FUJITSU

shaping tomorrow with you

OpenStack Cluster Zero-Downtime Upgrade ft. Kolla

2017 May 11

Duong Ha-Quang and Hieu LE
Fujitsu Vietnam Limited

Who are we?

FUJITSU

- Duong Ha-Quang
  - Software Engineer at Fujitsu Vietnam
  - Core reviewer of Kolla
  - Email: duonghq@vn.fujitsu.com
  - IRC: duonghq

- Hieu LE
  - Software Engineer at Fujitsu Vietnam
  - Official Vietnam OpenStack UG organizer
  - Email: hieulq@vn.fujitsu.com
  - IRC: hieulq

Introduction

FUJITSU

- Some reviews and thought about zero-downtime upgrade for OpenStack services.
- Ideas in this presentation are just concept.
- PoC in Kolla.

Agenda

FUJITSU

1. OpenStack Upgrading overview
   - OpenStack upgrade assertion tag
   - OpenStack rolling upgrade requirements
2. From minimal to zero
3. Zero downtime upgrade proposal in Kolla
   - Kolla support for configuration management
   - Kolla support for OSM
   - Proposal/Demo

OpenStack Upgrading

FUJITSU

- One of the most demand feature for every system
- Cold upgrade is easy
- Service-level agreement is more strictly nowadays, we need decreasing downtime of the system.

