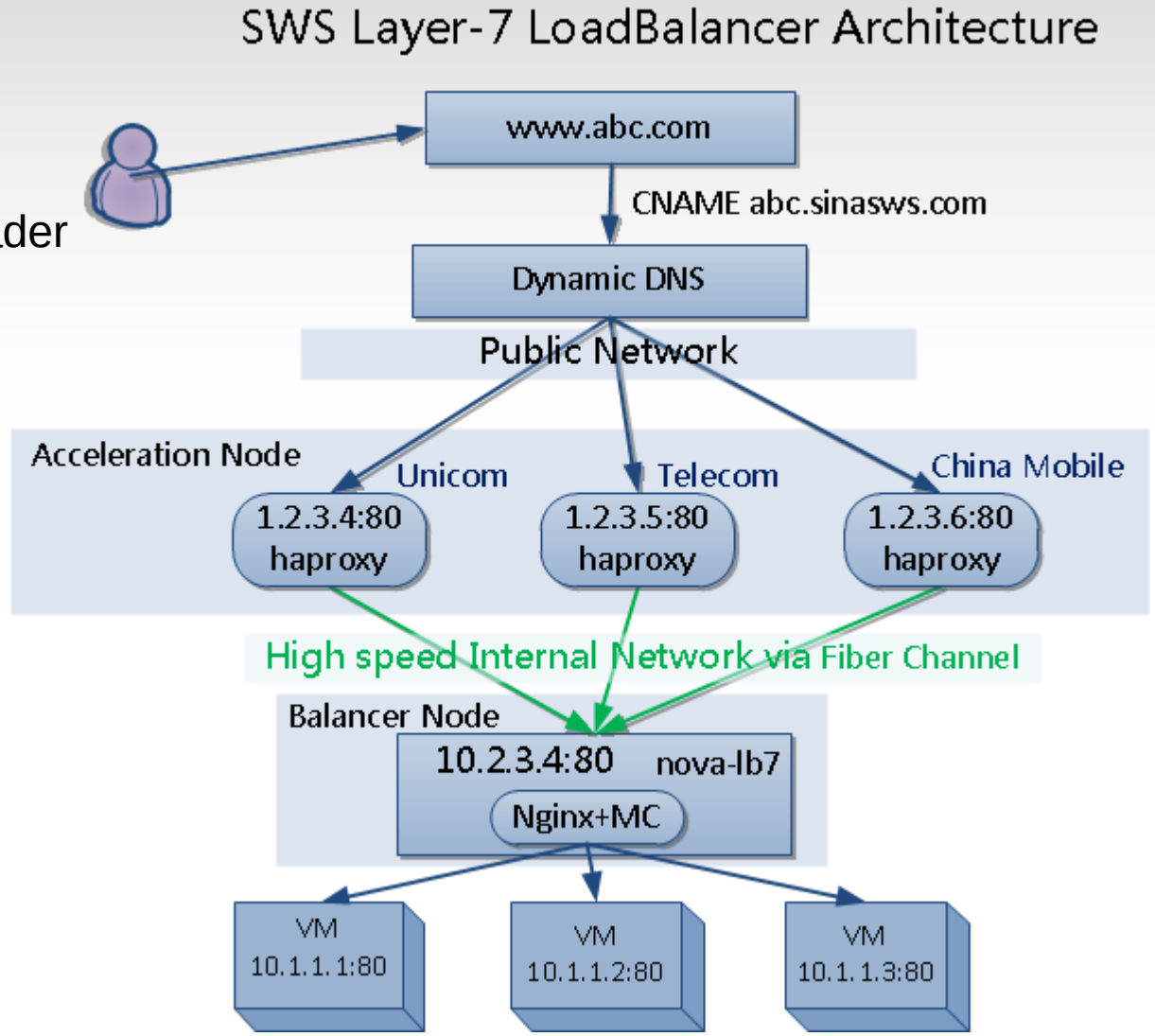


L7 Load Balancer

Layer 7 Load Balancer

Consideration:

