OpenStackクラスタ間マイグレーション

Havana to Mitaka

常松伸哉 / GMO Pepabo, Inc.

2017.07.20 OpenStack Days Tokyo 2017

常松伸哉　@tnmt

プリンシパルエンジニア

技術部 技術基盤チーム

https://blog.tnmt.info

GMOペパボ

企業理念: もっとおもしろくできる

ミッション: インターネットで可能性をつなげる、ひろげる

ホスティング

ホームページを作成する時に必要なサーバーや、あると便利なサービスを取り揃えています。

EC支援

大規模なネットショップから個人間の取引まで、“インターネットで何かを売りたい買いたい”方をサポートします。

ハンドメイド

個性あふれるハンドメイド作品と出会えるCtoCオンラインマーケットを中心にサービス提供しています。

コミュニティ

ネットでの情報発信を手軽に楽しめるサービスで、みなさんにコミュニケーションの場を提供しています。

LOLIPOP! マネージドクラウド リリース αテスト中！

アジェンダ

- プライベートクラウド導入経緯と今まで
- 2バージョン並行運用について
  - 事例, トラブルシュート
- 並行・自社運用を経て今後

プライベートクラウド導入経緯


NyAH

Nyah is ...

- GMOペパボのプライベートクラウドのコードネーム
- OpenStackで構築された仮想インフラ基盤
- 各サービス・商材のサーバー環境として利用中
- 2014年構想 7月よりOpenStack検討開始
- 2015年5月 グループ会社の支援を受け、Havanaスタック運用開始
- 2017年1月 自社構築にて、Mitakaスタックの運用開始