Source: http://imgur.com/a/kZCrS

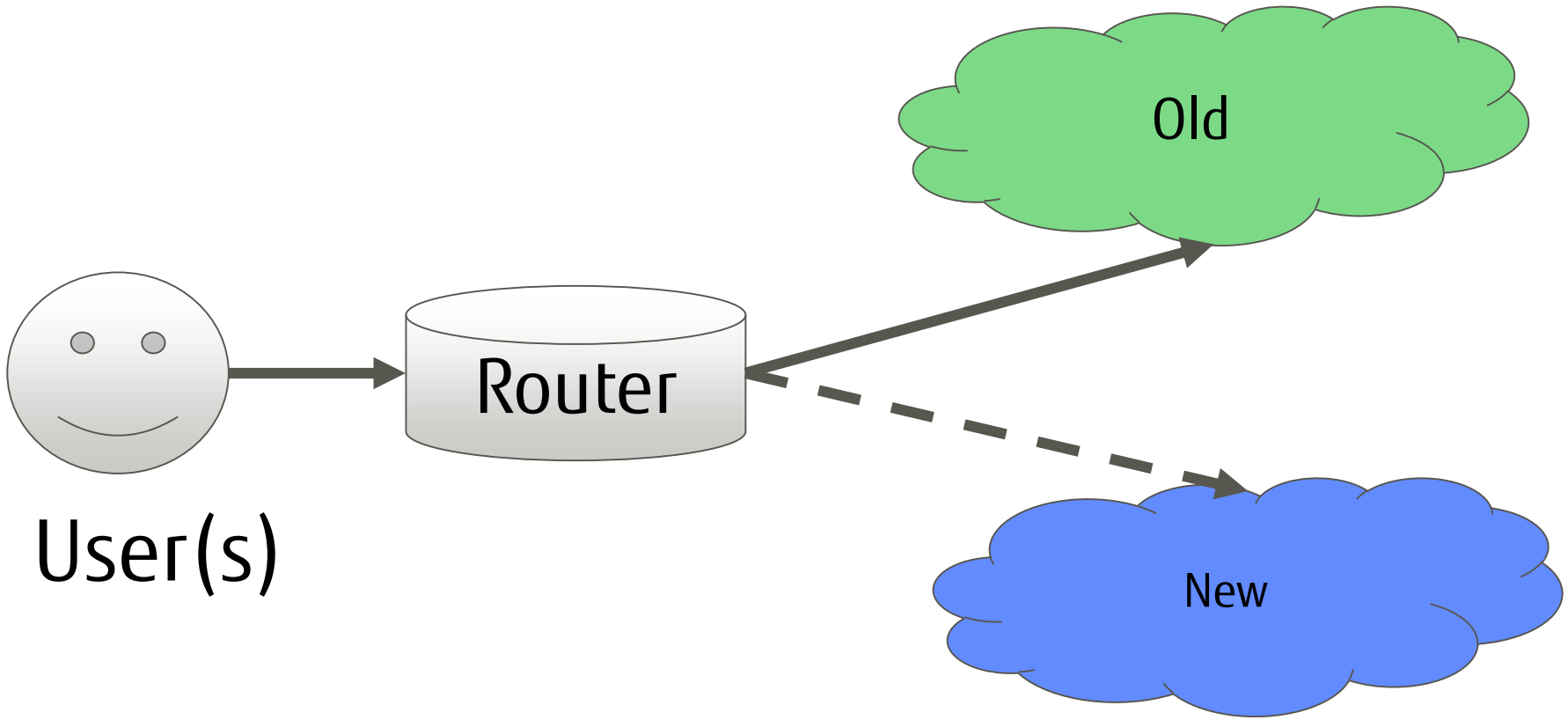
Review of current upgrading model

FUJITSU

- BlueGreen deployment and Canary release
- Rolling upgrade

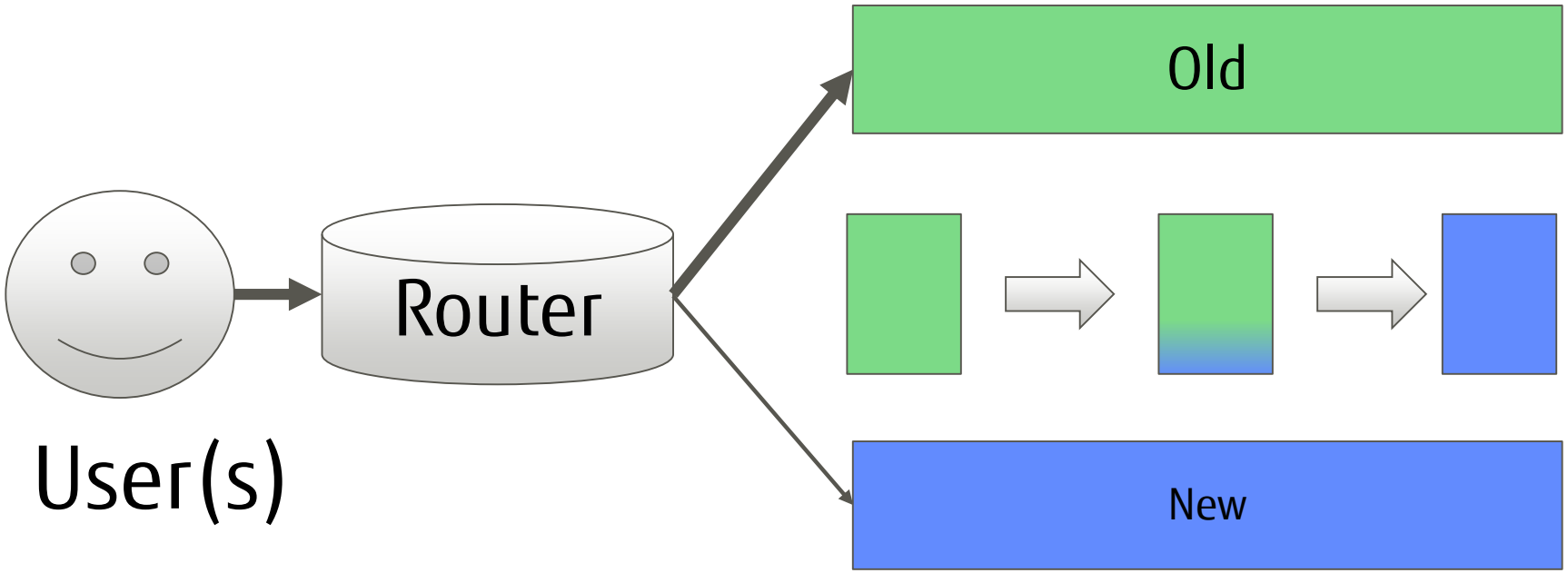
BlueGreen deployment

FUJITSU



https://martinfowler.com/bliki/BlueGreenDeployment.html

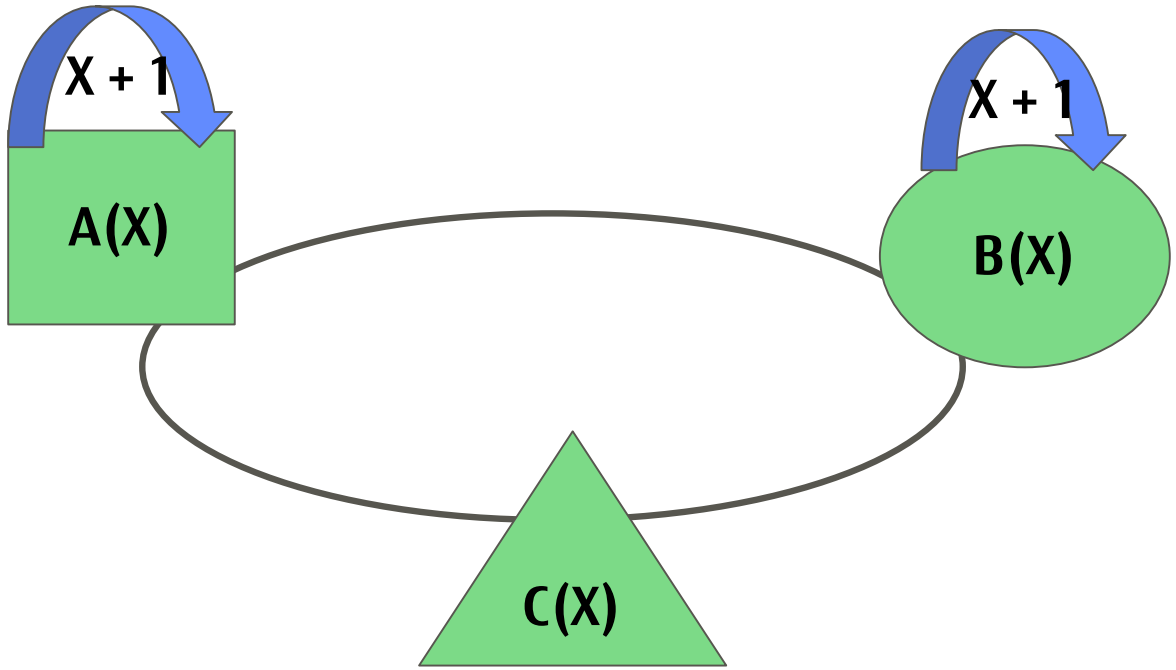
Canary release



https://martinfowler.com/bliki/CanaryRelease.html

Rolling upgrade

- Eliminates the need to restart all services on new code simultaneously.
- Requires mixed-version services work together properly in mid-upgrade.
- May have downtime of some services at a time.



OpenStack rolling upgrade requirements

FUJITSU

1. Online Schema Migration (OSM)
2. Maintenance Mode
3. Live Migration
4. Multi-version Interoperability
5. Graceful Shutdown
6. Upgrade Orchestration
7. Upgrade Gating
8. Project Tagging

https://specs.openstack.org/openstack/openstack-user-stories/user-stories/proposed/rolling-upgrades.html

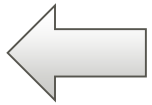
https://github.com/openstack/governance/blob/master/reference/projects.yaml

OpenStack upgrade assertion tag

FUJITSU

TC of OpenStack defines five upgrade-related tags:

1. assert:supports-upgrade
2. assert:supports-accessible-upgrade
3. assert:supports-rolling-upgrade
4. assert:supports-zero-downtime-upgrade
5. assert:supports-zero-impact-upgrade

