どう使う？ データセンターネットワーキング最前線

LINE 実用例

Verda Network Development Team, LINE Corporation

Hiroki Shirokura
Internet Week 2021

I’m Hiroki Shirokura from LINE

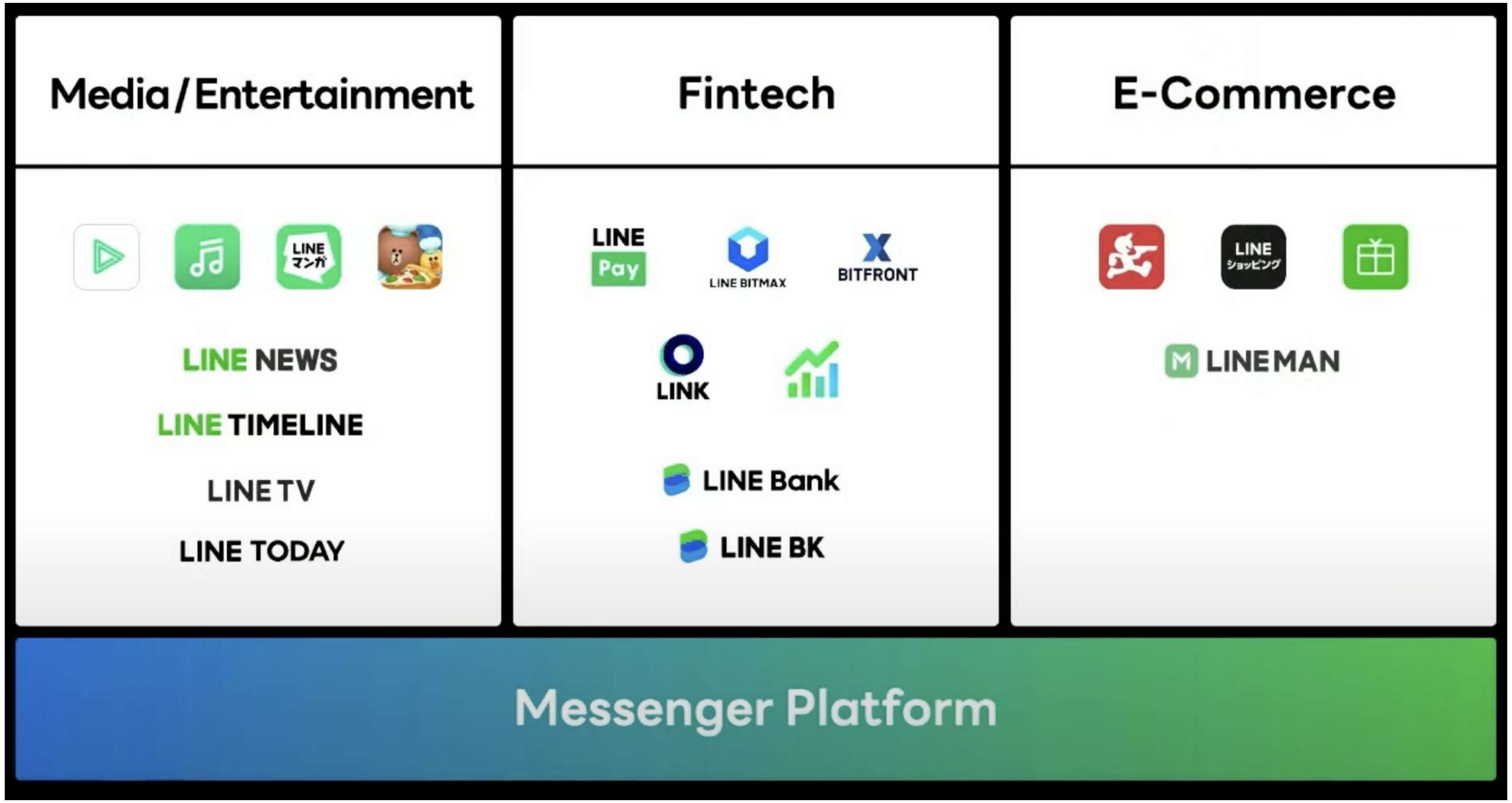
- Senior Software Engineer @ Private Cloud
    - Responsibility: SDN, Cloud Networking
        - Design / Implementation / Reliability
        - SRv6, BGP OSS Upstream Developer
            - FRRouting, ExaBGP, etc..
            - https://github.com/slankdev/
    - HN: slankdev

I ❤ both Control-plane, Data-plane

Agenda

- About LINE Corporation and its infrastructure
- Looking back LINE’s Software Defined Networking
  - Pain Point / Case Study / Knowledge

About LINE



Media/Entertainment

LINE NEWS

LINE TIMELINE

LINE TV

LINE TODAY

Fintech

LINE Pay

LINE BITMAX

BITFRONT

LINK

LINE Bank

LINE BK

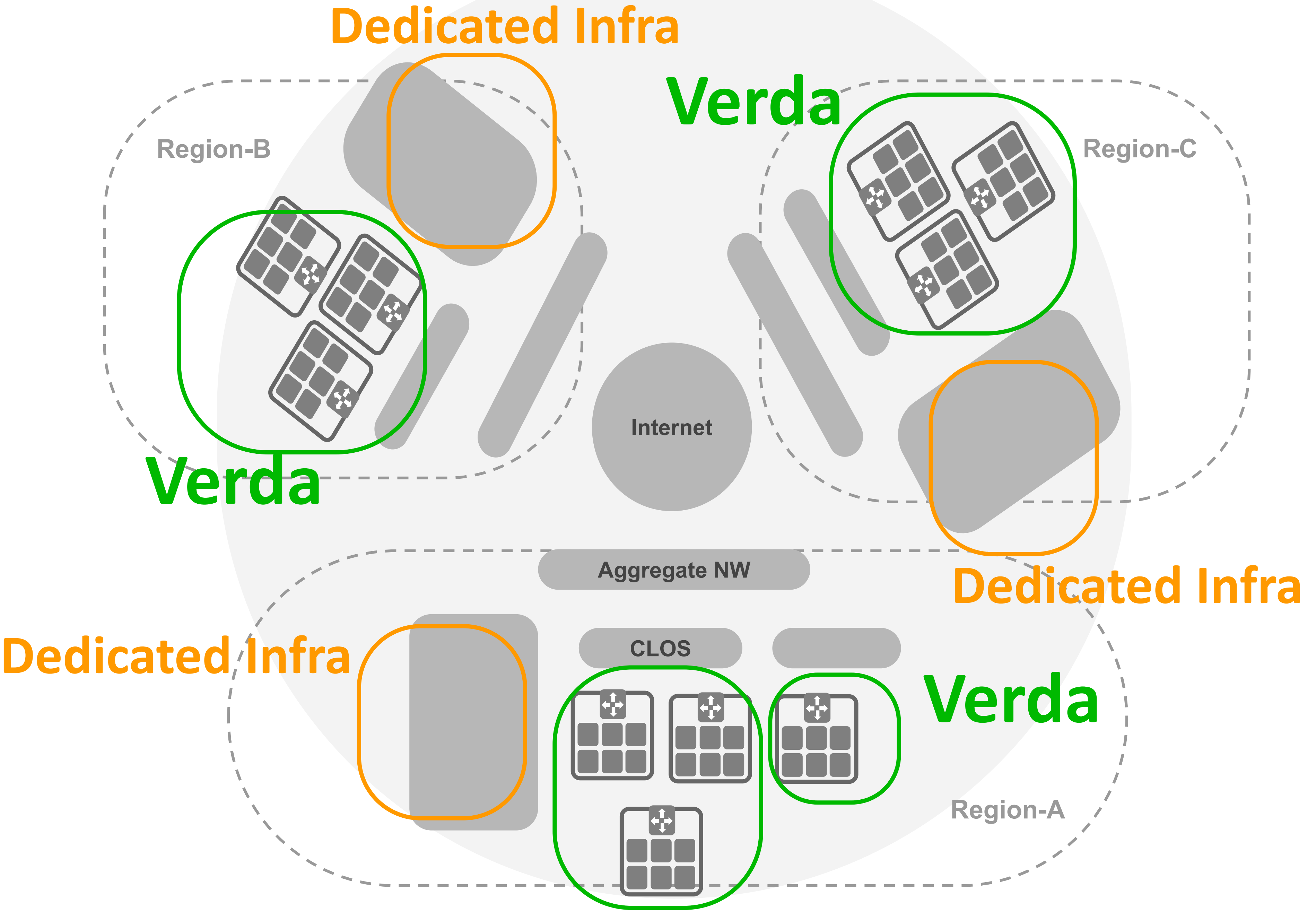
E-Commerce

LINE ショッピング

LINE MAN

Messenger Platform

https://linedevday.linecorp.com/2021/ja/sessions/1



Dedicated Infra

Verda

Region-B

Region-C

Verda

Internet

Aggregate NW

Dedicated Infra

Dedicated Infra

CLOS

Verda

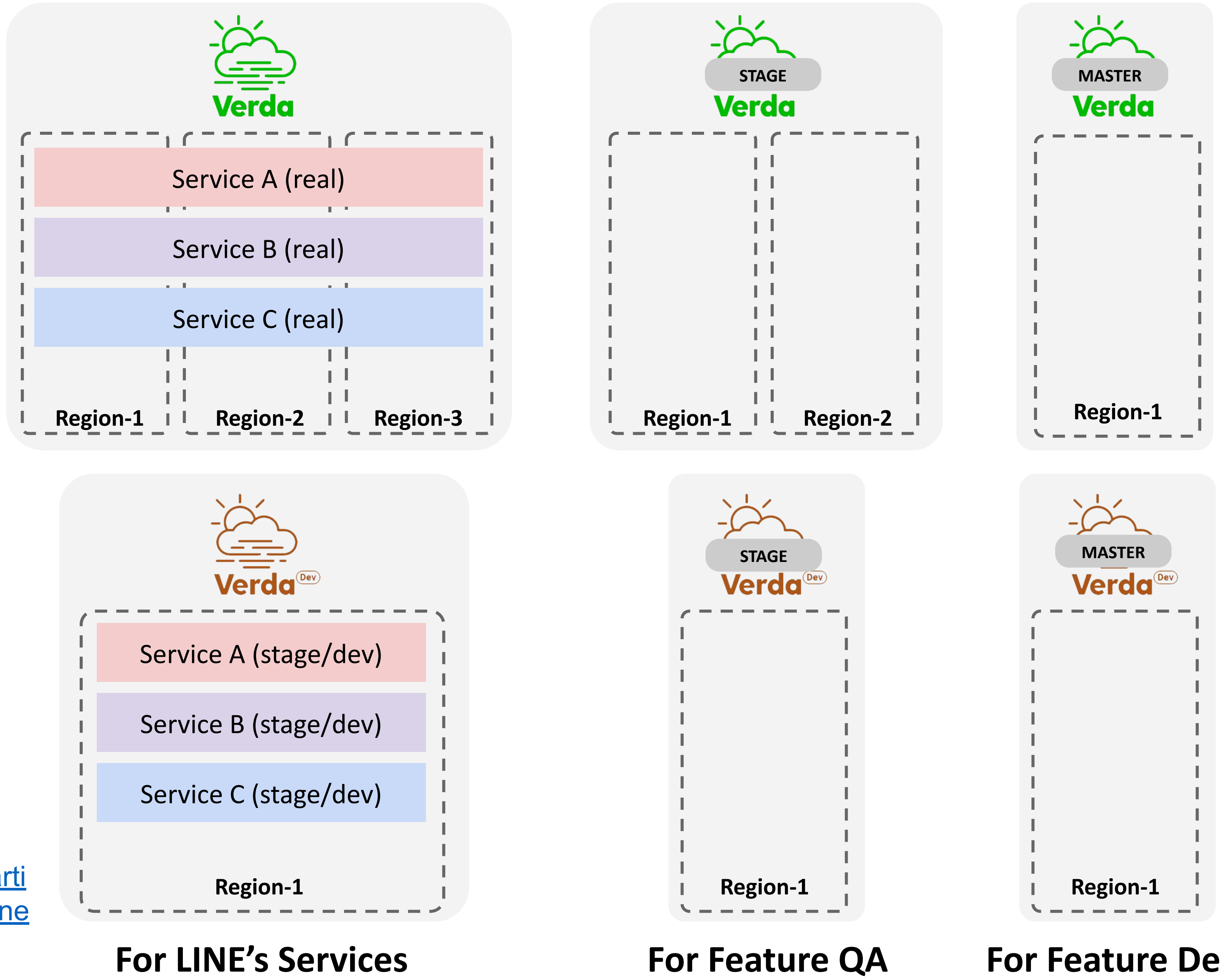
Region-A

Total VMs	**85,000+** (New 10k VMs / Half)
Total PMs	**30,000+**
Total HVs	**4,000+**

Jul. 2021

Verda-Prod

Verda-Dev



Verda

Service A (real)

Service B (real)

Service C (real)

Region-1 Region-2 Region-3

STAGE Verda

Region-1 Region-2

MASTER Verda

Region-1

Verda Dev

Service A (stage/dev)

Service B (stage/dev)

Service C (stage/dev)