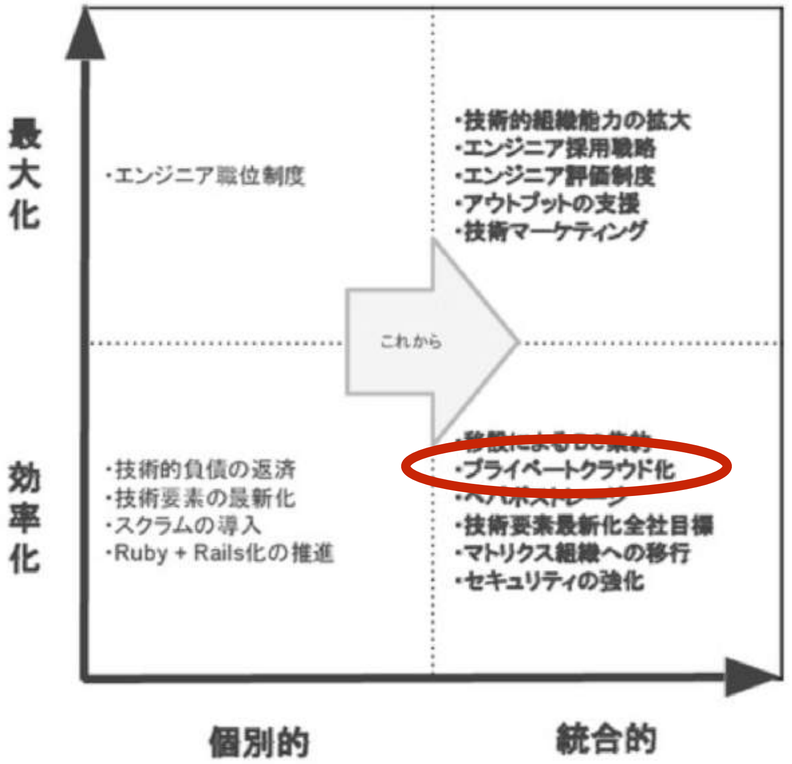
2016年5月

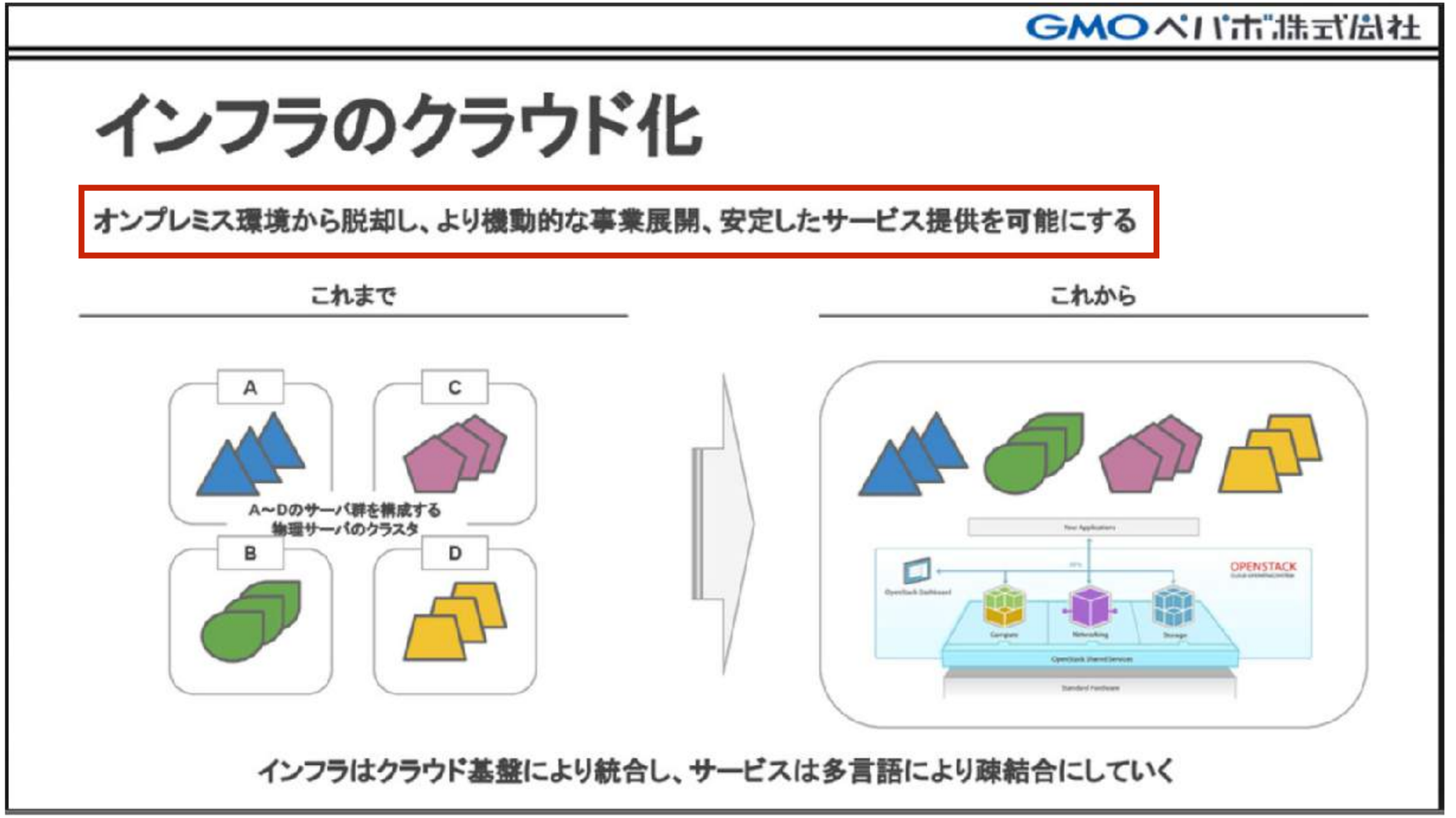
OpenStack運用1年を振り返ったまとめを発表

https://speakerdeck.com/tnmt/pepabos-privatecloud-nyah-after-that