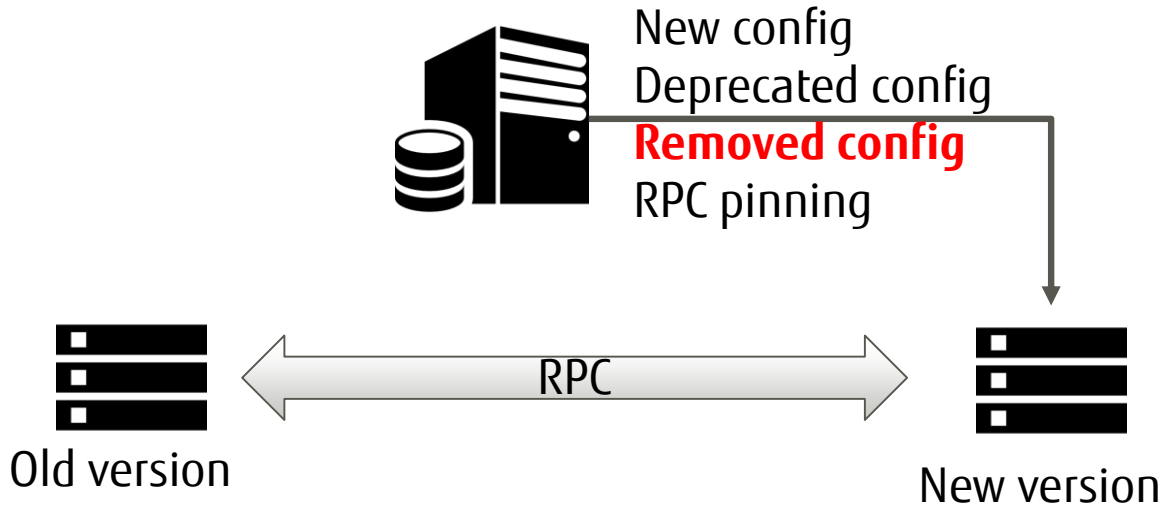


https://governance.openstack.org/tc/reference/tags/

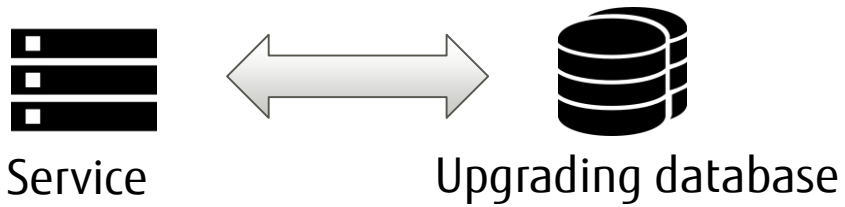
FUJITSU

Zero-downtime upgrade requires to enhance:

- Configuration management (CM)



- Database migration (OSM)



Two main approaches for OSM

FUJITSU

- Trigger-based
  - E.g. Keystone, Glance
  - Other: Facebook [1]
- Triggerless
  - E.g. Neutron
  - Binary log-based [2]

[1] https://www.facebook.com/notes/mysql-at-facebook/online-schema-change-for-mysql/430801045932/