Region-1

STAGE Verda Dev

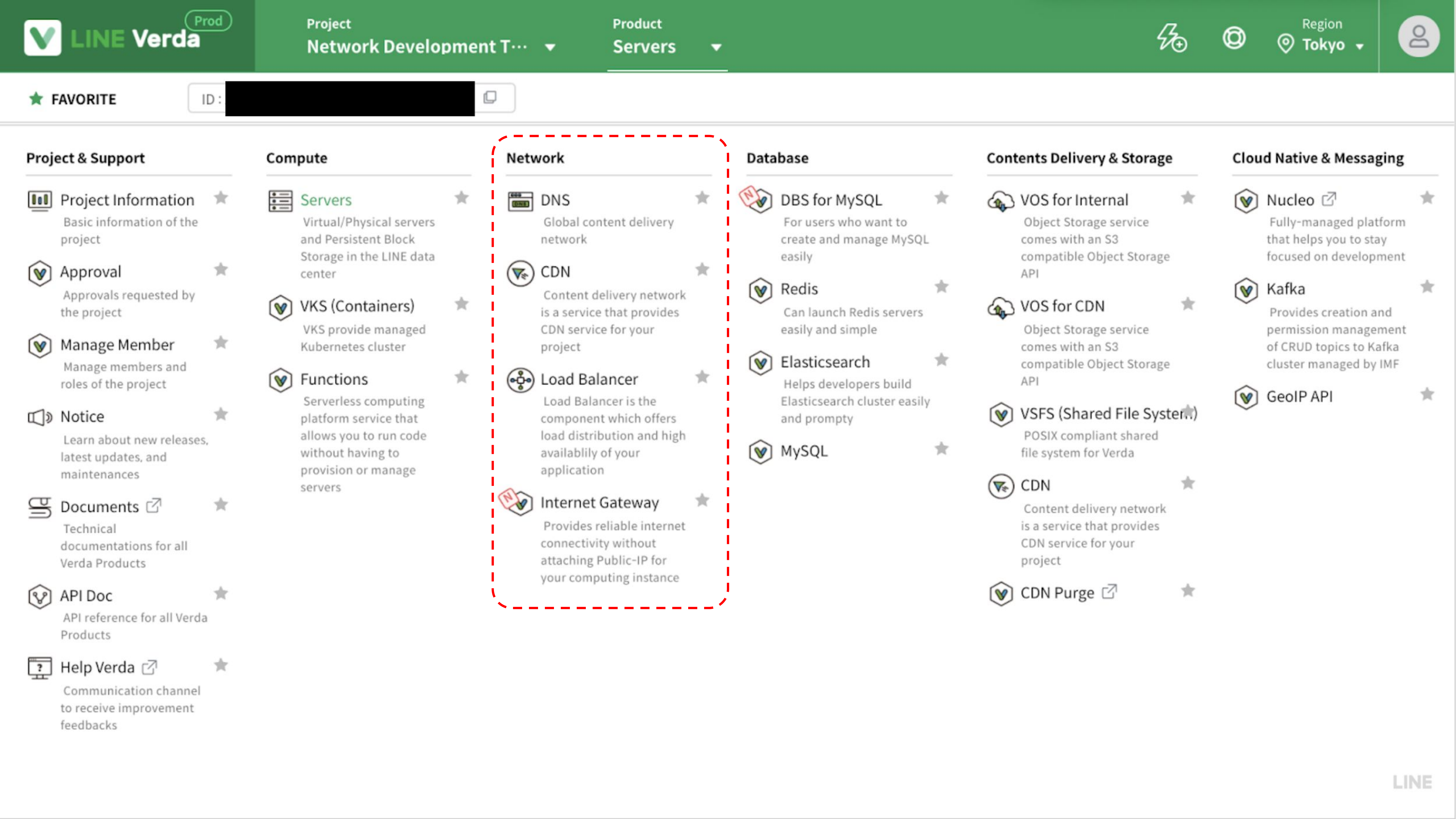
Region-1

MASTER Verda Dev

Region-1

For LINE's Services **For Feature QA** **For Feature Dev**

https://superuser.openstack.org/articles/2020-superuser-award-nominee-line/



LINE Verda Prod

Project
Network Development T…

Product
Servers

Region
Tokyo

FAVORITE

ID : [redacted]

Project & Support

Project Information
Basic information of the project

Approval
Approvals requested by the project

Manage Member
Manage members and roles of the project

Notice
Learn about new releases, latest updates, and maintenances

Documents
Technical documentations for all Verda Products

API Doc
API reference for all Verda Products

Help Verda
Communication channel to receive improvement feedbacks

Compute

Servers
Virtual/Physical servers and Persistent Block Storage in the LINE data center

VKS (Containers)
VKS provide managed Kubernetes cluster

Functions
Serverless computing platform service that allows you to run code without having to provision or manage servers

Network

DNS
Global content delivery network

CDN
Content delivery network is a service that provides CDN service for your project

Load Balancer
Load Balancer is the component which offers load distribution and high availablity of your application

Internet Gateway
Provides reliable internet connectivity without attaching Public-IP for your computing instance

Database

DBS for MySQL
For users who want to create and manage MySQL easily

Redis
Can launch Redis servers easily and simple

Elasticsearch
Helps developers build Elasticsearch cluster easily and prompty

MySQL

Contents Delivery & Storage

VOS for Internal
Object Storage service comes with an S3 compatible Object Storage API

VOS for CDN
Object Storage service comes with an S3 compatible Object Storage API

VSFS (Shared File System)
POSIX compliant shared file system for Verda

CDN
Content delivery network is a service that provides CDN service for your project

CDN Purge

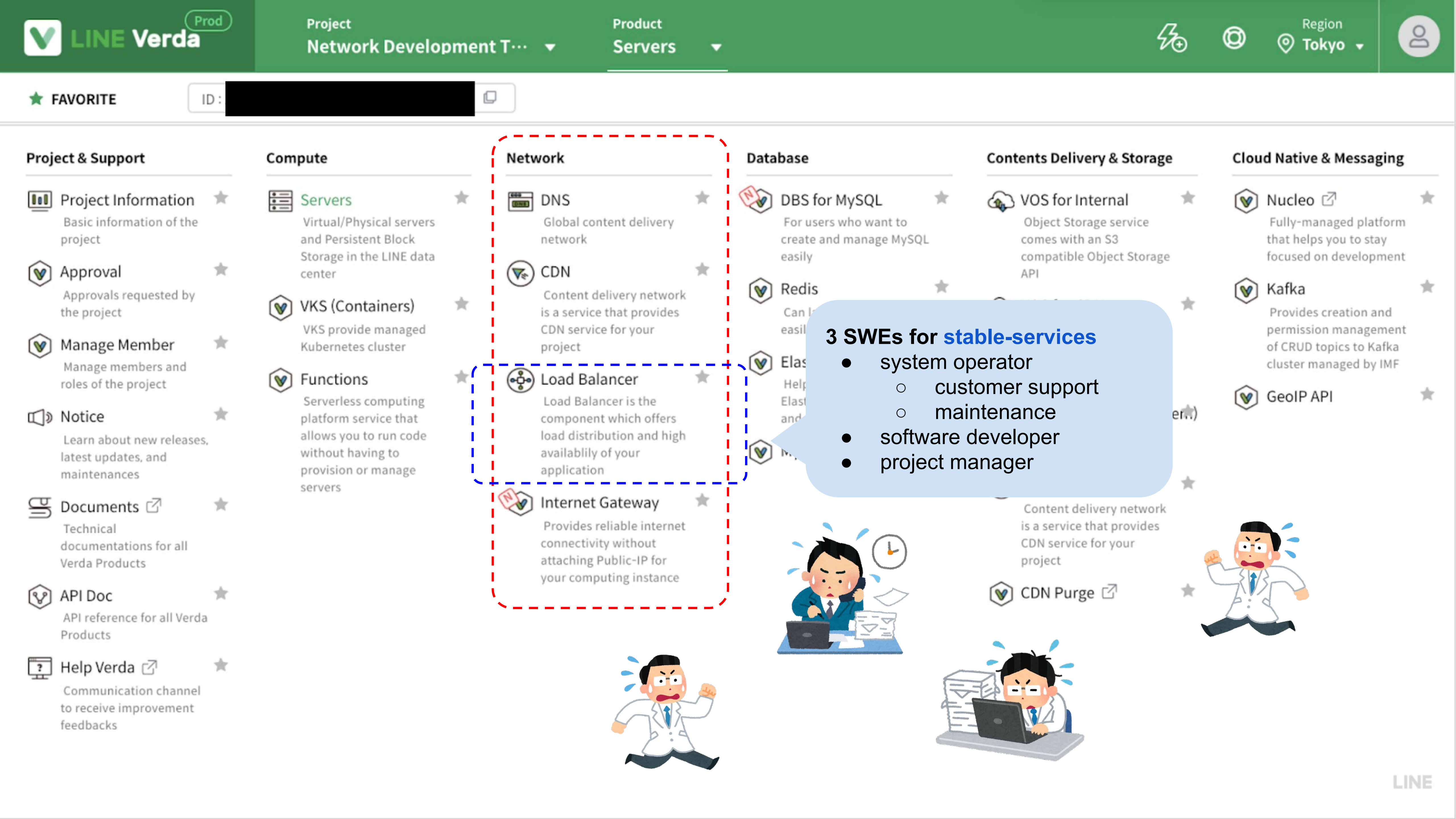
Cloud Native & Messaging

Nucleo
Fully-managed platform that helps you to stay focused on development

Kafka
Provides creation and permission management of CRUD topics to Kafka cluster managed by IMF

GeoIP API

LINE



LINE Verda Prod

Project
Network Development T···

Product
Servers

Region
Tokyo

FAVORITE

ID : [redacted]

Project & Support

Project Information
Basic information of the project

Approval
Approvals requested by the project

Manage Member
Manage members and roles of the project

Notice
Learn about new releases, latest updates, and maintenances

Documents
Technical documentations for all Verda Products

API Doc
API reference for all Verda Products

Help Verda
Communication channel to receive improvement feedbacks

Compute

Servers
Virtual/Physical servers and Persistent Block Storage in the LINE data center

VKS (Containers)
VKS provide managed Kubernetes cluster

Functions
Serverless computing platform service that allows you to run code without having to provision or manage servers

Network

DNS
Global content delivery network

CDN
Content delivery network is a service that provides CDN service for your project

Load Balancer
Load Balancer is the component which offers load distribution and high availablity of your application

Internet Gateway
Provides reliable internet connectivity without attaching Public-IP for your computing instance

Database

DBS for MySQL
For users who want to create and manage MySQL easily

Redis
Can l
easil

Elas
Help
Elas
and

M

Contents Delivery & Storage

VOS for Internal
Object Storage service comes with an S3 compatible Object Storage API

er...)

Content delivery network is a service that provides CDN service for your project

CDN Purge

Cloud Native & Messaging

Nucleo
Fully-managed platform that helps you to stay focused on development

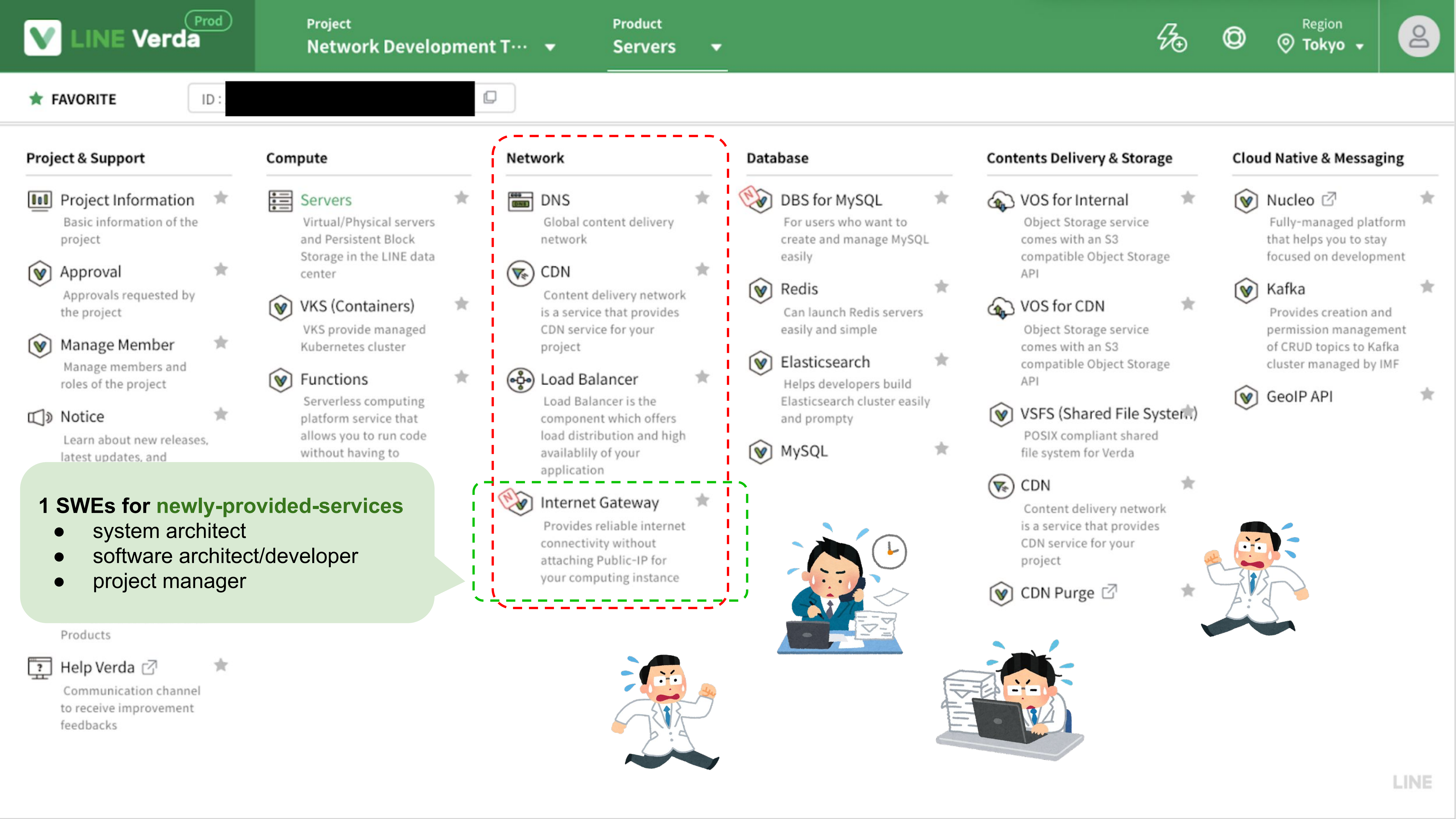
Kafka
Provides creation and permission management of CRUD topics to Kafka cluster managed by IMF

GeoIP API

3 SWEs for stable-services

- system operator
  - customer support
  - maintenance
- software developer
- project manager

LINE



LINE Verda Prod

Project
Network Development T··· ▾

Product
Servers ▾

Region
Tokyo ▾

★ FAVORITE

ID : [redacted]

Project & Support

Project Information ★
Basic information of the project

Approval ★
Approvals requested by the project

Manage Member ★
Manage members and roles of the project

Notice ★
Learn about new releases, latest updates, and

Products

Help Verda ★
Communication channel to receive improvement feedbacks

Compute

Servers ★
Virtual/Physical servers and Persistent Block Storage in the LINE data center