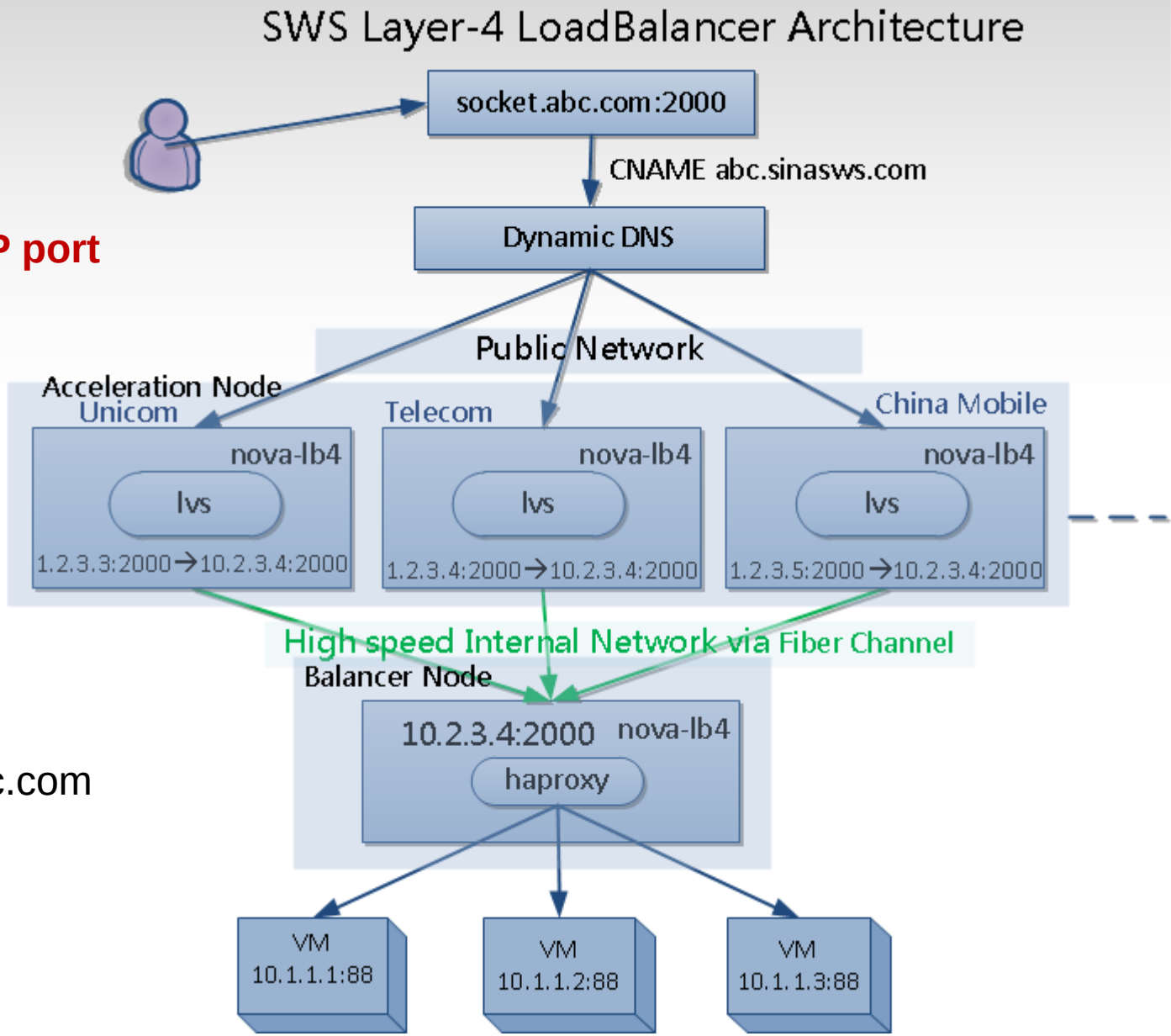
1. dispatch request by ***Host*** header
2. nginx module