[2] https://github.com/github/gh-ost

Zero downtime upgrade proposal

FUJITSU

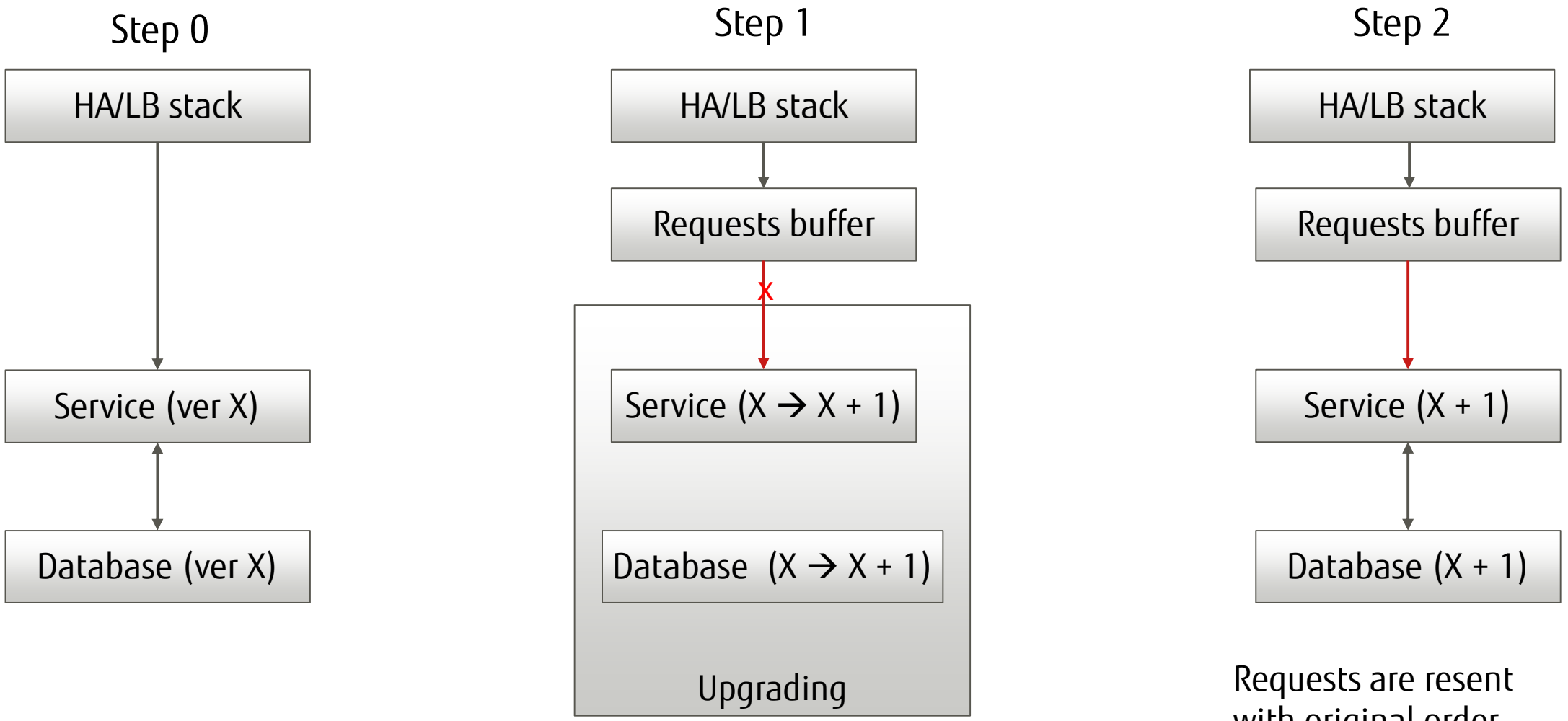
Database online schema migration: 2 candidate solutions

1. Buffer requests in upgrade period.
2. Utilize checkpoint/snapshot and binary log of database.

Zero downtime upgrade proposal (1)

FUJITSU

Buffering HTTP and RPC requests in upgrade period.



Zero downtime upgrade proposal (1)

FUJITSU

- Buffering requests in upgrade period
- There are 2 request types need buffering:
  - RESTful HTTP requests from user and inter-projects.
  - Internal service RPC requests (through MQ)
- Requests must be put in buffer in received order for replay correctly (best with timestamp)

Zero downtime upgrade proposal (1)