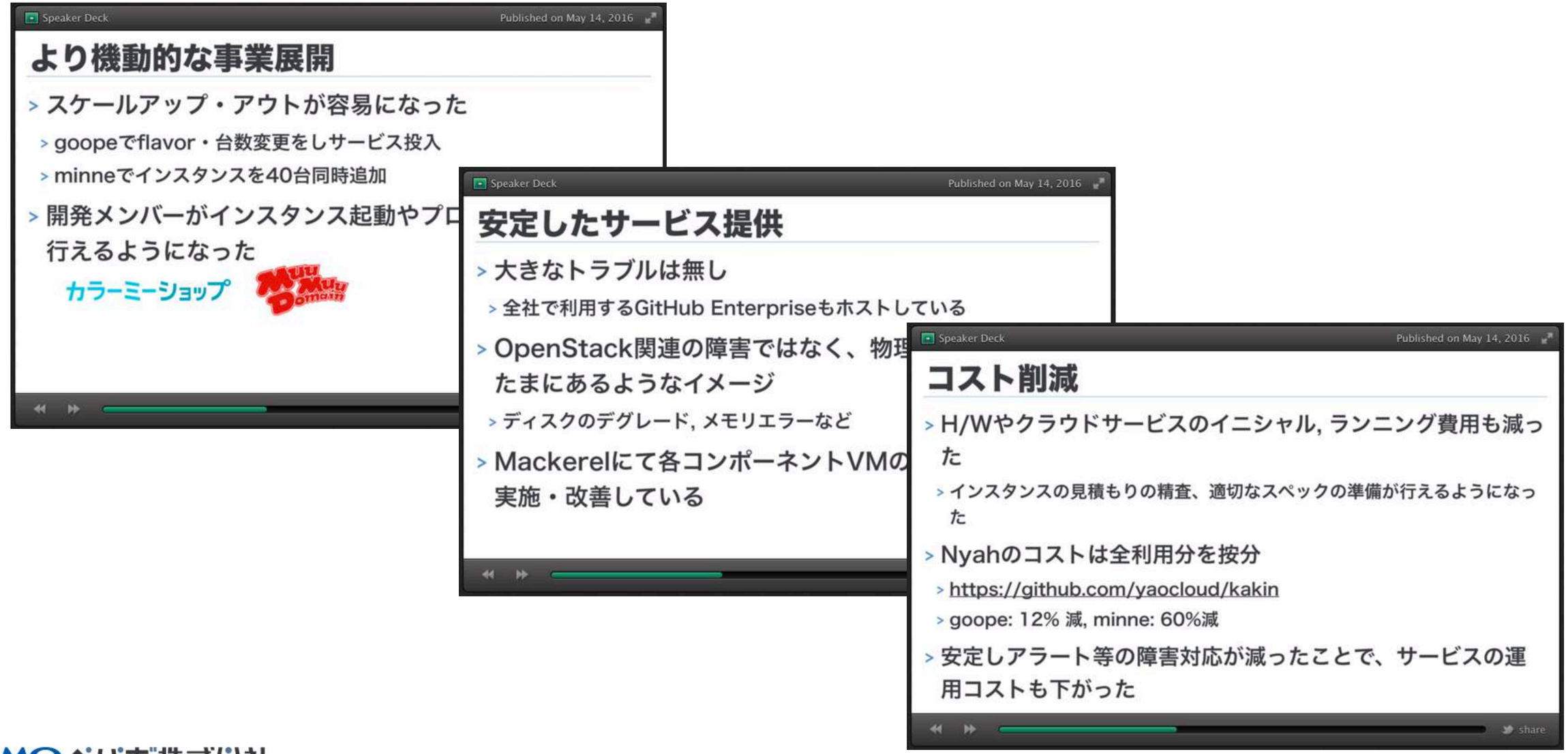
2016年5月発表の振り返り

振り返り: 導入経緯

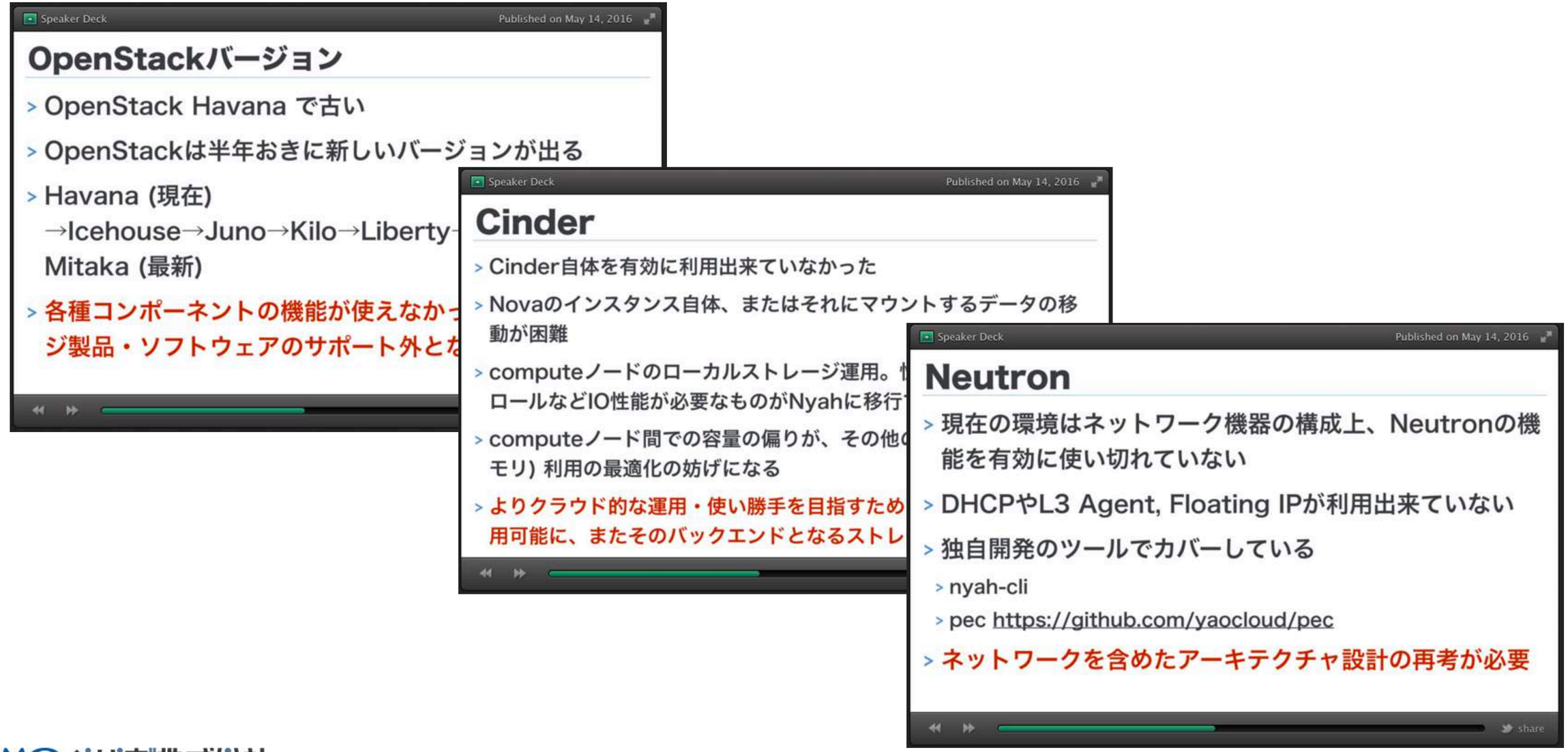

最大化
効率化
・エンジニア職位制度
・技術的組織能力の拡大
・エンジニア採用戦略
・エンジニア評価制度
・アウトプットの支援
・技術マーケティング
これから
・技術的負債の返済
・技術要素の最新化
・スクラムの導入
・Ruby + Rails化の推進
・ブライベートクラウド化
・技術要素最新化全社目標
・マトリクス組織への移行
・セキュリティの強化
個別的
統合的