VKS (Containers) ★
VKS provide managed Kubernetes cluster

Functions ★
Serverless computing platform service that allows you to run code without having to

Network

DNS ★
Global content delivery network

CDN ★
Content delivery network is a service that provides CDN service for your project

Load Balancer ★
Load Balancer is the component which offers load distribution and high availablity of your application

Internet Gateway ★
Provides reliable internet connectivity without attaching Public-IP for your computing instance

Database

DBS for MySQL ★
For users who want to create and manage MySQL easily

Redis ★
Can launch Redis servers easily and simple

Elasticsearch ★
Helps developers build Elasticsearch cluster easily and prompty

MySQL ★

Contents Delivery & Storage

VOS for Internal ★
Object Storage service comes with an S3 compatible Object Storage API

VOS for CDN ★
Object Storage service comes with an S3 compatible Object Storage API

VSFS (Shared File Syster..) ★
POSIX compliant shared file system for Verda

CDN ★
Content delivery network is a service that provides CDN service for your project

CDN Purge ★

Cloud Native & Messaging

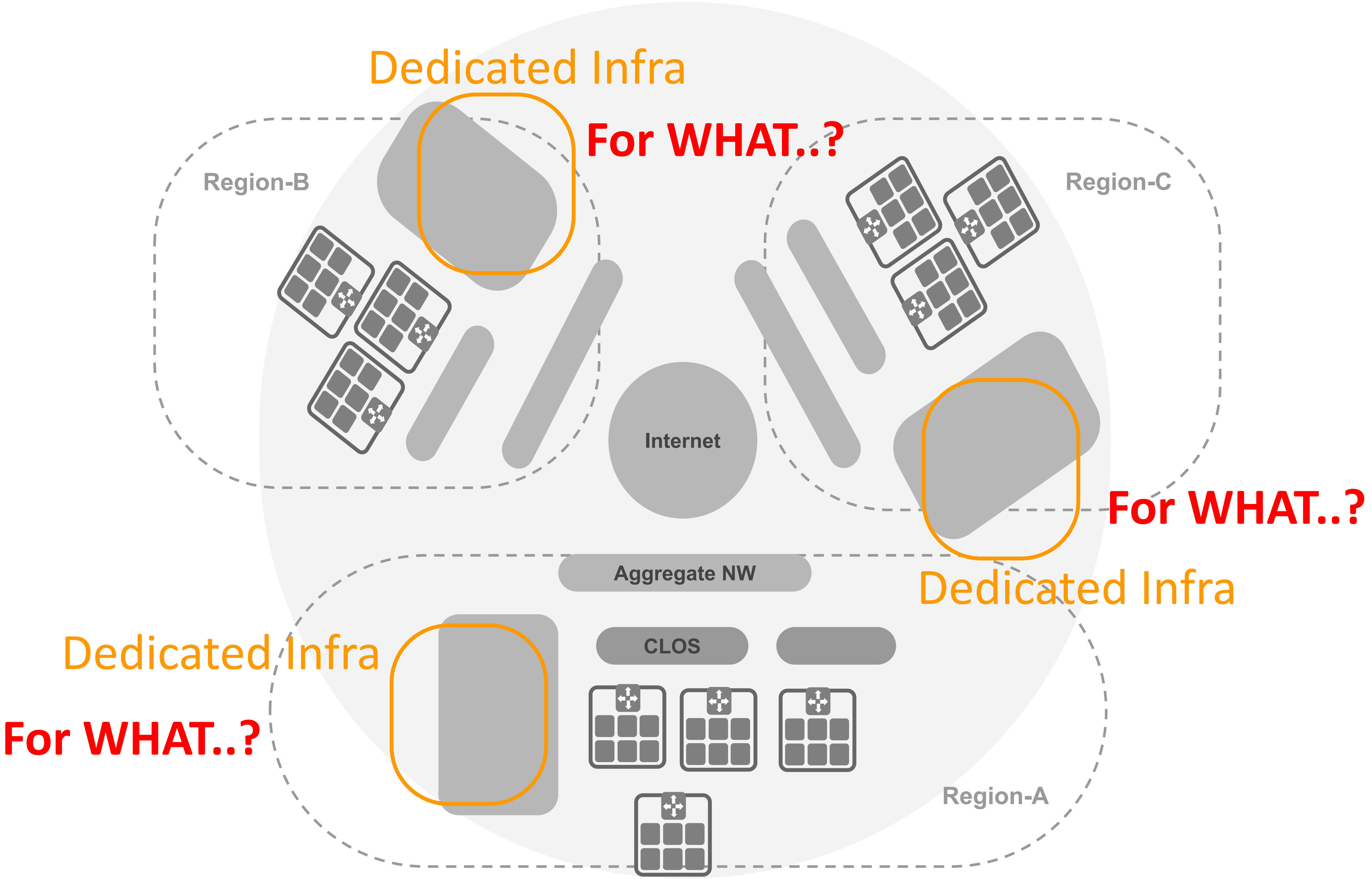
Nucleo ★
Fully-managed platform that helps you to stay focused on development

Kafka ★
Provides creation and permission management of CRUD topics to Kafka cluster managed by IMF

GeoIP API ★

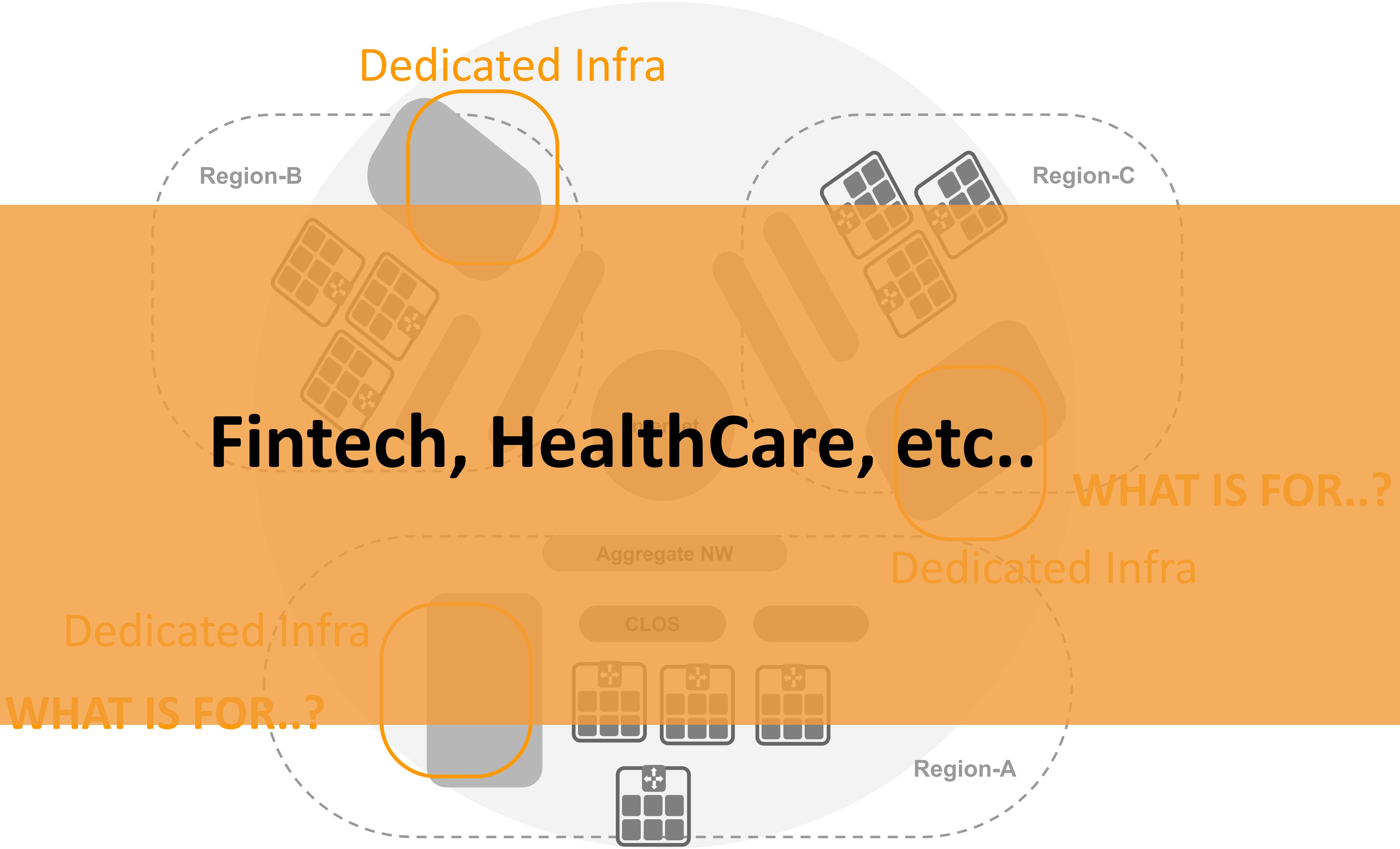
1 SWEs for newly-provided-services

- system architect
- software architect/developer
- project manager


Dedicated Infra
For WHAT..?
Region-B
Region-C
Internet
For WHAT..?
Dedicated Infra
Aggregate NW
Dedicated Infra
CLOS
For WHAT..?
Region-A

LINE

Fintech, HealthCare, etc..



LINE

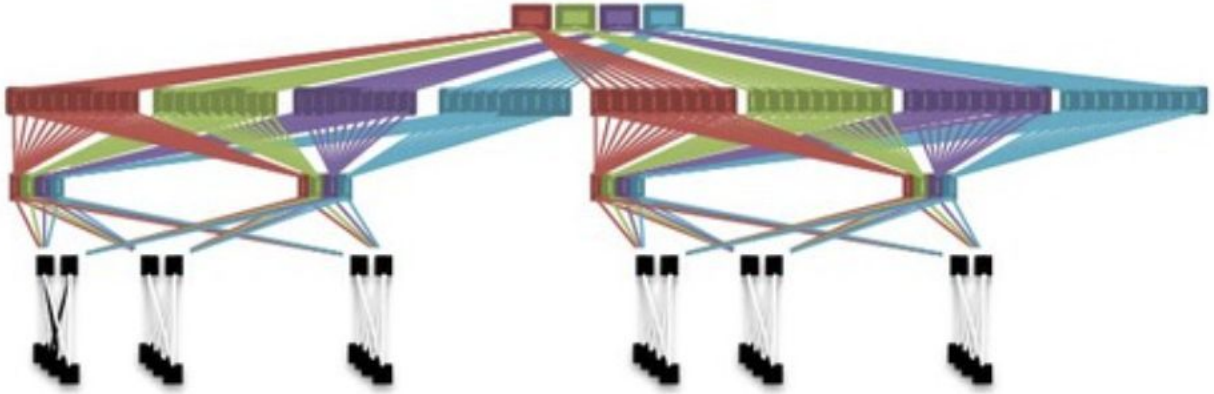
LINE Services and Networks

Latest Infra Challenge

Full L3 CLOS Network*

- Single tenant network
- LINE message service and related services running

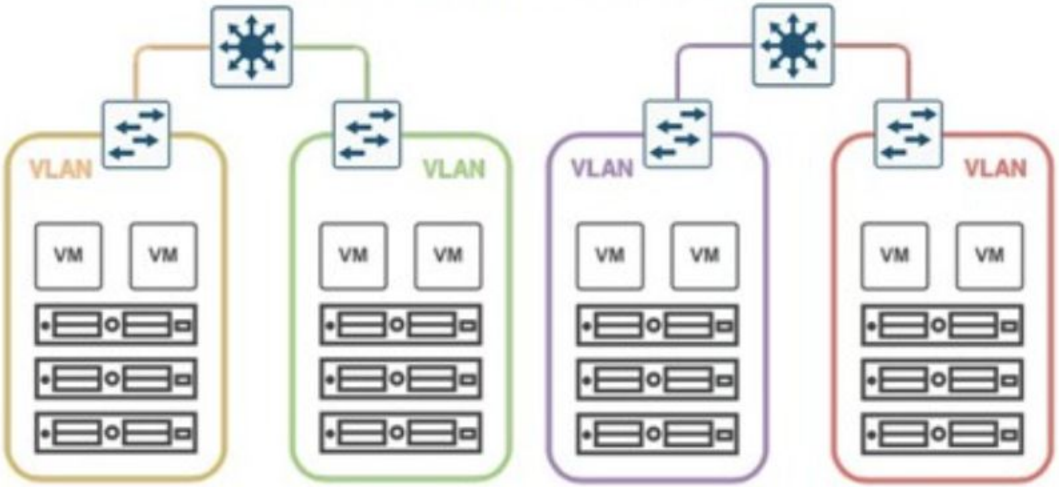
Messaging, Manga, Game, ...



Exclusive Network for Services

- Service with specific requirements running
- Building specific network for each service

Financial, HealthCare ...