FUJITSU

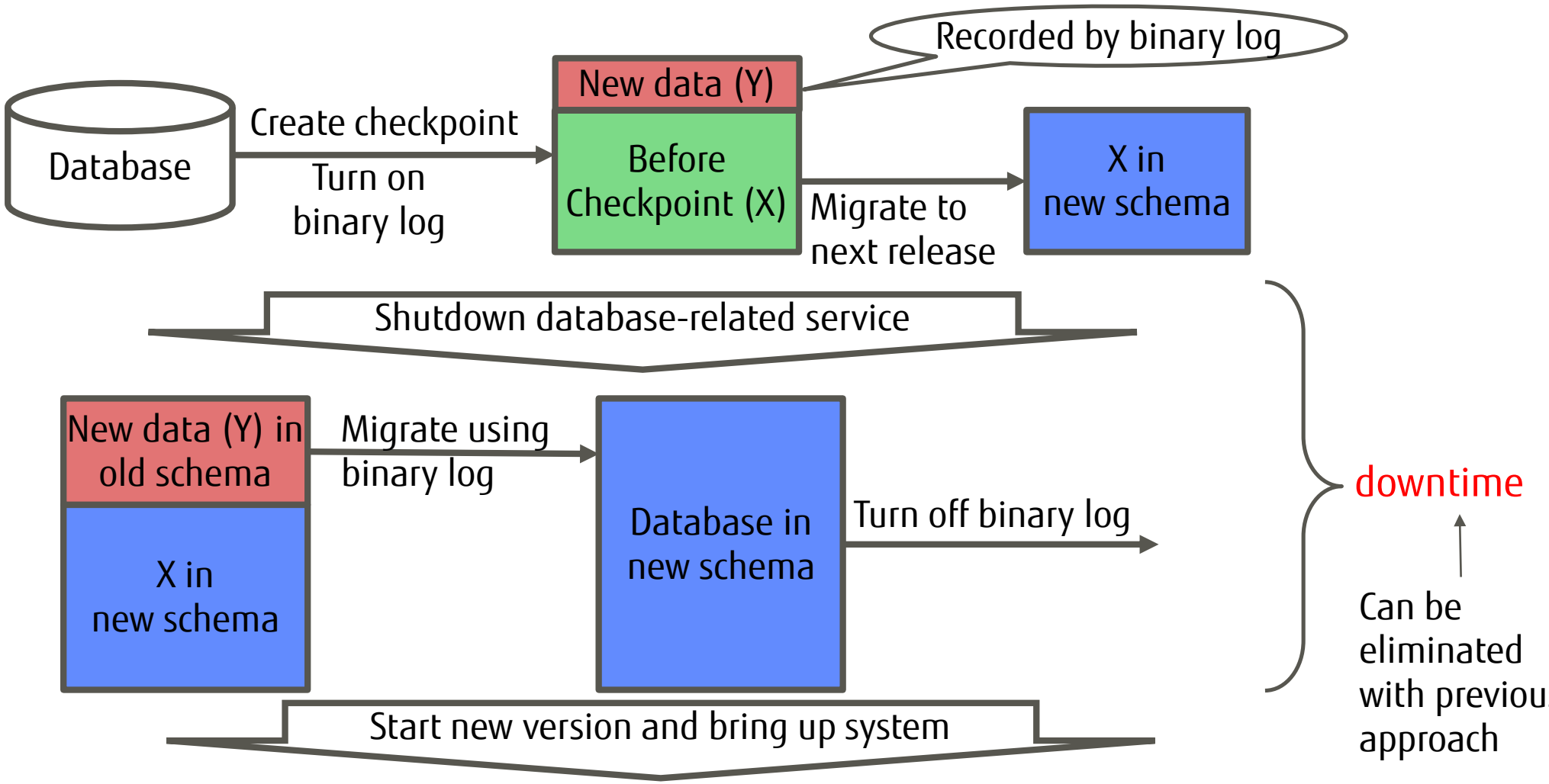
- Buffering requests in upgrade period
- Pros:
  - From user's POV: no service perceivable downtime if migration time is short enough.
- Cons:
  - From user's POV: system is lag when requests are queued in buffer.
  - If migration time is long (mainly in database migration), some requests can timeout [1].
  - Buffer can be very large if database migration time is long. ⇨ Next idea

[1] https://blueprints.launchpad.net/keystone/+spec/allow-expired

Zero downtime upgrade proposal (2)

FUJITSU

Utilize checkpoint/snapshot and binary log of database.



Zero downtime upgrade proposal (2)

FUJITSU

- Utilize checkpoint/snapshot and binary log of database.
- Pros:
  - Internal system downtime is much smaller than previous approach (only downtime for delta change vs whole database).
- Cons:
  - From user's POV: there is a short downtime.
  - Implementation is more complicated than previous approach.