GMOペパボ株式会社
インフラのクラウド化
オンプレミス環境から脱却し、より機動的な事業展開、安定したサービス提供を可能にする
これまで
これから
A
C
B
D
A〜Dのサーバ群を構成する
物理サーバのクラスタ
OPENSTACK
インフラはクラウド基盤により統合し、サービスは多言語により疎結合にしていく

振り返り: 効果


Speaker Deck
Published on May 14, 2016
より機動的な事業展開
> スケールアップ・アウトが容易になった
> goopeでflavor・台数変更をしサービス投入
> minneでインスタンスを40台同時追加
> 開発メンバーがインスタンス起動やプロ
行えるようになった
カラーミーショップ
Speaker Deck
Published on May 14, 2016
安定したサービス提供
> 大きなトラブルは無し
> 全社で利用するGitHub Enterpriseもホストしている
> OpenStack関連の障害ではなく、物理
たまにあるようなイメージ
> ディスクのデグレード, メモリエラーなど
> Mackerelにて各コンポーネントVMの
実施・改善している
Speaker Deck
Published on May 14, 2016
コスト削減
> H/Wやクラウドサービスのイニシャル, ランニング費用も減った
> インスタンスの見積もりの精査、適切なスペックの準備が行えるようになった
> Nyahのコストは全利用分を按分
> https://github.com/yaocloud/kakin
> goope: 12% 減, minne: 60%減
> 安定しアラート等の障害対応が減ったことで、サービスの運用コストも下がった
share

振り返り: 課題


Speaker Deck
Published on May 14, 2016
OpenStackバージョン
> OpenStack Havana で古い
> OpenStackは半年おきに新しいバージョンが出る
> Havana (現在)
→Icehouse→Juno→Kilo→Liberty
Mitaka (最新)
> 各種コンポーネントの機能が使えなか
ジ製品・ソフトウェアのサポート外とな
Speaker Deck
Published on May 14, 2016
Cinder
> Cinder自体を有効に利用出来ていなかった
> Novaのインスタンス自体、またはそれにマウントするデータの移動が困難
> computeノードのローカルストレージ運用。
ロールなどIO性能が必要なものがNyahに移行
> computeノード間での容量の偏りが、その他
モリ) 利用の最適化の妨げになる
> よりクラウド的な運用・使い勝手を目指すため
用可能に、またそのバックエンドとなるストレ
Speaker Deck
Published on May 14, 2016
Neutron
> 現在の環境はネットワーク機器の構成上、Neutronの機能を有効に使い切れていない
> DHCPやL3 Agent, Floating IPが利用出来ていない
> 独自開発のツールでカバーしている
> nyah-cli
> pec https://github.com/yaocloud/pec
> ネットワークを含めたアーキテクチャ設計の再考が必要
share