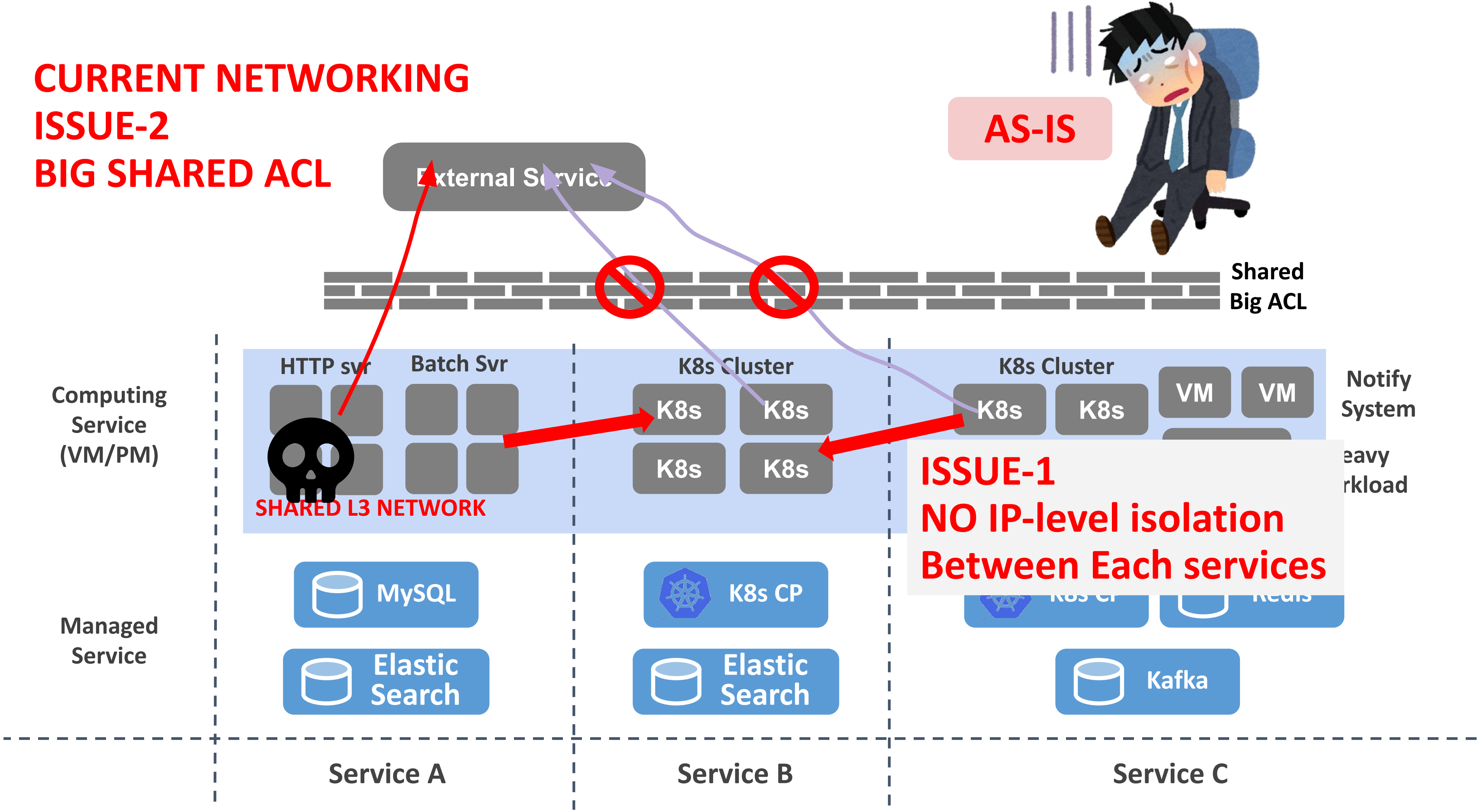


Many fragment underlay networks
Many works to design and build
Management cost increases

* Excitingly simple multi-path OpenStack networking: LAG-less, L2-less, yet fully redundant https://www.slideshare.net/linecorp/excitingly-simple-multipath-openstack-networking-lagless-l2less-yet-fully-redundant

Background:
Virtual Private Cloud is needed

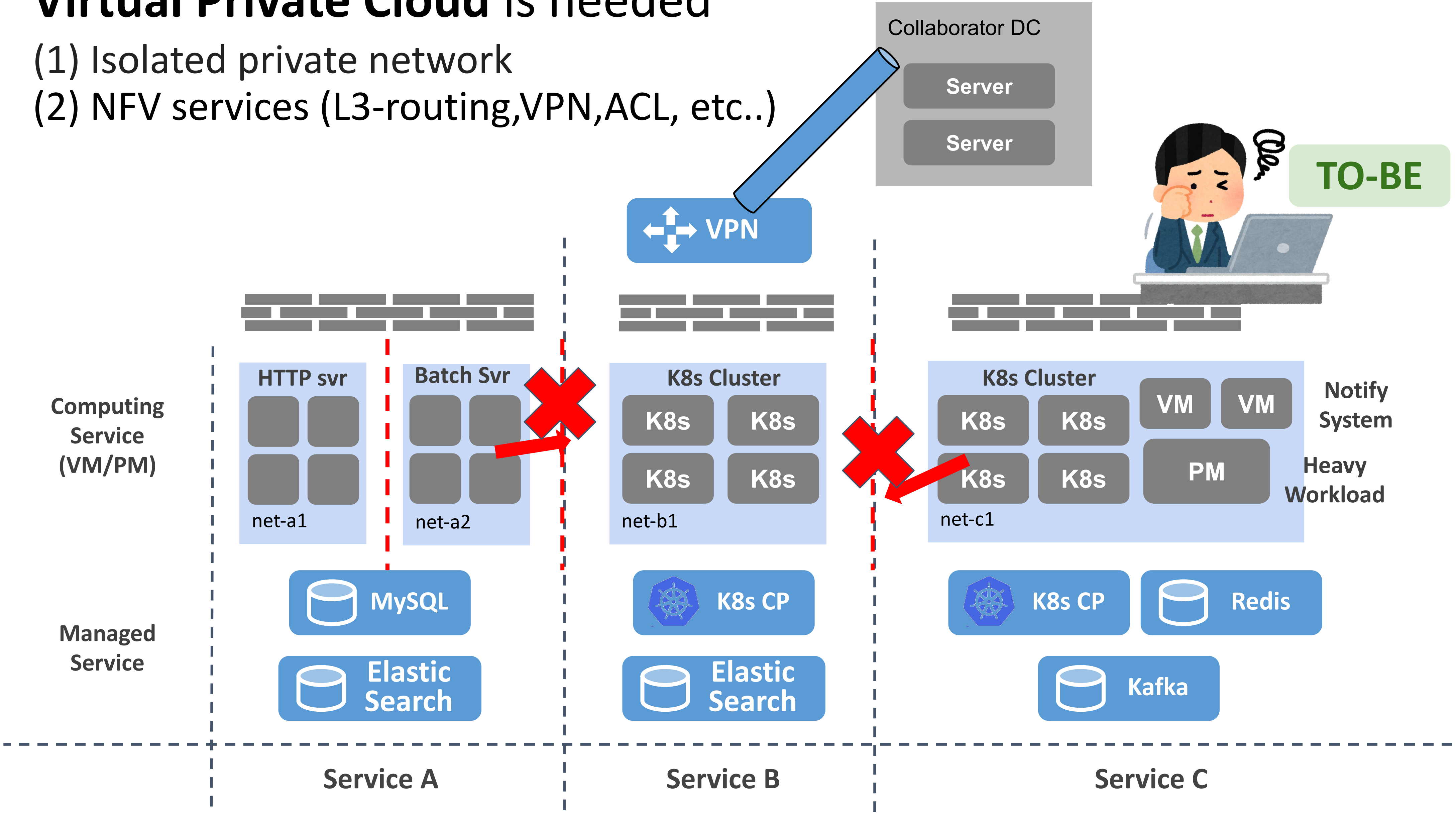
CURRENT NETWORKING
ISSUE-2
BIG SHARED ACL



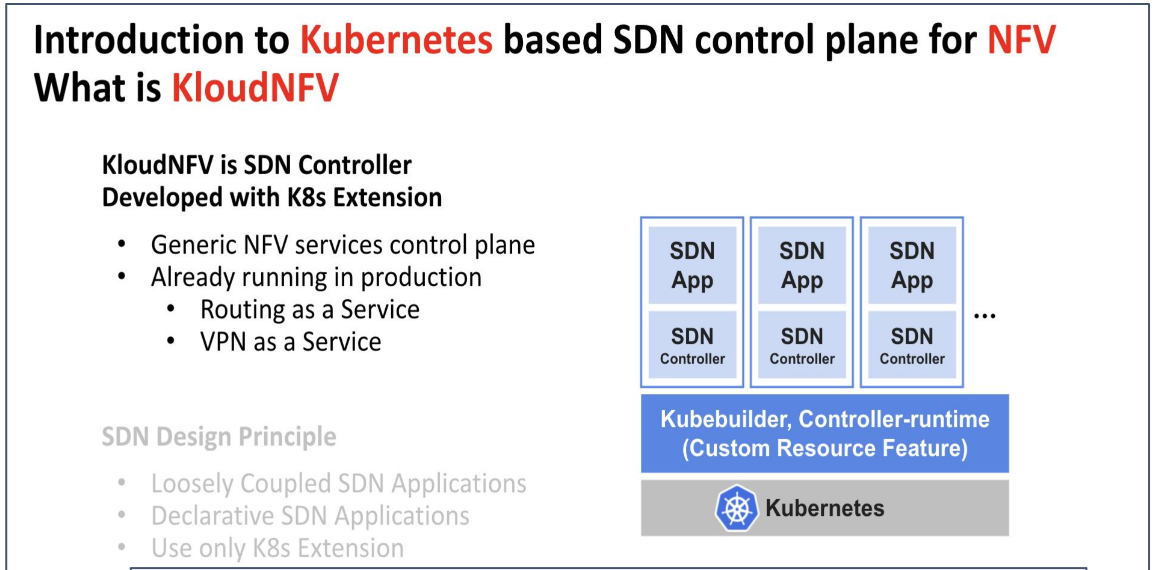
ISSUE-1
NO IP-level isolation
Between Each services

Background:
Virtual Private Cloud is needed

(1) Isolated private network
(2) NFV services (L3-routing,VPN,ACL, etc..)



KloudNFV - Original NFV Service Deployment Platform



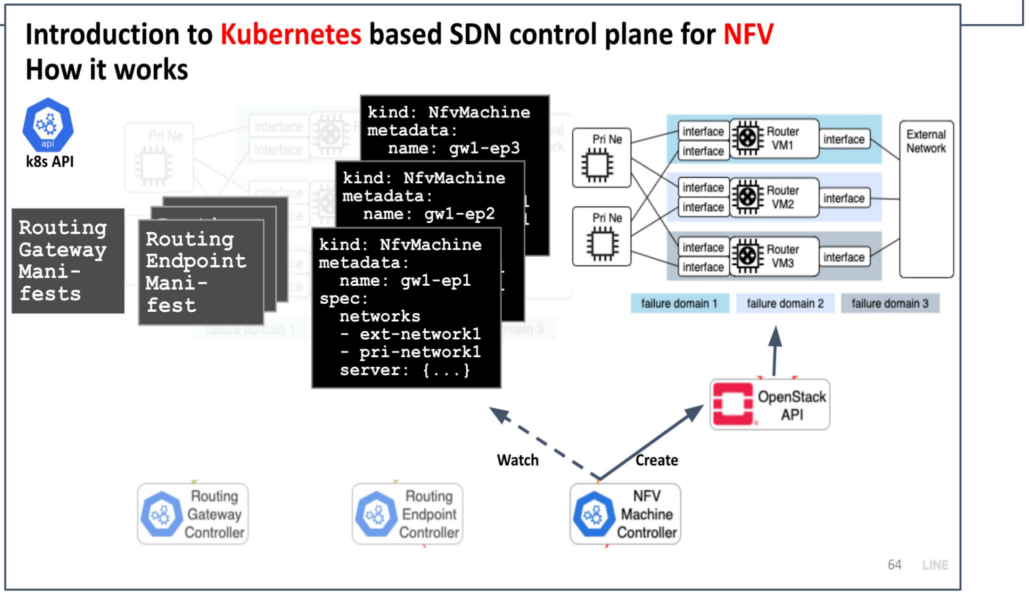
Introduction to Kubernetes based SDN control plane for NFV
What is KloudNFV

KloudNFV is SDN Controller
Developed with K8s Extension

- Generic NFV services control plane
- Already running in production
  - Routing as a Service
  - VPN as a Service

SDN Design Principle

- Loosely Coupled SDN Applications
- Declarative SDN Applications
- Use only K8s Extension



Introduction to Kubernetes based SDN control plane for NFV
How it works

```
Routing
Gateway
Mani-
fests
```

```
Routing
Endpoint
Mani-
fest
```

```
kind: NfvMachine
metadata:
  name: gw1-ep3
```

```
kind: NfvMachine
metadata:
  name: gw1-ep2
```

```
kind: NfvMachine
metadata:
  name: gw1-ep1
spec:
  networks
  - ext-network1
  - pri-network1
  server: {...}
```

ネットワークエンジニアといく
JANOG コミュニティのみんなと作る
JANOG 48

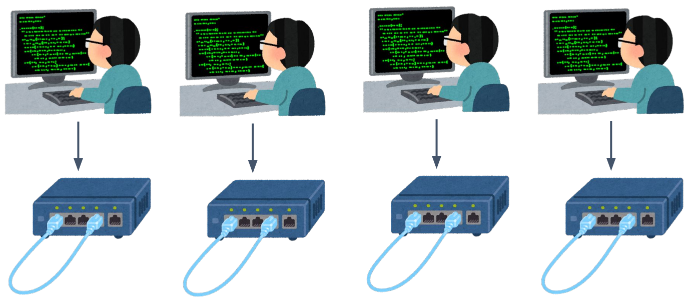
https://youtu.be/bTwTFVgq-1M?t=1108

Looking Back (1)
SRv6 Network SDN

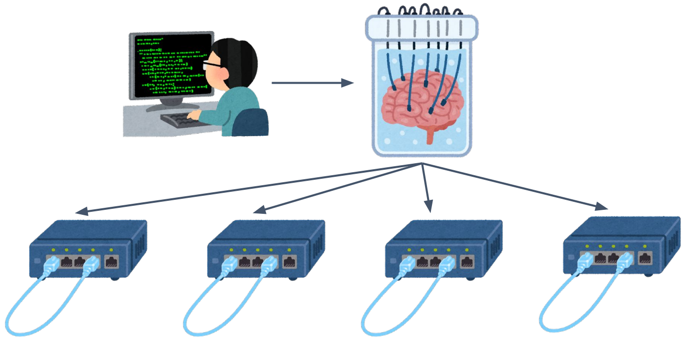
What is SDN, Why we need SDN

- What is Software Defined Networking
  - **Original Software Logic** belongs to **Company’s Business Logics** for **Network Control**
  - Well Known as:
    - **No many Logging-In** to Network Equipment and updating configuration for Network Ops
    - Be able to configure **from Single Point to Many** Network Equipments

- Why we need Software Defined Networking
  - Basically we **love Commodity Logic** instead of Original one
  - Manything can’t be achieved with ONLY Commodity (ex: Automating EVPN, Its Configuration)
  - It’s Difficult to make the Logic to fit for many cases
    - Let’s device actual logic, But let’s unify the interface,database,etc....
    - That is the Sense and Approach of SDN



