



# Deep Dive into the Upgrade of SUSE OpenStack Cloud

Jiří Suchomel  
SUSE Cloud Engineer  
[jiri.suchomel@suse.com](mailto:jiri.suchomel@suse.com)

# Overview

**Upgrade introduction**

**Non-disruptive, what does it mean?**

**Upgrading SUSE OpenStack Cloud**

**Migrating the database**

**Closer look at interesting cases**

# Why upgrade?

## Get new features and bug fixes

### OpenStack release cycle

- New release twice a year
- Distributions usually do not offer every release
- Upgrade needs to skip releases, which was not officially supported

## How to upgrade and not limit users of your cloud

- Not to limit *too much*
- Non-disruptive (whatever that really means)

# Non-disruptive Upgrade

## What is the requirement, exactly?

- What is really beneficial to users

What is technically possible with the platform

## Existing workloads are functional during the whole upgrade

- They are running and accessible
- User cannot create new instances or stop existing ones
- OpenStack services can be down for part of the upgrade
  - But try to minimize the downtime
- Don't force users to do non-disruptive if they don't require it

# SUSE OpenStack Cloud 6, 7 & 8

## Deployed on regular or virtual hardware

- No containers: no help from fancy container management tools

## Crowbar: deployment and configuration tool

- Ruby on Rails & Chef
- “Admin Server” runs crowbar and contains all Chef cookbooks

## SUSE Linux Enterprise Server (SLES)

- Every node, including Admin Server
- New Service Pack: new kernel, lots of new packages, reboot required

# SUSE OpenStack Cloud Versions

## SUSE OpenStack Cloud 6

- OpenStack Liberty
- SLES 12 SP1

## SUSE OpenStack Cloud 7

- OpenStack Newton
- SLES 12 SP2

## SUSE OpenStack Cloud 8

- OpenStack Pike
- SLES 12 SP3
- Crowbar version

# Upgrading: Basic Concepts

## Highly Available setup

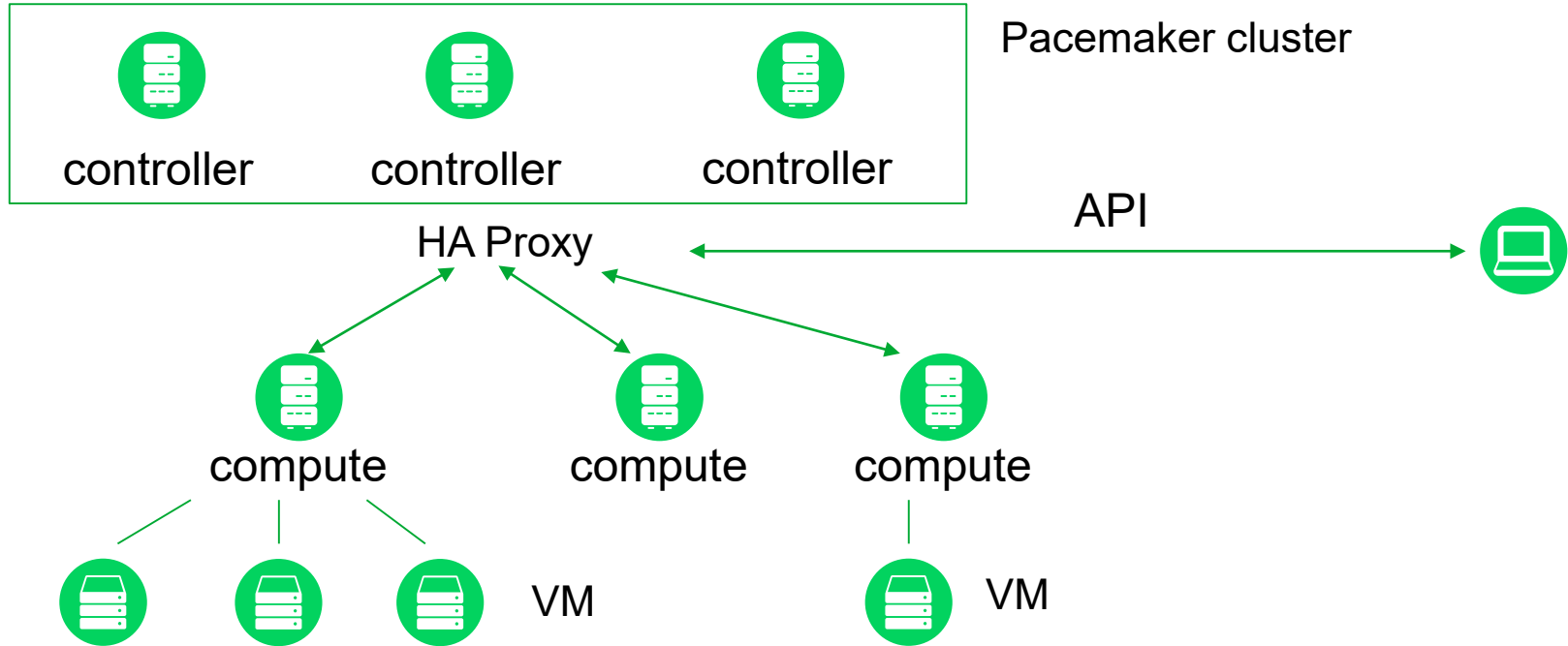
- Pacemaker clusters for services in control plane
- While a control node is being upgraded, API is provided by rest of cluster
- Still some downtime while switching to new versions