振り返り: 2016年方針

次世代アーキテクチャ

GMOペパボ株式会社

Bayt化・プライベートクラウド基盤整備の後、次に必要となるストレージ、データベースを検証・導入し、事業を牽引するインフラに。

■ **Nyahの自前構築**
・新規ラックの導入
・NW機器の増設
・サーバの購入
・新しいバージョンのOpenStackを使用したプライベートクラウドの構築

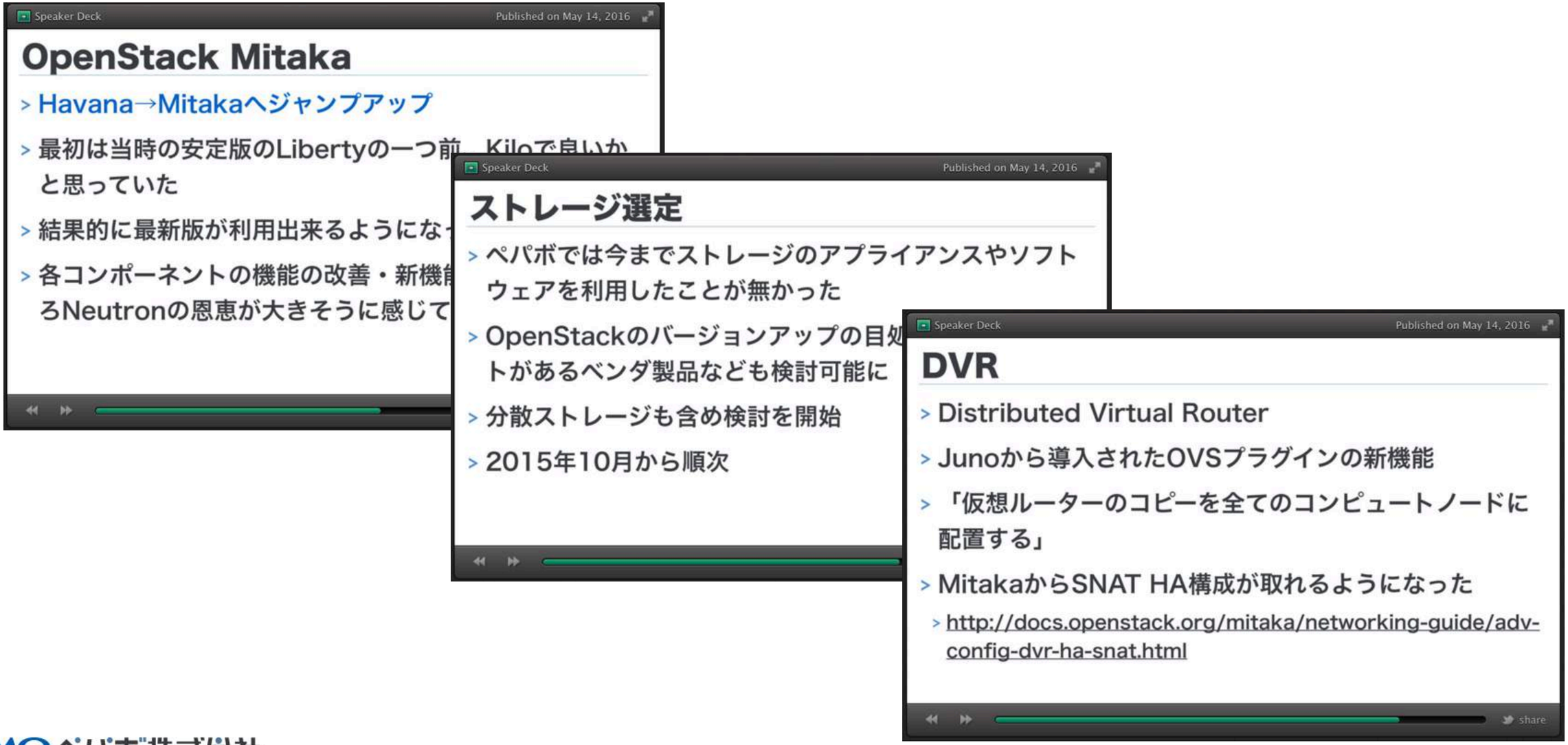
■ **高可用DB**
・サーバの購入
・高可用性を担保する技術開発

■ **ファイルシステム**
・ストレージ専用サーバの購入
・次世代ストレージの構築

(次世代ホスの話はこちら)