Without SDN



With SDN

SDN Architecture Variants

- Type-1: Almost Dataplane Configuration is done by SDN
  - SDN agents execute “ip route add xxx” to own network-system
  - Can do anything, but high development cost

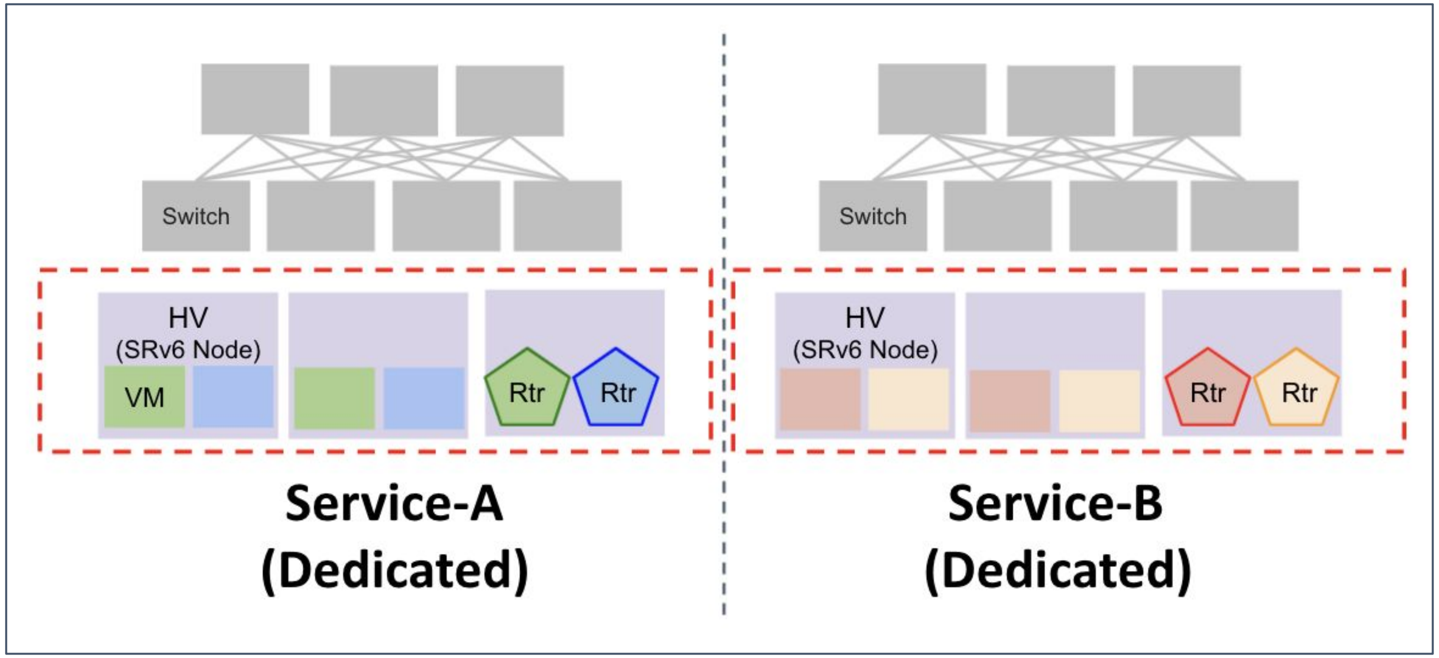
- Type-2: Almost Controlplane(routing-proto) Configuration is done by SDN
  - SDN agents execute “vtysh -c ‘router bgp 1 vrf vrf1’ -c ‘bgp router-id 1.1.1.1’”
  - Some constraint exist, but low development cost
    - Can use existing technology’s strong point
    - ex: health check, maintenance technique, etc..

- Practice: Prioritize “Type-2 -> Type-1”
  - For newer technology (like a srv6) will be used as Type-1
  - Few month/year later, it should be moved as Type-2 in some cases

(*)These are defined for only this presentation

Gen-1,2,3 SRv6 Overlay Network Design

- Gen1: https://www.janog.gr.jp/meeting/janog44/program/srv6/
- Gen2,3: Overlay Network Terminator (Baremetal → vm)
  - Maintenance of virtual router cluster can be controlled by SDN
  - Lower physical equipment per each environment
- Issues
  - HealthCheck & Failover feature development cost and its flexiblity
  - -> Type-1 development cost...



SDN Architecture Variants

- Type-1: Almost Dataplane Configuration is done by SDN
  - SDN agents execute “ip route add xxx” to own network-system
  - Can do anything, but high development cost

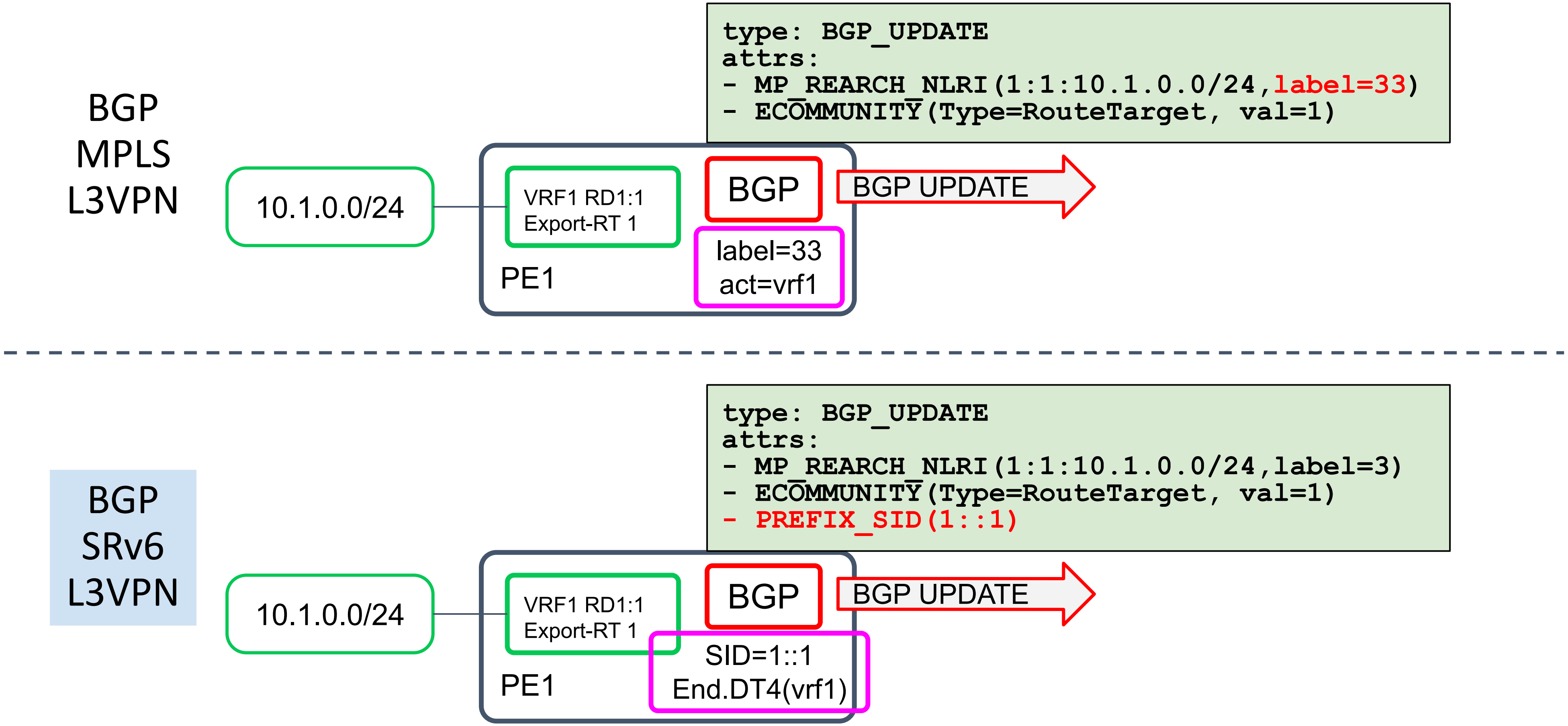
- Type-2: Almost Controlplane(routing-proto) Configuration is done by SDN
  - SDN agents execute “vtysh -c ‘router bgp 1 vrf vrf1’ -c ‘bgp router-id 1.1.1.1’”
  - Some constraint exist, but low development cost
    - Can use existing technology’s strong point
    - ex: health check, maintenance technique, etc..

- Practice: Prioritize “Type-2 -> Type-1”
  - For newer technology (like a srv6) will be used as Type-1
  - Few month/year later, it should be moved as Type-2 in some cases

(*)These are defined for only this presentation

draft-ietf-bess-srv6-services: SRv6 BGP based Overlay Services

- Additional Sub-Type of Prefix SID Path Attribute
  - [new] Type-5: L3VPN Service SID
  - [new] Type-6: L2VPN Service SID
  - Extension of IPVPN(RFC4364), EVPN(RFC7432) to support VPN with SRv6 in addition MPLS



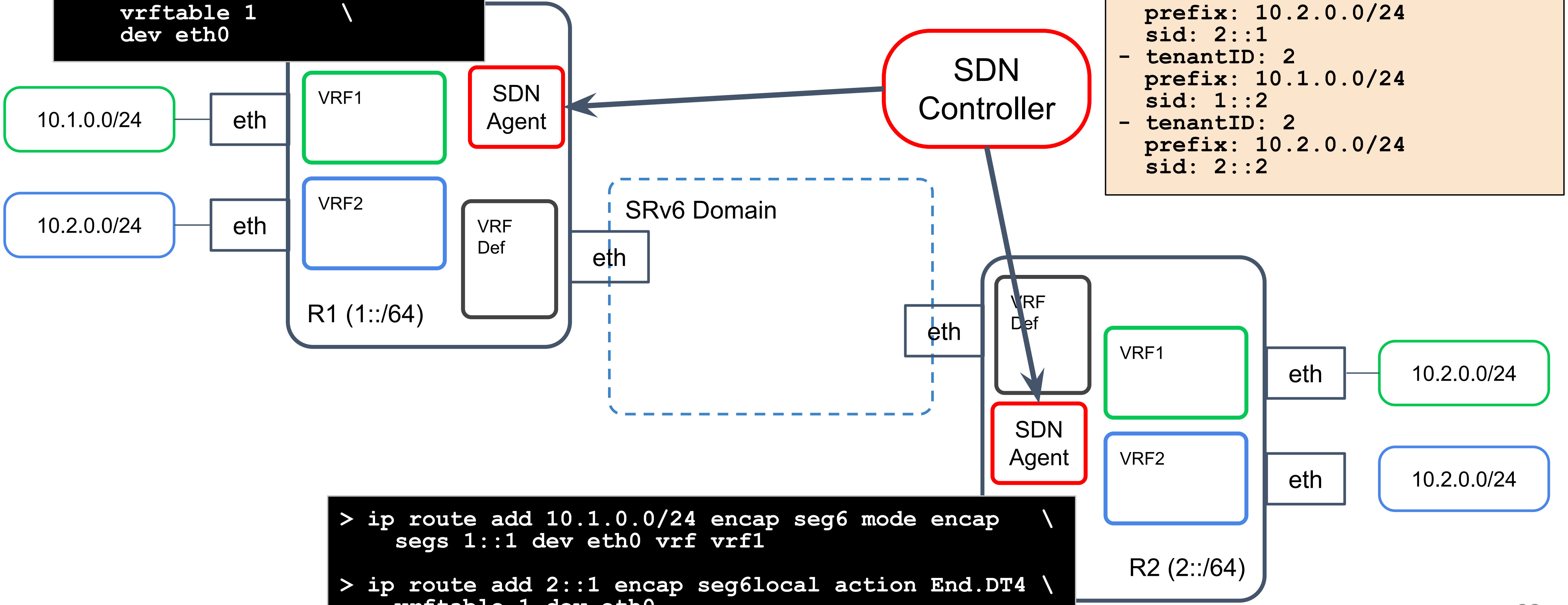
```
type: BGP_UPDATE
attrs:
- MP_REARCH_NLRI(1:1:10.1.0.0/24,label=33)
- ECOMMUNITY(Type=RouteTarget, val=1)
```

```
type: BGP_UPDATE
attrs:
- MP_REARCH_NLRI(1:1:10.1.0.0/24,label=3)
- ECOMMUNITY(Type=RouteTarget, val=1)
- PREFIX_SID(1::1)
```