## nova live-migrate

- Live migrate all running instances from a compute node
- Disable compute service
- Upgrade packages, reboot, reconfigure and start services
- Continue with next node

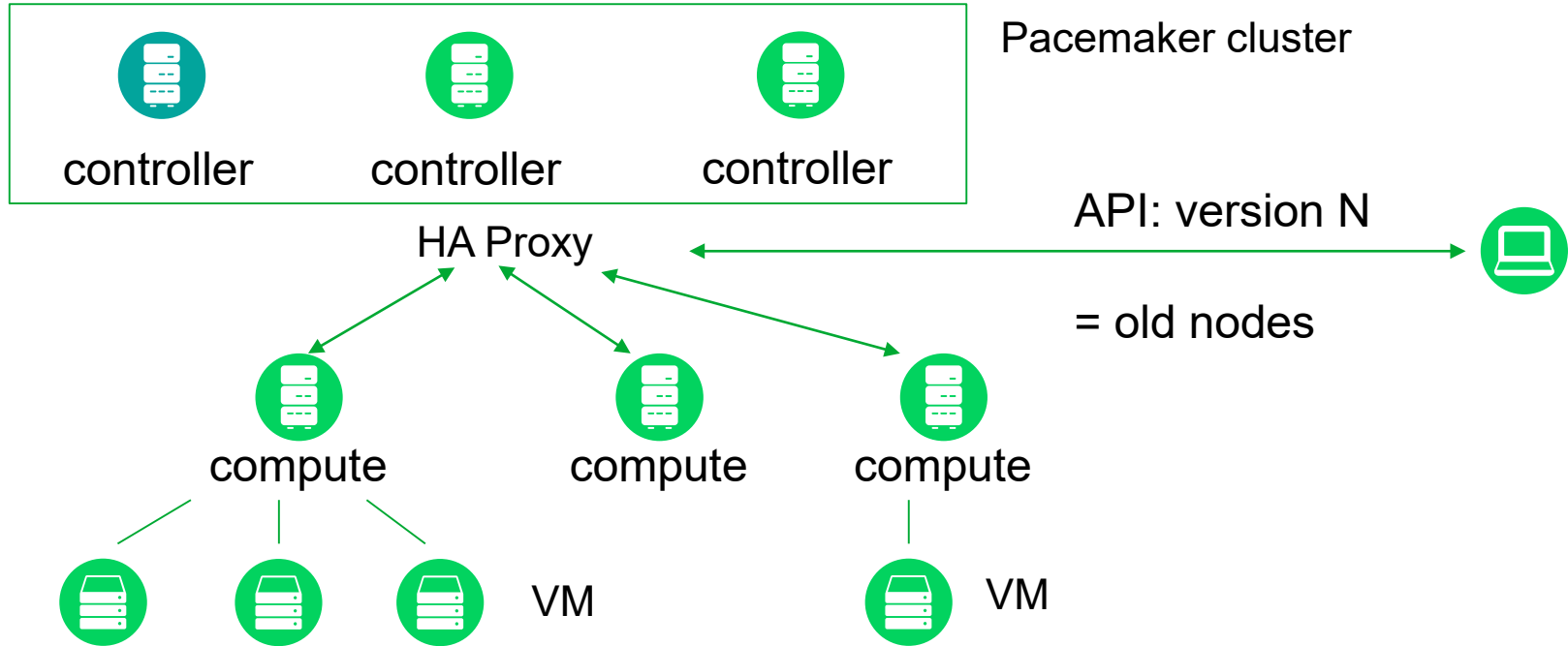
**Sounds simple, right?**

# Upgrading SUSE OpenStack Cloud

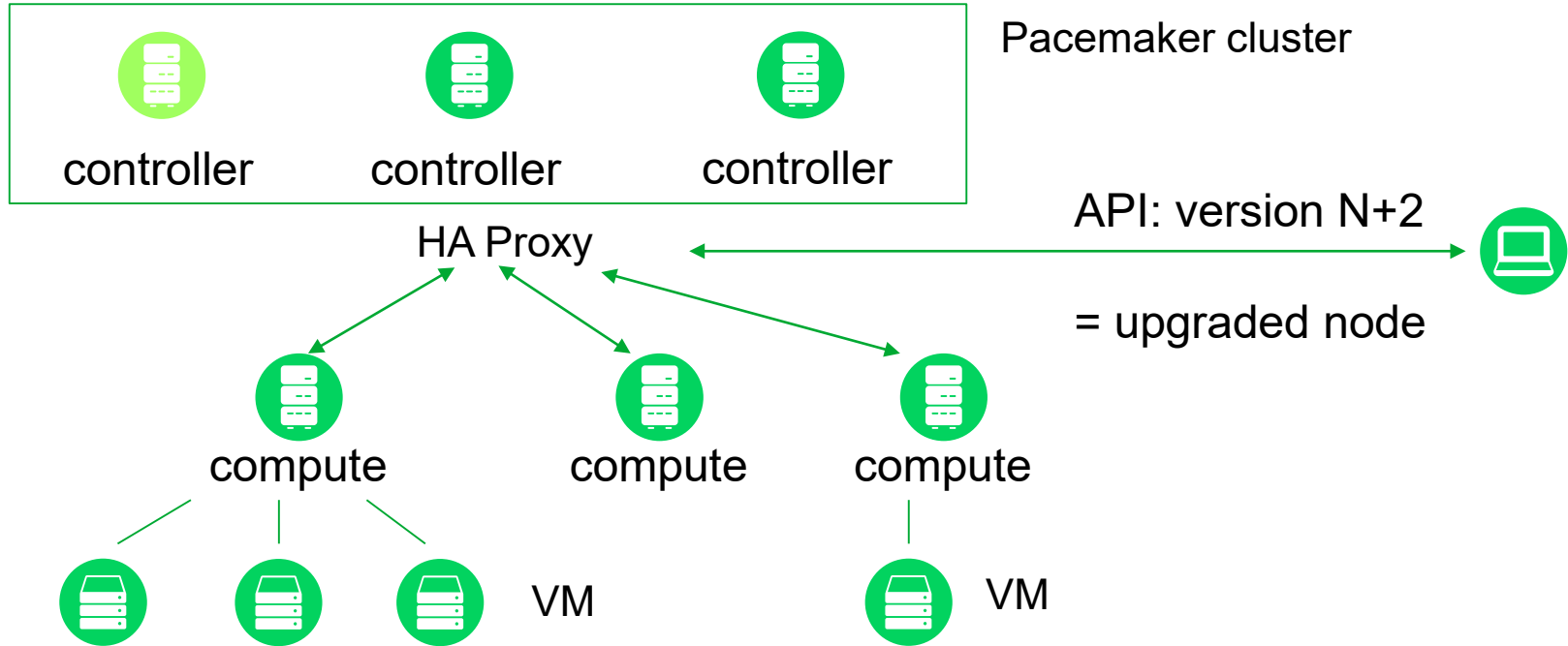




# First Controller Node Being Upgraded

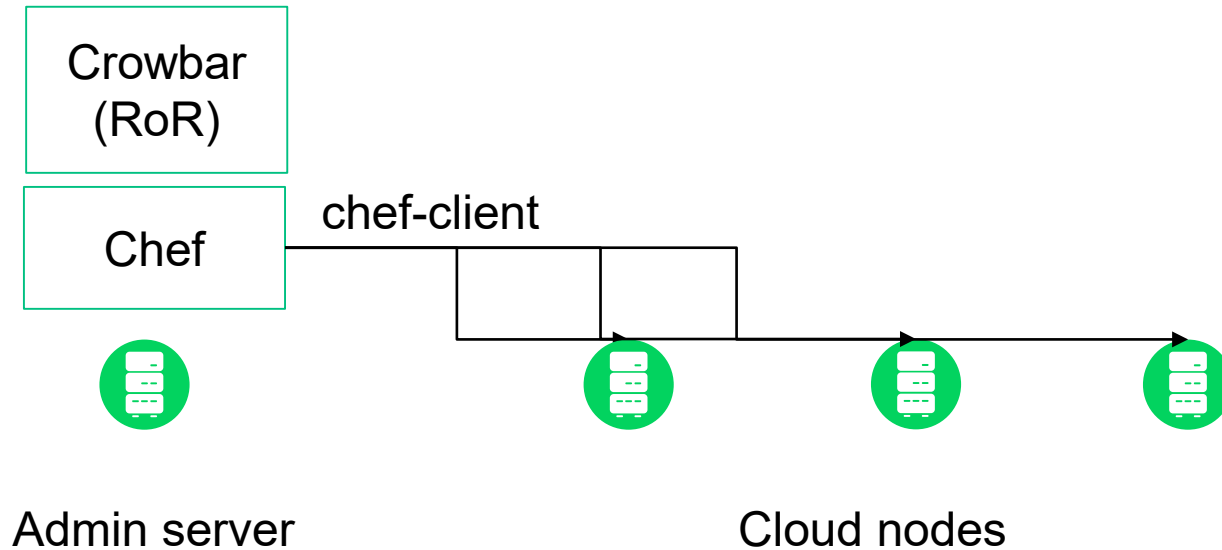


# First Controller Node Fully Upgraded

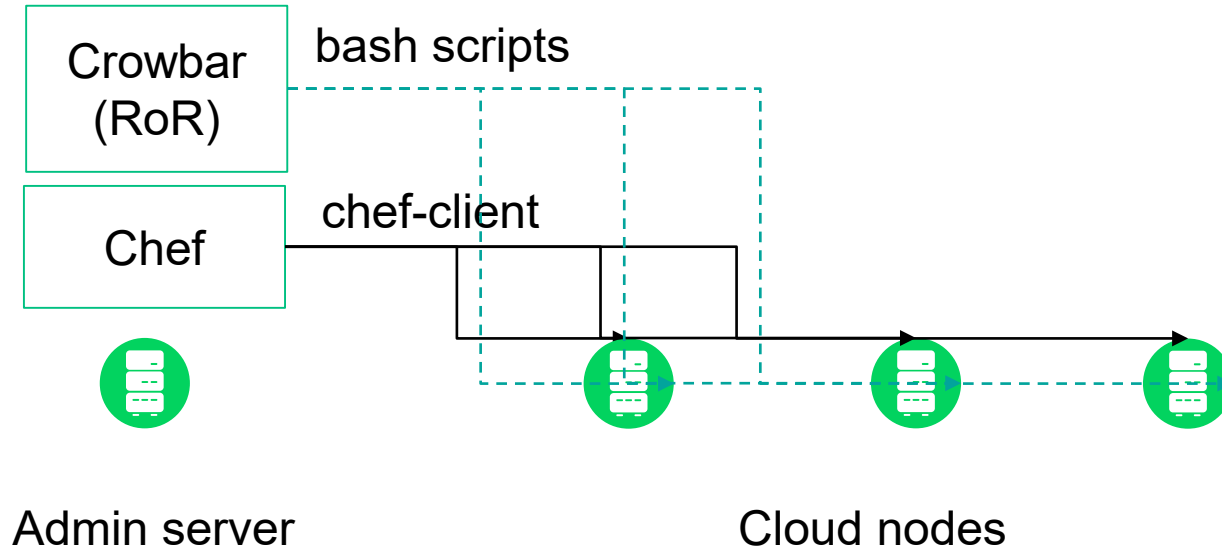


**Looking More Closely...**

# Upgrade Architecture



# Upgrade Architecture



# Upgrading in Detail: nova

**Control plane is fully upgraded to new OpenStack release**

**Compute nodes are running old versions of OpenStack**

- Version N compared to N+2 version on control plane