※ 水色背景がこれから新規に構築していくもの

振り返り: 自社でのOpenStack構築

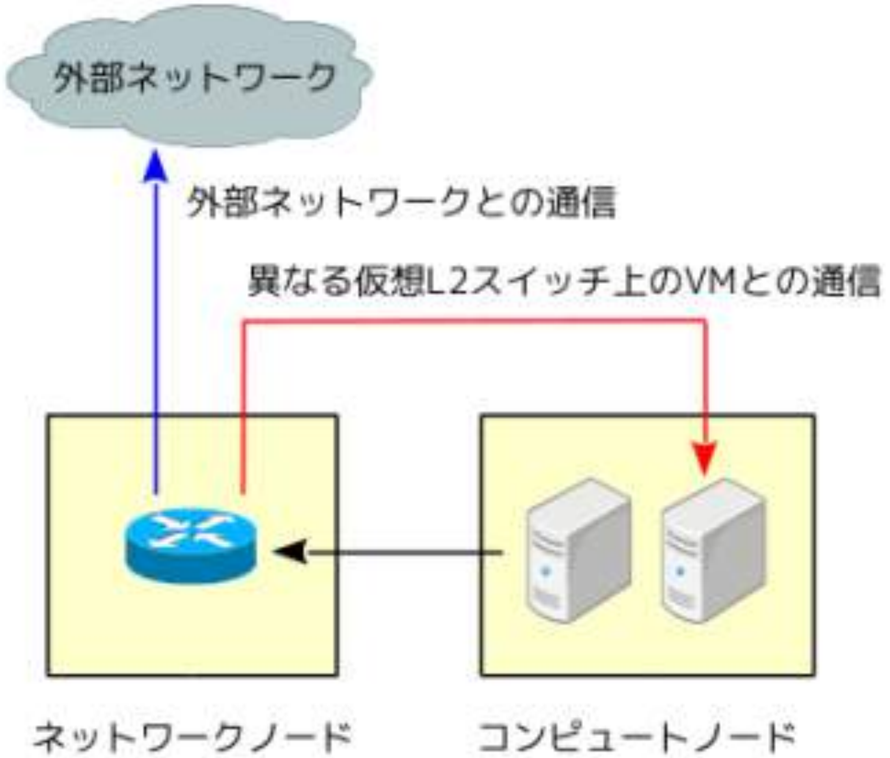


現在のNyahの状況

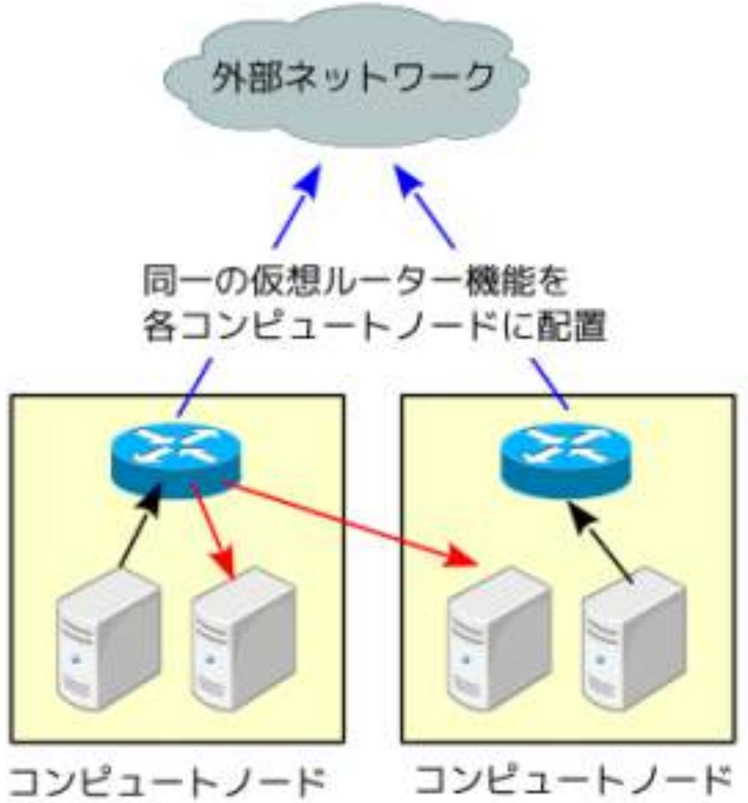
Nyah before after

- OpenStackバージョン: Havana → Mitaka
- Cinder利用: 無 → 有 (Dell EMC ScaleIO)
- Neutron: 一部利用 → DVR + SNAT HA