Type-1 :: IPv6 Routing Proto + SDN Controller

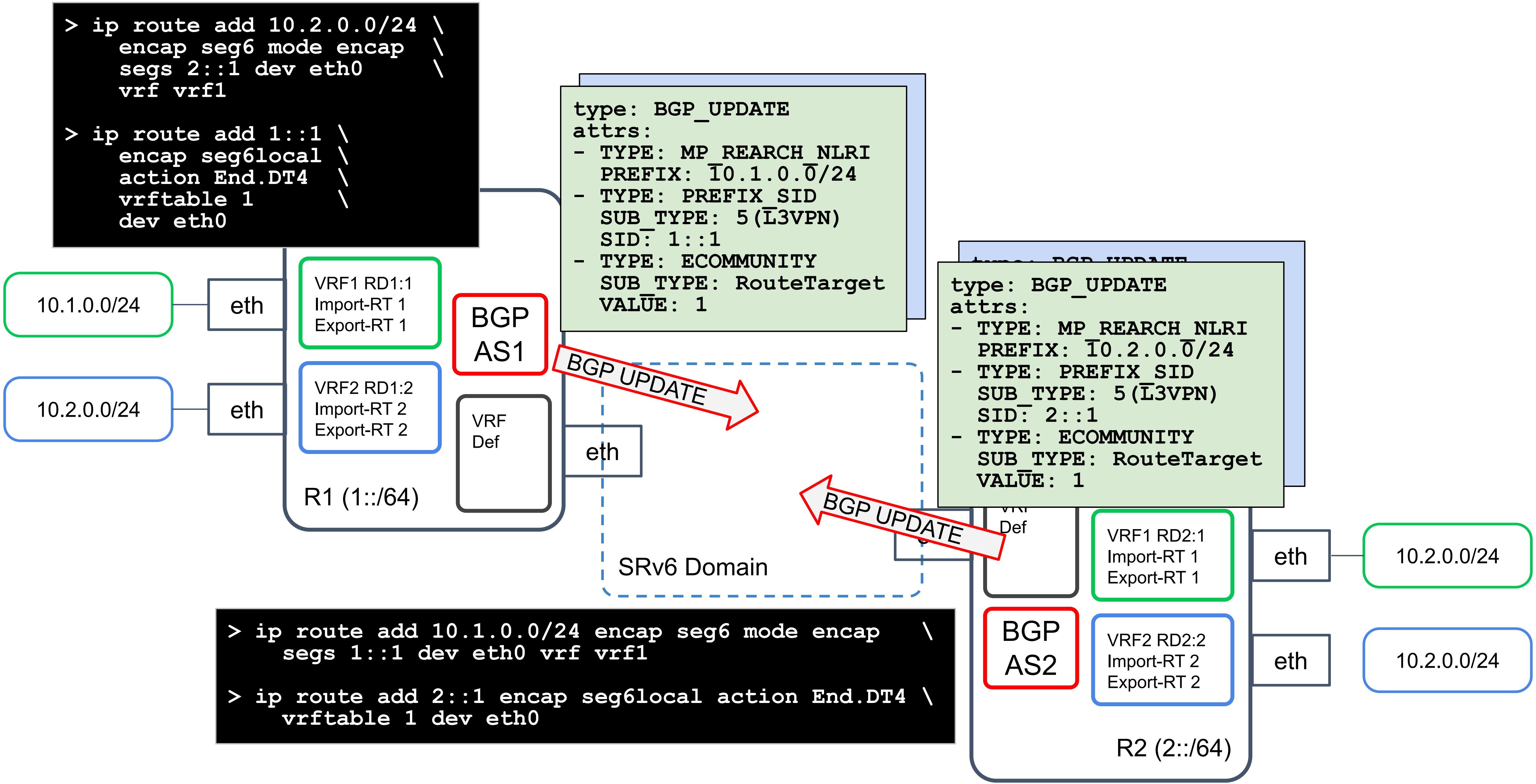
```
> ip route add 10.2.0.0/24 \
    encap seg6 mode encap  \
    segs 2::1 dev eth0     \
    vrf vrf1

> ip route add 1::1 \
    encap seg6local \
    action End.DT4  \
    vrftable 1      \
    dev eth0
```

```
nodes:
- { name: R1, locator: 1::/64 }
- { name: R2, locator: 2::/64 }
networks:
- tenantID: 1
  prefix: 10.1.0.0/24
  sid: 1::1
- tenantID: 1
  prefix: 10.2.0.0/24
  sid: 2::1
- tenantID: 2
  prefix: 10.1.0.0/24
  sid: 1::2
- tenantID: 2
  prefix: 10.2.0.0/24
  sid: 2::2
```



```
> ip route add 10.1.0.0/24 encap seg6 mode encap   \
    segs 1::1 dev eth0 vrf vrf1

> ip route add 2::1 encap seg6local action End.DT4 \
    vrftable 1 dev eth0
```

Type-2 :: All Routing Proto (BGP-SRv6-L3VPN)



```
> ip route add 10.2.0.0/24 \
    encap seg6 mode encap  \
    segs 2::1 dev eth0     \
    vrf vrf1

> ip route add 1::1 \
    encap seg6local \
    action End.DT4  \
    vrftable 1      \
    dev eth0
```

```
type: BGP_UPDATE
attrs:
- TYPE: MP_REARCH_NLRI
  PREFIX: 10.1.0.0/24
- TYPE: PREFIX_SID
  SUB_TYPE: 5(L3VPN)
  SID: 1::1
- TYPE: ECOMMUNITY
  SUB_TYPE: RouteTarget
  VALUE: 1
```

```
type: BGP_UPDATE
attrs:
- TYPE: MP_REARCH_NLRI
  PREFIX: 10.2.0.0/24
- TYPE: PREFIX_SID
  SUB_TYPE: 5(L3VPN)
  SID: 2::1
- TYPE: ECOMMUNITY
  SUB_TYPE: RouteTarget
  VALUE: 1
```

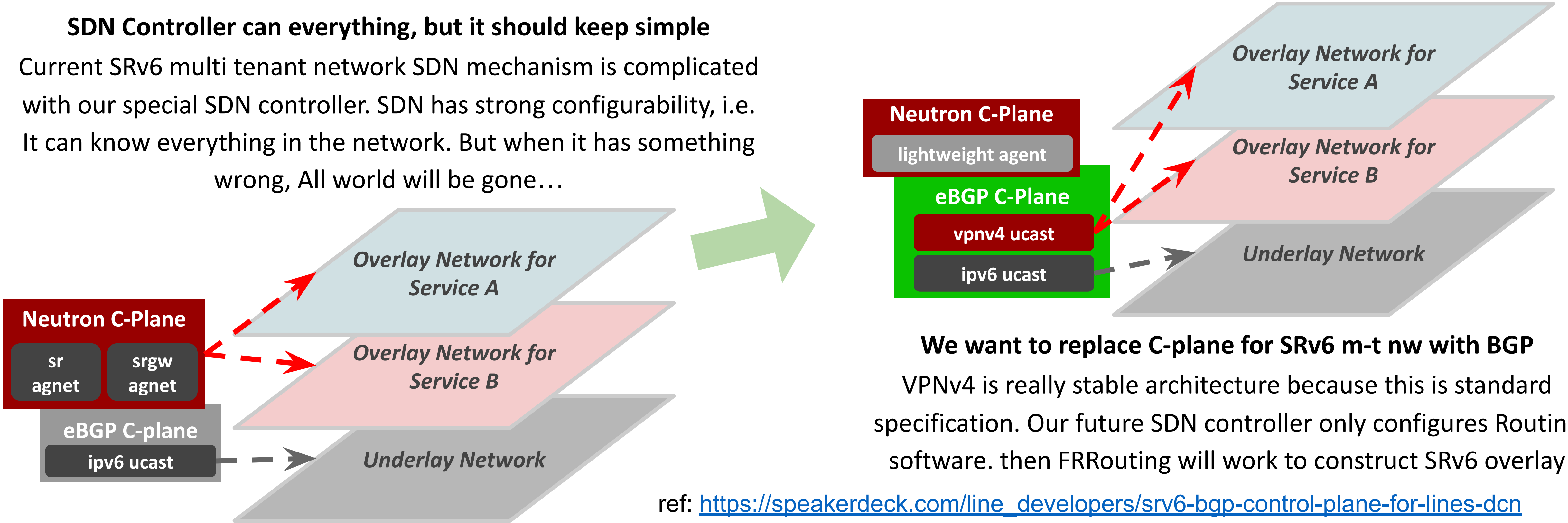
```
> ip route add 10.1.0.0/24 encap seg6 mode encap   \
    segs 1::1 dev eth0 vrf vrf1

> ip route add 2::1 encap seg6local action End.DT4 \
    vrftable 1 dev eth0
```

Gen-4 SRv6 Overlay Network Design

BGP VPNv4 SRv6 for SRv6 Multi-tenant Networking

SDN Controller can everything, but it should keep simple

Current SRv6 multi tenant network SDN mechanism is complicated with our special SDN controller. SDN has strong configurability, i.e. It can know everything in the network. But when it has something wrong, All world will be gone…



We want to replace C-plane for SRv6 m-t nw with BGP

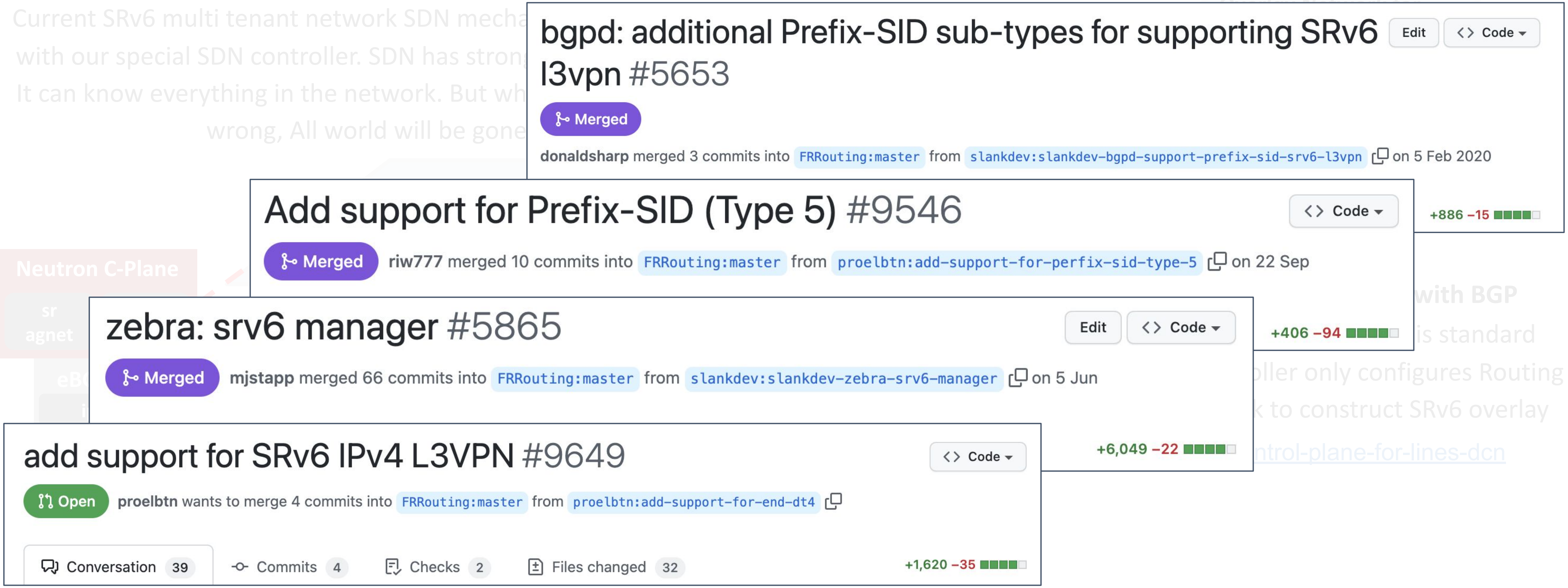
VPNv4 is really stable architecture because this is standard specification. Our future SDN controller only configures Routing software. then FRRouting will work to construct SRv6 overlay

ref: https://speakerdeck.com/line_developers/srv6-bgp-control-plane-for-lines-dcn

Gen-4 SRv6 Overlay Network Design
BGP VPNv4 SRv6 for SRv6 Multi-tenant Networking

SDN Controller can everything, but it should keep simple

Current SRv6 multi tenant network SDN mecha
with our special SDN controller. SDN has stron
It can know everything in the network. But wh
wrong, All world will be gone



bgpd: additional Prefix-SID sub-types for supporting SRv6 l3vpn #5653

Merged

donaldsharp merged 3 commits into `FRRouting:master` from `slankdev:slankdev-bgpd-support-prefix-sid-srv6-l3vpn` on 5 Feb 2020

+886 −15

Add support for Prefix-SID (Type 5) #9546

Merged

riw777 merged 10 commits into `FRRouting:master` from `proelbtn:add-support-for-perfix-sid-type-5` on 22 Sep

+406 −94

zebra: srv6 manager #5865

Merged

mjstapp merged 66 commits into `FRRouting:master` from `slankdev:slankdev-zebra-srv6-manager` on 5 Jun

+6,049 −22

add support for SRv6 IPv4 L3VPN #9649

Open

proelbtn wants to merge 4 commits into `FRRouting:master` from `proelbtn:add-support-for-end-dt4`

Conversation 39 · Commits 4 · Checks 2 · Files changed 32

+1,620 −35

Neutron C-Plane

sr agnet

with BGP
is standard
oller only configures Routing
k to construct SRv6 overlay
ntrol-plane-for-lines-dcn

SDN Architecturing Knowledge(1)
Design Software Automation Aware Network

- Using Commodity Protocol to get simplicity for SDN Logic
  - No inline healthcheck mechanism by SDN Logic
  - No inline failover mechanism by SDN Logic
  - In our case, The commodity specification is already exist
    - VPNv4 with SRv6 backend
    - Of course upstreaming cost was really high

- Another good points:
  - Recruitment, On-boarding, Reusability

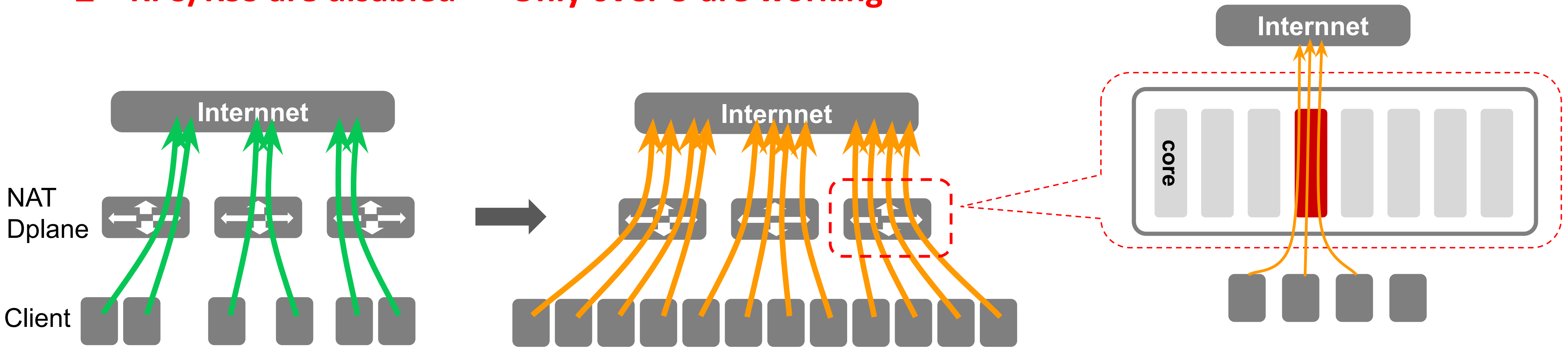
- But if there is no Commodity, we need to consider how to
  - Make commodity? or Wait for commodity? or Type-1?