**What does it mean for the instances?**

- Still running, still reachable
- For live-migrations, nova-api and nova-compute must be able to communicate
- Working fine[\*] with Newton-Pike combination, not so well for Liberty-Newton
- When upgrading SOC6, we upgrade all nova services at all compute nodes

# Upgrading in Detail: Does nova really work?

## [upgrade\_levels]

- All nova services must use the same RPC API
  - `compute=auto`
- Not all operations are affected (e.g., volume attaching is)
- After all nova services are upgraded to the latest version, tell services to start using the latest RPC
  - `systemctl kill -s HUP`

# Upgrading in Detail: nova-placement

## **Nova-placement setup required since Ocata**

- Simple upgrading from Newton to Pike would be broken

## **Somehow included in Newton**

## **Some patches needed to be ported back to Newton**

## **Maintenance update for SUSE OpenStack Cloud 7**

- Updates nova set up in Newton version to correctly enable the service
- Requirement before starting the upgrade



# Upgrading in Details: Network Connectivity

**Instances are running, but are they accessible from the network?**

**Don't shut down the control node providing network connectivity to nodes**

**Evacuation of routers and load balances before the node upgrade**

- And back when upgrading the next network node

**Some services need to run**

- Neutron, keystone, rabbitmq

**neutron-ha-tool**

# Upgrading in Detail: Database Schema

## db sync commands

- Migrate the database schema and update the existing database entries
- Executed after the package upgrade
- Only one version of the service should run in the cluster