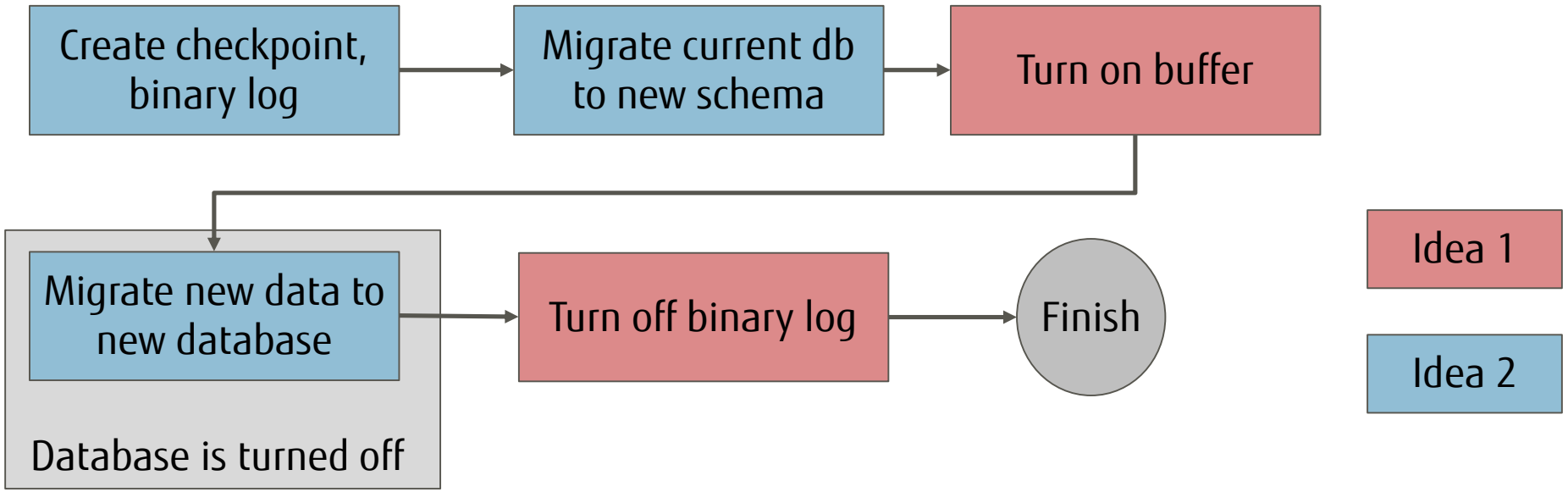
Zero downtime upgrade proposal

FUJITSU

Two candidate methods can be combined to get advantage from both methods:

- Use 2nd approach but add buffer layer and turn on when database is shutdown

→ Zero downtime from user's POV, only a bit lag.



PoC in Kolla

FUJITSU

- Kolla's mission is to provide production-ready containers and deployment tools for operating OpenStack clouds.
- Three official deliverables:
  - kolla(-image)
    -> Docker images
  - kolla-ansible
    -> Deploy OpenStack using Ansible
  - kolla-kubernetes
    -> Deploy OpenStack inside k8s cluster

Kolla support for configuration management

FUJITSU

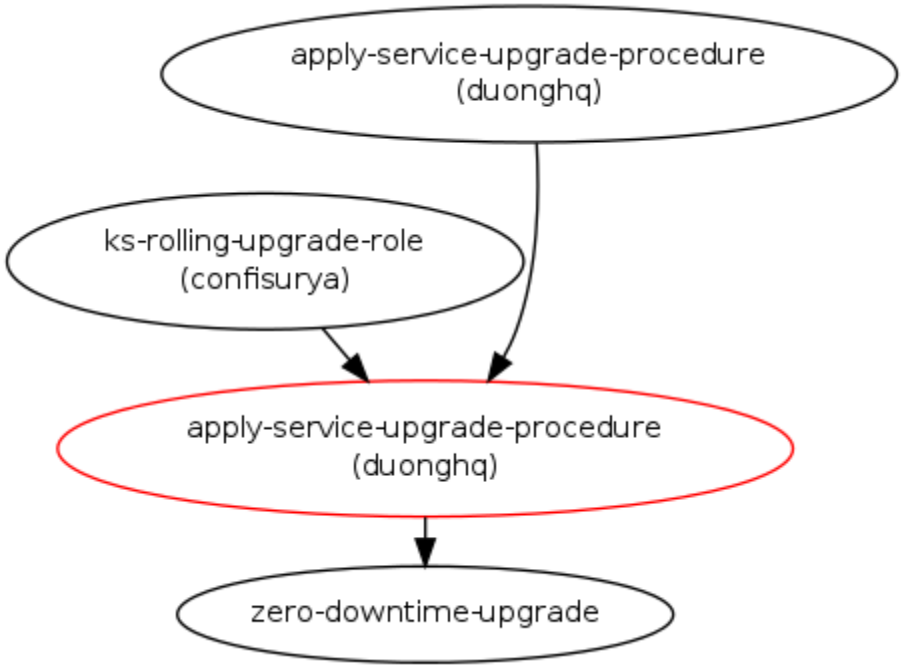
- Kolla-Ansible has implemented mechanism for configuration management functions: Configurations overridden [1]

- Kolla-Kubernetes posed good potential to automated CM

[1] http://docs.openstack.org/developer/kolla/advanced-configuration.html

Kolla support for OSM (1)