L4 Load Balancer

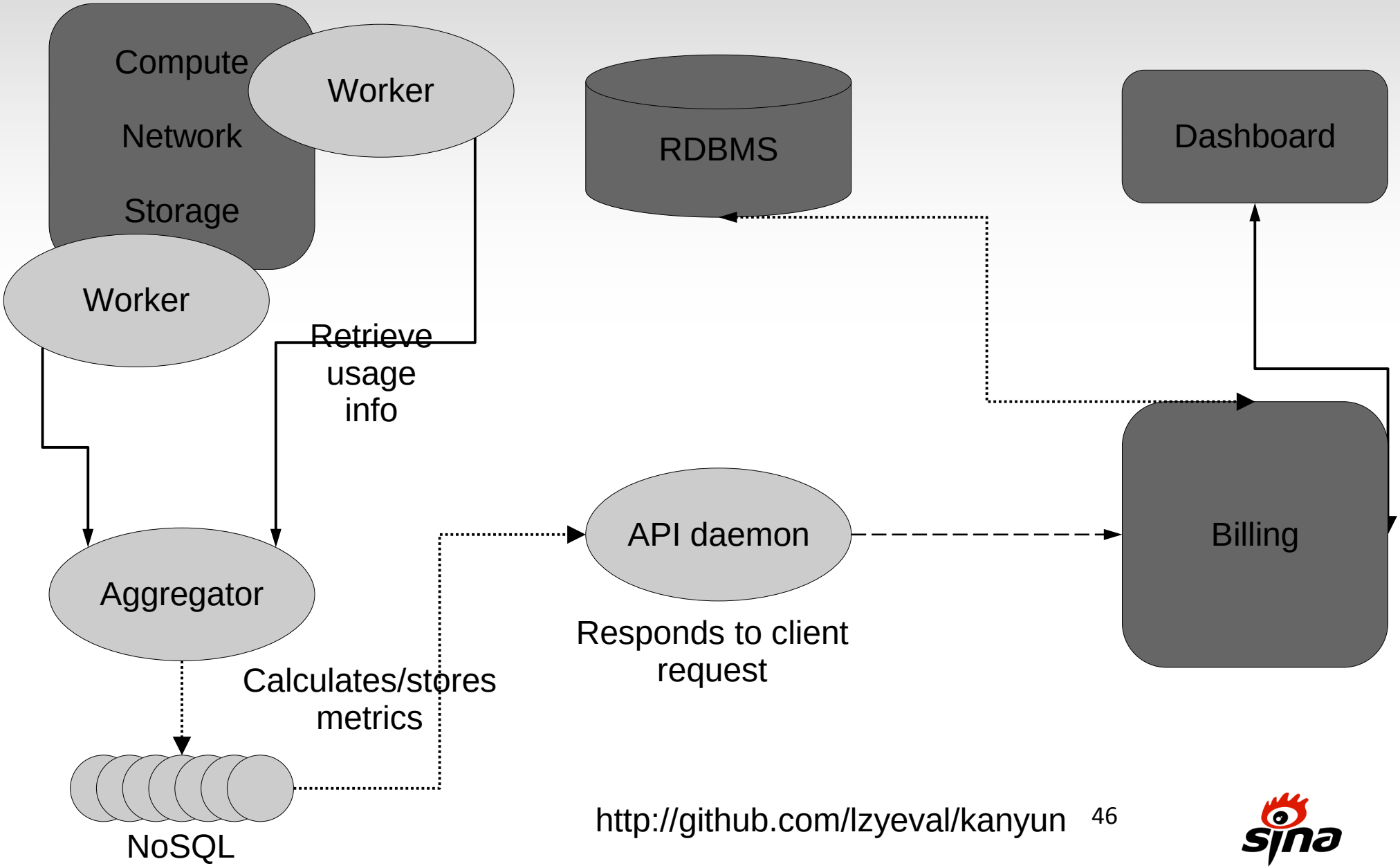
Layer 4 Load Balancer

Consideration:
1. dispatch request by **TCP port**
2. lvs + haproxy



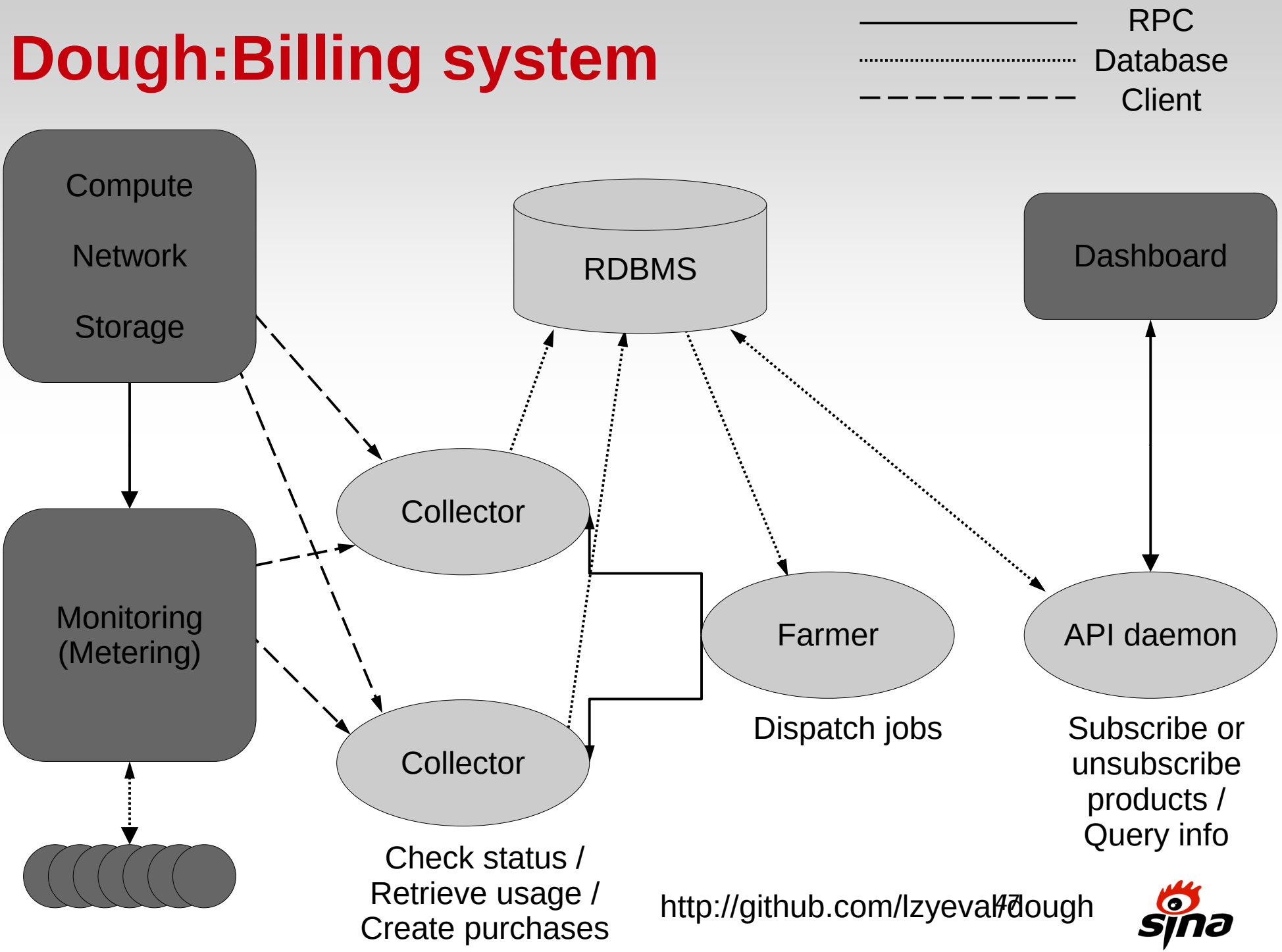
ssh –p 2000 root@socket.abc.com

Kanyun: Monitoring system



http://github.com/lzyeval/kanyun

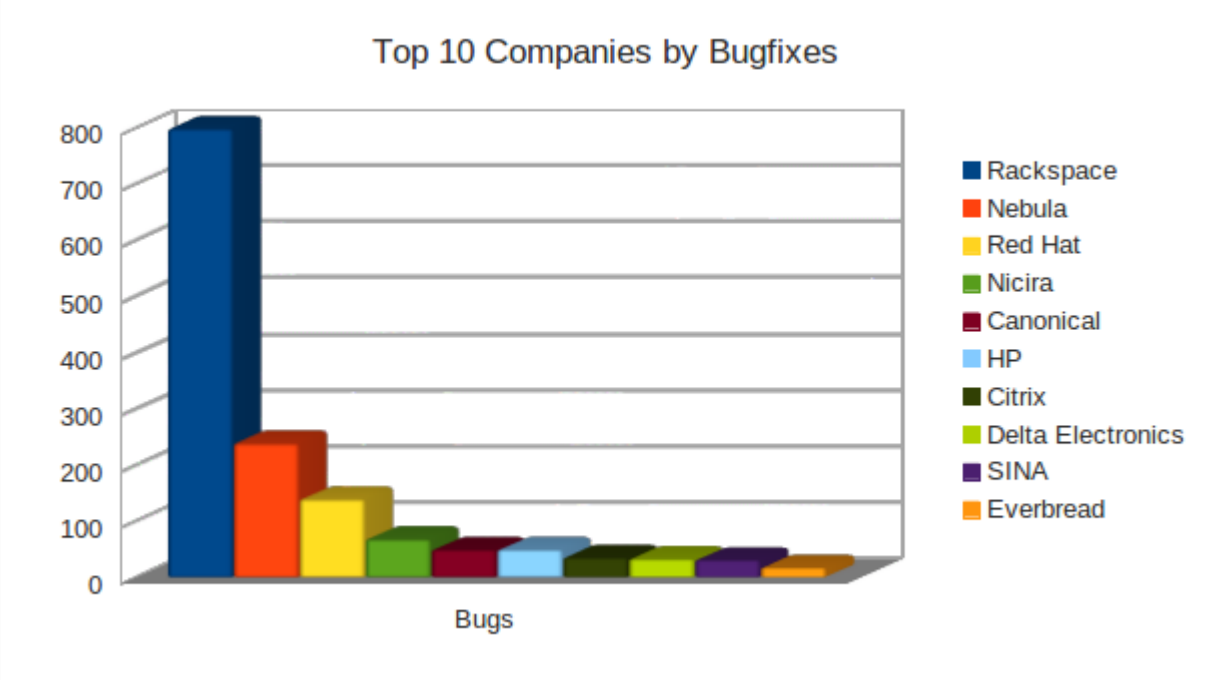
Dough:Billing system



http://github.com/lzyeval/dough

Sina Contributions

Top 10 Bugfix Company



Sina Contributions

- Sina creating open source project “**Dough**” to contribute metering & billing capability
- Present in OpenStack Design Summit & Conference
- OpenStack APEC Conference Promoter

Q & A

@程辉
Technical Manager @ Sina

手机：***
邮箱：chenghui@staff.sina.com.cn
Gtalk: freedomhui@gmail.com