Looking Back (2)

NAT as a Service

SDN System Architecture Design Knowledge(2)
NAT dplane performance issue and its kernel panic

- About Distributed NAT routing architecture: linedevday/2020/2076 , gihyo/line2021/0002
- Background
    - Increasing users after 1st release
    - There were 6 Linux servers as NAT dplane
        - They are working as act/act, No session state sync
        - 8vCPU/8GB-RAM x6 = 48vCPU
        - **RPS/RSS are disabled → Only 6vCPU are working**

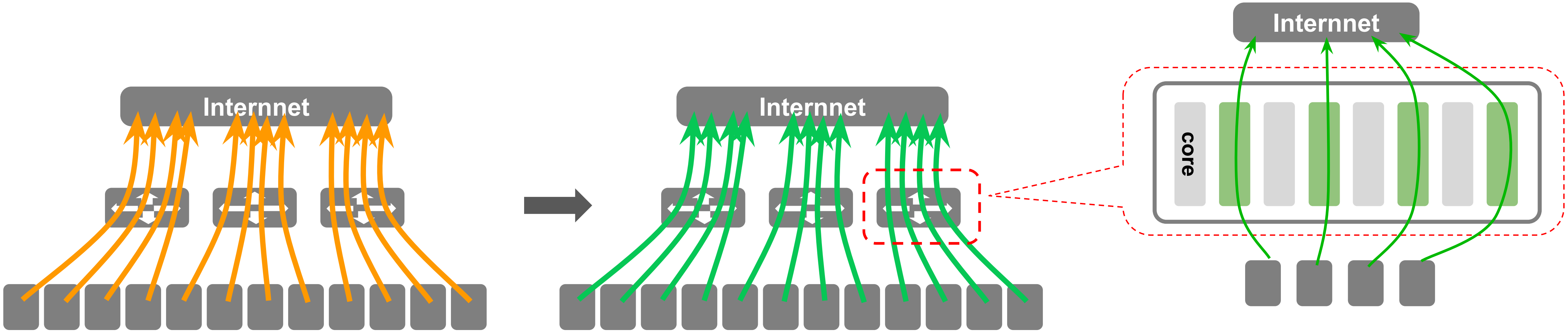


Immediately after release

Increased users

SDN System Architecture Design Knowledge(2)
NAT dplane performance issue and its kernel panic

- We enable **RPS** to use all cores
- Few days later… **weird kernel panics** are occured in some servers
- Few weeks later… All dplane servers are downed one by one, due to the same issue…
  - There are some 秘孔 to make the server downed...

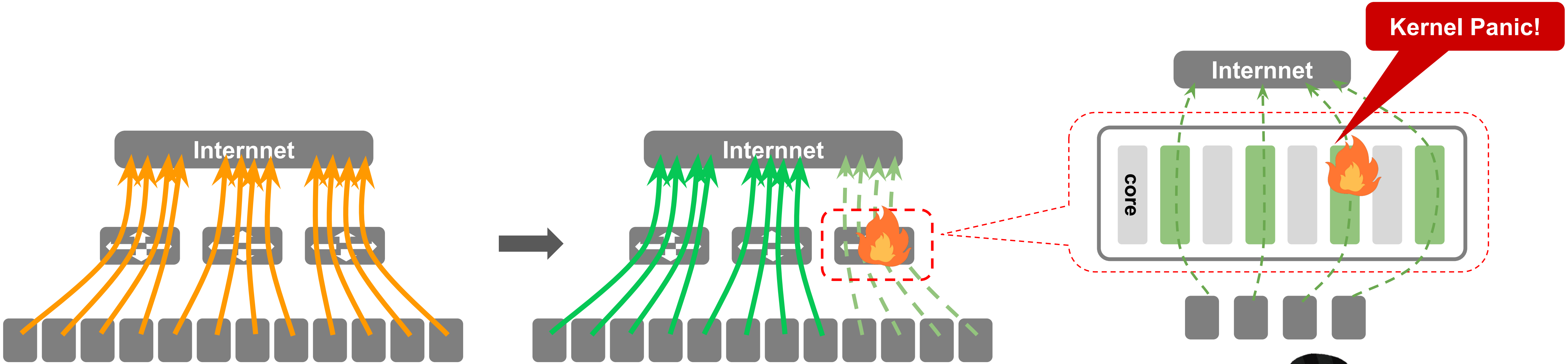


Increased users

RPS enabled

SDN System Architecture Design Knowledge(2)
NAT dplane performance issue and its kernel panic

- We enable **RPS** to use all cores
- Few days later… **weird kernel panics** are occured in some servers
- Few weeks later… All dplane servers are downed one by one, due to the same issue…
  - There are some 秘孔 to make the server downed...



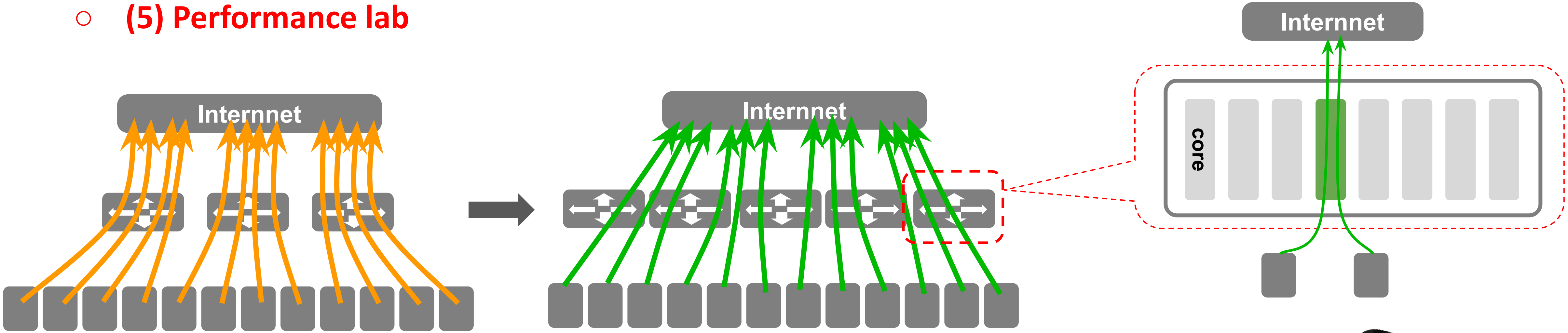
Increased users

RPS enabled

It was HELL...

SDN System Architecture Design Knowledge(2)
NAT dplane performance issue and its kernel panic

- Then, we disalbed RPS again
- And we scaled out dplane nodes **x3** (6 servers → 18 servers)
- **Lesson learned**
  - **(1) If your environment isn’t Majority case, be careful for tuning (LWT-BPF, etc..)**
  - **(2) Scale out is right**
  - **(3) Almost user work-loads were HTTPs/HTTP, It was easy to maintain**
  - **(4) Operation Rehearsal**
  - **(5) Performance lab**



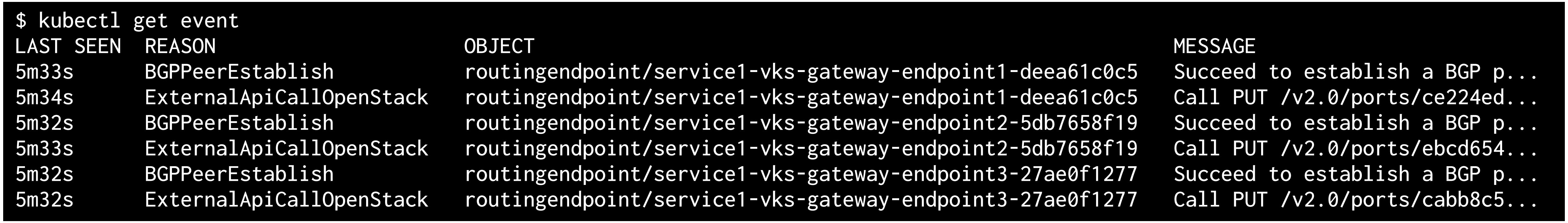
Increased users

Scaled out

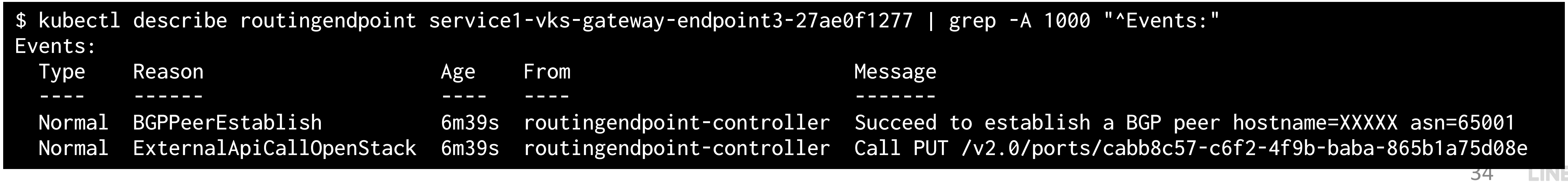
Looking Back (3)
In-House-Dev Team Building

It's ALWAYS been My Turn ?

- Do nothing, but necessary route are disappear from VRF…?
  - Hey Software Developer! What is that…!?
  - Many system (sys-a ⟶ sys-b ⟶ sys-c ⟶ sys-d)
    - sys-a is developed by us
    - sys-b is developed by us
    - sys-c is developed by us
    - sys-d … ah...
- Approach practice: Make it visible what is occured at there



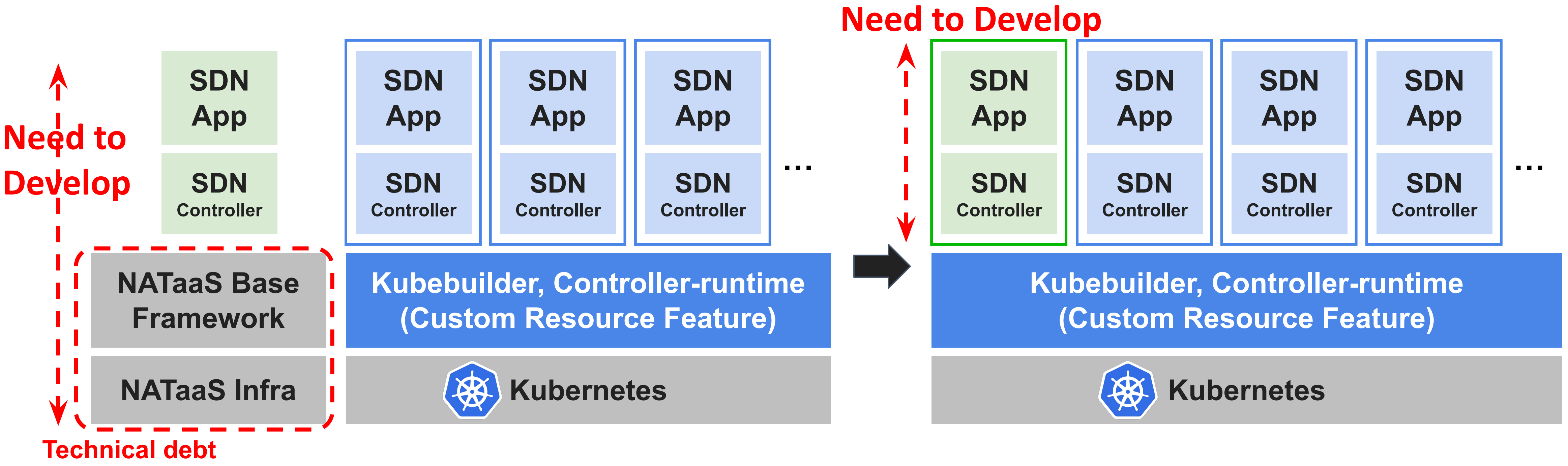
```
$ kubectl get event
LAST SEEN  REASON                   OBJECT                                                 MESSAGE
5m33s      BGPPeerEstablish         routingendpoint/service1-vks-gateway-endpoint1-deea61c0c5   Succeed to establish a BGP p...
5m34s      ExternalApiCallOpenStack   routingendpoint/service1-vks-gateway-endpoint1-deea61c0c5   Call PUT /v2.0/ports/ce224ed...
5m32s      BGPPeerEstablish         routingendpoint/service1-vks-gateway-endpoint2-5db7658f19   Succeed to establish a BGP p...
5m33s      ExternalApiCallOpenStack   routingendpoint/service1-vks-gateway-endpoint2-5db7658f19   Call PUT /v2.0/ports/ebcd654...
5m32s      BGPPeerEstablish         routingendpoint/service1-vks-gateway-endpoint3-27ae0f1277   Succeed to establish a BGP p...
5m32s      ExternalApiCallOpenStack   routingendpoint/service1-vks-gateway-endpoint3-27ae0f1277   Call PUT /v2.0/ports/cabb8c5...
```



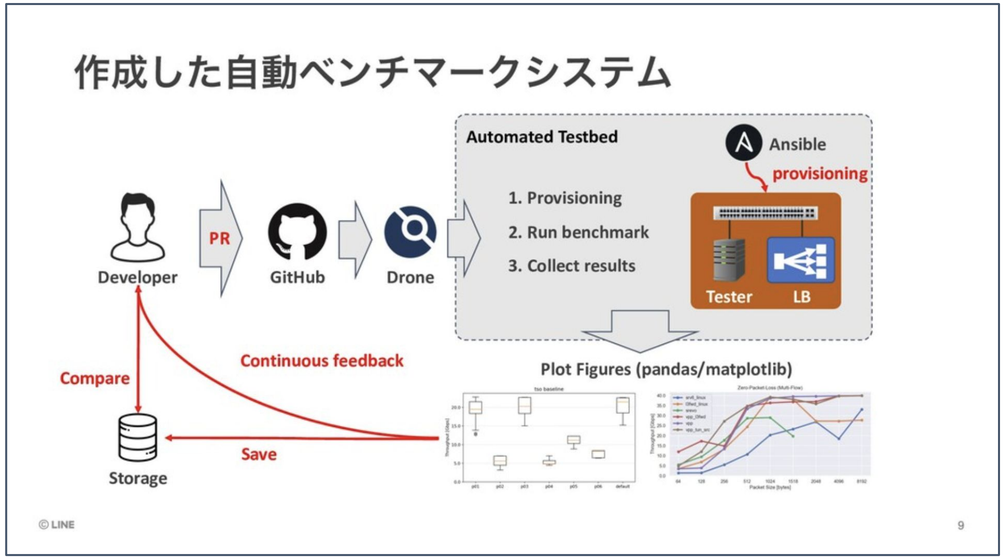
```
$ kubectl describe routingendpoint service1-vks-gateway-endpoint3-27ae0f1277 | grep -A 1000 "^Events:"
Events:
  Type    Reason                    Age    From                        Message
  ----    ------                    ----   ----                        -------
  Normal  BGPPeerEstablish          6m39s  routingendpoint-controller  Succeed to establish a BGP peer hostname=XXXXX asn=65001
  Normal  ExternalApiCallOpenStack  6m39s  routingendpoint-controller  Call PUT /v2.0/ports/cabb8c57-c6f2-4f9b-baba-865b1a75d08e
```