## Cinder online migration introduced in Ocata

- Intended for data migration (as opposed to schema changes)
- Breaks our idea of simple db\_sync after skipping release
- Online migration required to be run in Ocata
- db\_sync makes sure online migration was executed

## Should we install Ocata packages temporarily?

- Official OpenStack position

# Upgrading in Detail: Cinder db Migration

## Ocata's online migration moved to Pike version of a package

- `cinder-manage db sync 96`
- `cinder-manage db online_data_migrations`
- `cinder-manage db sync`

## Same with nova when upgrading from Liberty to Newton

- However, nova online migrations on Pike (SUSE OpenStack 8) must be run at the end

# Upgrading in Detail: Postponed Upgrade

Introduced in SUSE OpenStack Cloud 8

**API fully running once control plane is upgraded**

**Nova services running Pike and Newton can communicate**

- Usable cloud, although not fully upgraded
- Beware of collocation of services (e.g., cinder-volume on compute node)
- Nova is using Newton RPC versions

**User can even use crowbar to modify the configuration**

**Upgrade of compute nodes when the right time comes**

# Upgrading in Detail: Supported Scenarios

## **Prechecks: make sure the system is healthy and supported**

- Pacemaker/Ceph cluster health
- Latest updates
- Services collocation
- Database backend

## **Do not rely on Admin server!**

- Cinder storage
- STONITH

# Database Upgrade and Migration

# OpenStack Database Engine

## SUSE OpenStack Cloud 7: PostgreSQL

## SUSE OpenStack Cloud 8: Mariadb

- Mariadb is where upstream is going
- Necessary to provide the upgrade path without data loss

# Deployment and Orchestration Challenges

**We need to change the DB engine from PostgreSQL to MariaDB**

**When to migrate the data**

- Before the upgrade to Cloud 8? After the upgrade? Coupled with the upgrade?

**Where to deploy the MariaDB instance**

- Separate node/cluster vs. the same node/cluster as PostgreSQL
- PostgreSQL can be deployed to a 2-node cluster
- MariaDB's galera cluster needs at least 3 nodes

**How to deploy MariaDB alongside PostgreSQL**

- Allow users to deploy MariaDB that is configured but not used by OpenStack



# How to Migrate: High-Level View

**Deploy MariaDB into existing SOC7 setup**

**Use the existing OpenStack tools to create schema in the target databases (e.g. `keystone-manage db_sync`)**

**Stop OpenStack services so they do not write to PostgreSQL**

**Migrate the data**

**Reconfigure services to use MariaDB**

# Migrating SQL Data

## Issues

- Some OpenStack components have different a schema for different backends
- UTF in MySQL does not support full character set
- Size limit of MySQL's TEXT

## Considered existing tools

## Ultimately we had to write our own

- Psql2mysql
- Precheck: check the source DB
- Migrate: needs target DB
- Work on set of databases

# Tools

# User Interface

## Web UI executed from crowbar dashboard

### CLI

- `crowbarctl upgrade status`
- `crowbarctl upgrade prechecks`
- `crowbarctl upgrade nodes`

### Log and status files to watch

- `/var/log/crowbar/admin-server-upgrade.log`
- `/var/lib/crowbar/upgrade/7-to-8-progress.yml`
- `/var/log/crowbar/node-upgrade.log`

# Links

## SUSE OpenStack Cloud

- <https://www.suse.com/products/suse-openstack-cloud>

## Crowbar and friends

- <https://crowbar.github.io>
- <https://github.com/crowbar>
- <https://github.com/SUSE/psql2mysql>

Questions?



We adapt. You succeed.

## **Unpublished Work of SUSE LLC. All Rights Reserved.**

This work is an unpublished work and contains confidential, proprietary and trade secret information of SUSE LLC. Access to this work is restricted to SUSE employees who have a need to know to perform tasks within the scope of their assignments. No part of this work may be practiced, performed, copied, distributed, revised, modified, translated, abridged, condensed, expanded, collected, or adapted without the prior written consent of SUSE. Any use or exploitation of this work without authorization could subject the perpetrator to criminal and civil liability.

## **General Disclaimer**

This document is not to be construed as a promise by any participating company to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. SUSE makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for SUSE products remains at the sole discretion of SUSE. Further, SUSE reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All SUSE marks referenced in this presentation are trademarks or registered trademarks of SUSE LLC. in the United States and other countries. All third-party trademarks are the property of their respective owners.