- For OpenStack projects had OSM native-supported
  - Patches for Neutron and Keystone OSM are in progress



https://blueprints.launchpad.net/kolla-ansible/+spec/apply-service-upgrade-procedure

Kolla support for OSM (2)

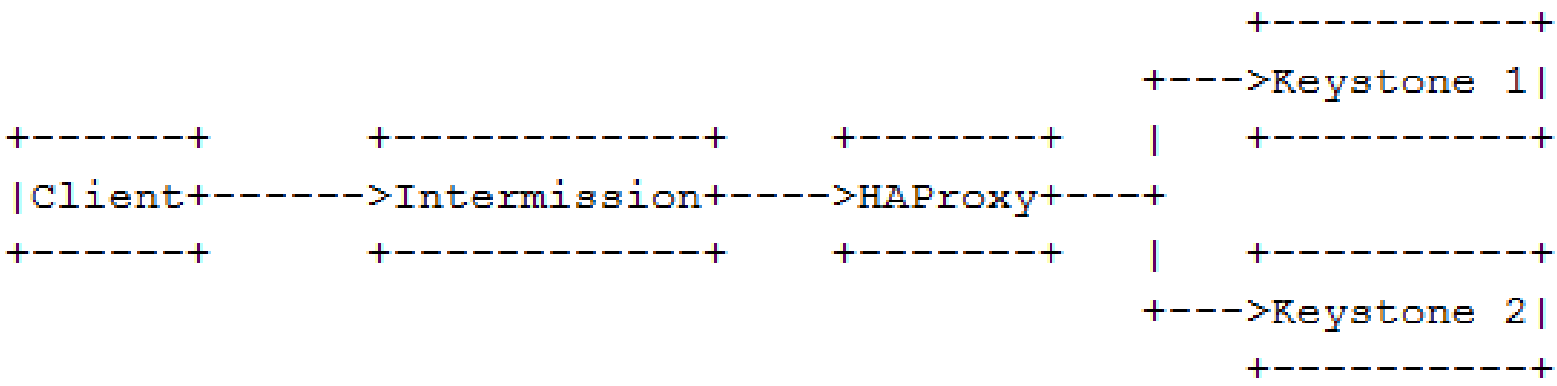
FUJITSU

- For OSM unsupported project
  - Implement above ideas at HA/LB layer.
  - Request buffer: Intermission/OpenResty

PoC scenario for 1st approach

FUJITSU

- PoC for HTTP requests buffering

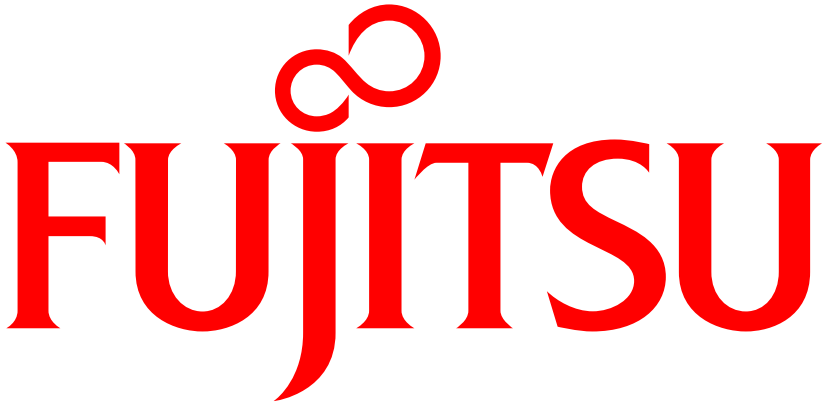


- Intermission/OpenResty configuration
  - proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
  - Intermission is bound to VIP:5000 and VIP:35357, HAProxy 5000 -> 5050, 35357 -> 35387
- Scenario:
  - Continuously send 200 create and delete network request to Ocata cluster
  - Upgrade Neutron to master code-based while requests are sending.

Demonstration

FUJITSU

- Used scripts:
  - https://github.com/vietstacker/zero-downtime-upgrade-scenario
- Rolling upgrade with Kolla, we have downtime here
  - https://www.youtube.com/watch?v=CfCBLeV1kIM
- PoC buffer request with Kolla
  - https://www.youtube.com/watch?v=6UDQXDINw84



shaping tomorrow with you