Develop Unify Platform for next development to Make development easier, faster and stabler

- Develop The System for the system
- ex: Restructure current Internet Gateway service with KloudNFV



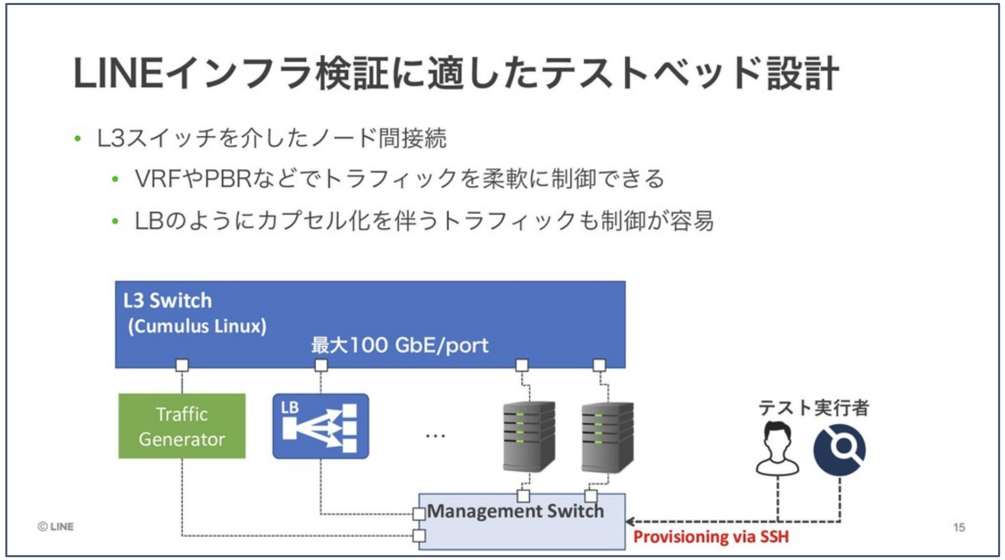
Performance Lab for In-House development



作成した自動ベンチマークシステム

© LINE

9

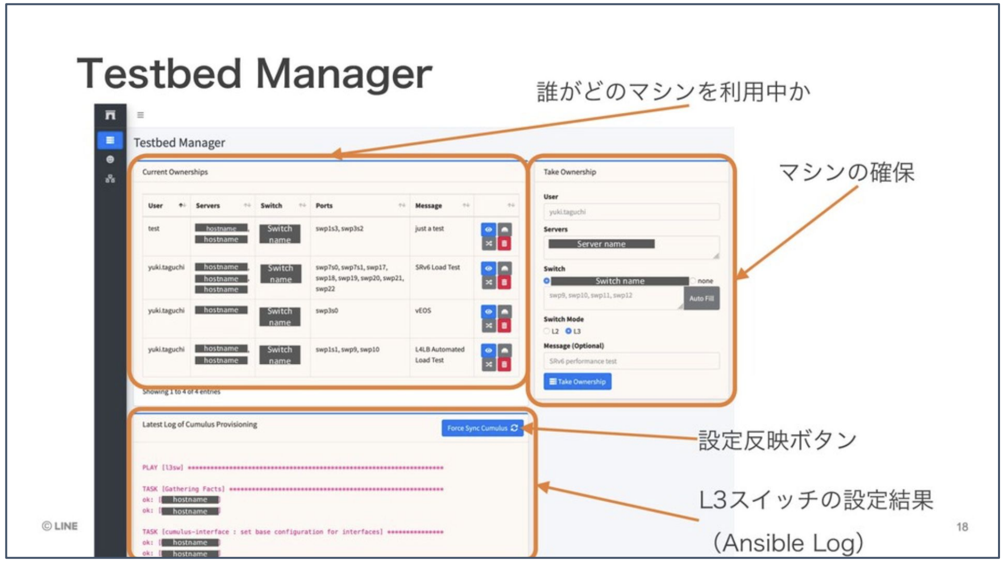


LINEインフラ検証に適したテストベッド設計

- L3スイッチを介したノード間接続
  - VRFやPBRなどでトラフィックを柔軟に制御できる
  - LBのようにカプセル化を伴うトラフィックも制御が容易

© LINE

15



Testbed Manager

誰がどのマシンを利用中か

マシンの確保

設定反映ボタン

L3スイッチの設定結果

（Ansible Log）

Testbed Manager

Current Ownerships

User	Servers	Switch	Ports	Message
test	hostname, hostname	Switch name	swp1s3, swp3s2	just a test
yuki.taguchi	hostname, hostname, hostname	Switch name	swp7s0, swp7s1, swp17, swp18, swp19, swp20, swp21, swp22	SRv6 Load Test
yuki.taguchi	hostname	Switch name	swp3s0	vEOS
yuki.taguchi	hostname, hostname	Switch name	swp1s1, swp9, swp10	L4LB Automated Load Test

Showing 1 to 4 of 4 entries

Take Ownership

User

yuki.taguchi

Servers: Server name

Switch: ● Switch name ○ none

swp9, swp10, swp11, swp12 Auto Fill

Switch Mode: ○ L2 ● L3

Message (Optional): SRv6 performance test

Take Ownership

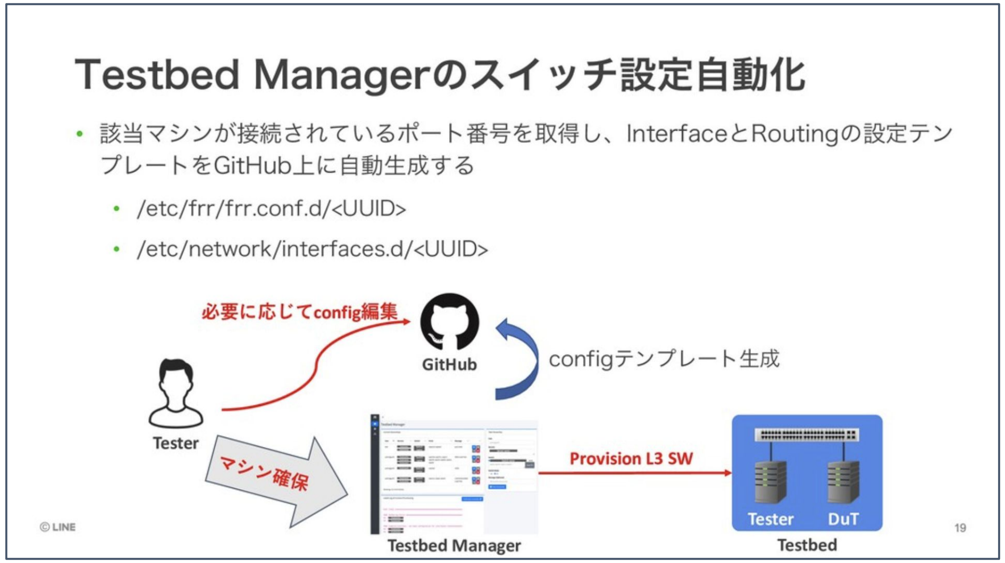
Latest Log of Cumulus Provisioning

Force Sync Cumulus

```
PLAY [l3sw] ****************************************************************

TASK [Gathering Facts] *****************************************************
ok: [ hostname ]
ok: [ hostname ]

TASK [cumulus-interface : set base configuration for interfaces] ***********
ok: [ hostname ]
ok: [ hostname ]
```

© LINE

18



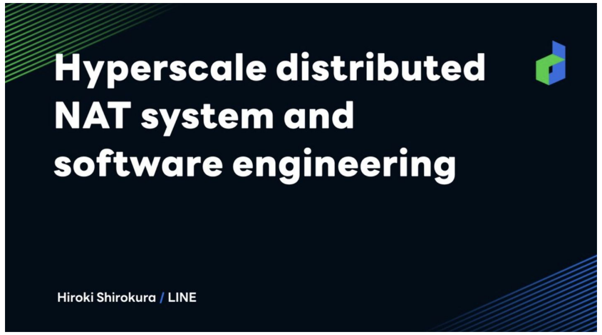
Testbed Managerのスイッチ設定自動化

- 該当マシンが接続されているポート番号を取得し、InterfaceとRoutingの設定テンプレートをGitHub上に自動生成する
  - /etc/frr/frr.conf.d/<UUID>
  - /etc/network/interfaces.d/<UUID>

© LINE

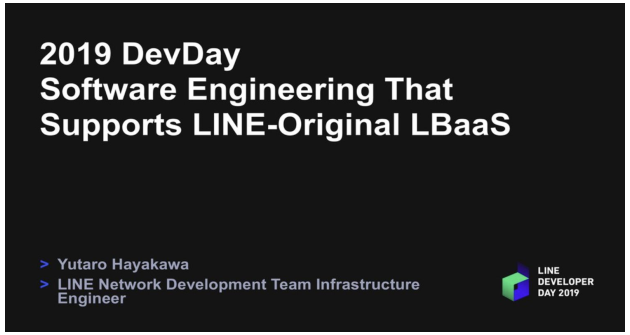
19

Many Network/Software Challenges (again)


Hyperscale distributed NAT system and software engineering

Hiroki Shirokura / LINE

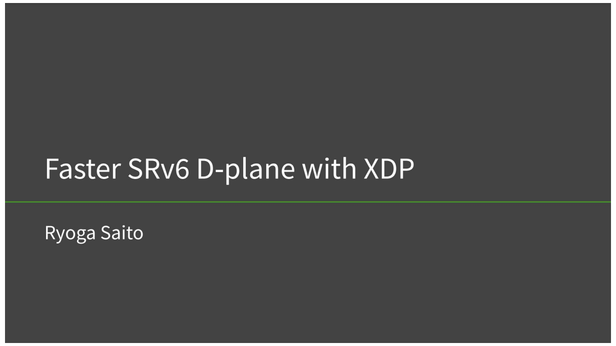
linedevday/2020/sessions/2076


2019 DevDay
Software Engineering That Supports LINE-Original LBaaS

> Yutaro Hayakawa
> LINE Network Development Team Infrastructure Engineer

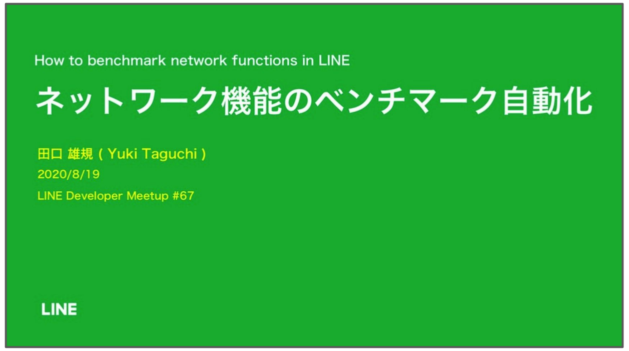
LINE DEVELOPER DAY 2019

linedevday/2019/sessions/F1-7


Faster SRv6 D-plane with XDP

Ryoga Saito

janog45/srv6xdp


How to benchmark network functions in LINE

ネットワーク機能のベンチマーク自動化

田口 雄規 (Yuki Taguchi)
2020/8/19
LINE Developer Meetup #67

LINE

line.connpass/184927

2019 DevDay
LINE's Next-Generation SDN Architecture

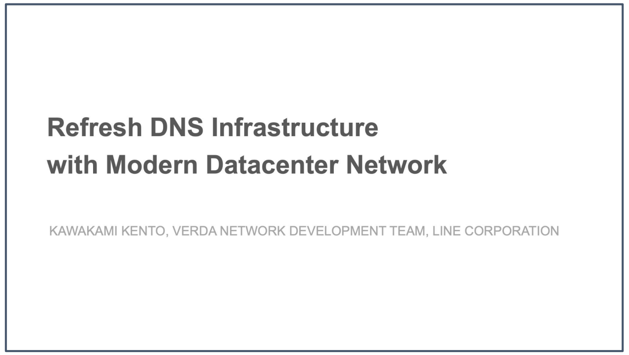
> Toshiki Tsuchiya
> LINE Service Network Team Infra Engineer

LINE DEVELOPER DAY 2019

linedevday/2019/sessions/E1-2

High Functional Cloud NFV System Design & Implementation
@ LINE Cloud
Verda Network Development Team, LINE Corporation
Hiroki Shirokura

janog48/linenfv


Refresh DNS Infrastructure with Modern Datacenter Network

KAWAKAMI KENTO, VERDA NETWORK DEVELOPMENT TEAM, LINE CORPORATION

janog48/linedns

LINEのネットワークオーケストレーション

Verda室 ネットワーク開発チーム 土屋俊貴

line.connpass/184927

Designing/Implementing Multi-tenancy Data Center Networking with SRv6 in Large Scale Platform

Hirofumi Ichihara
LINE corporation

nvidia/gtc

LINEのネットワークを
ゼロから再設計した話
JANOG43 Meeting 2019/01/24

Masayuki Kobayashi
LINE Corporation

janog43/line

Rapid Evolution Challenge @ LINE's Cloud
Verda Network Development Team, LINE Corporation
slankdev / Hiroki Shirokura
WIDE Meeting 2020.12.12

90min → 60 (±10) min session (& discussion), 30 min discussion

wide meeting 2019

Summary

- Many Infrastructure Challenges at LINE
  - Large scale private cloud
  - Fintech/HealthCare support
  - Many Original systems

- Automation/SDN aware system/network/team design
  - Use existing control plane if we can
  - Upstream control plane if we can
  - Scale out is right
  - System for the system

- Q: Software Engineer do it? Network Engineer do it?
- A: Both senses are needed
  - What is critical? What is pain point? by architectural level
  - Act-Stb, Act-Act, 2N, N+1, Blast-radius, Extensibility, Scalability