DVR



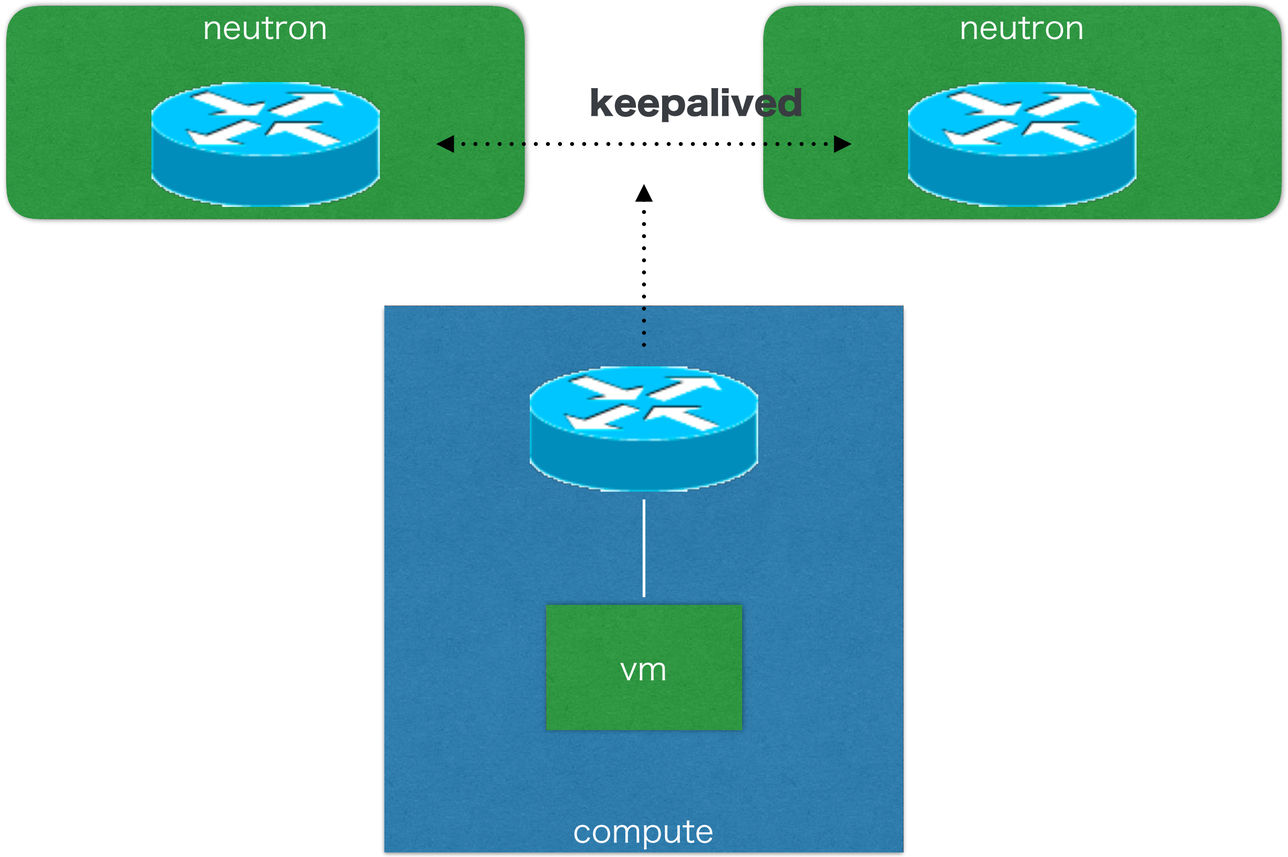
DVRを使用しない場合の通信経路



DVRを使用する場合の通信経路

https://www.school.ctc-g.co.jp/columns/nakai/nakai57.html

SNAT HA



Mitaka以降はneutronを冗長化することで、SNATルーターを分散配置し
keepalivedで冗長化することが出来るようになった

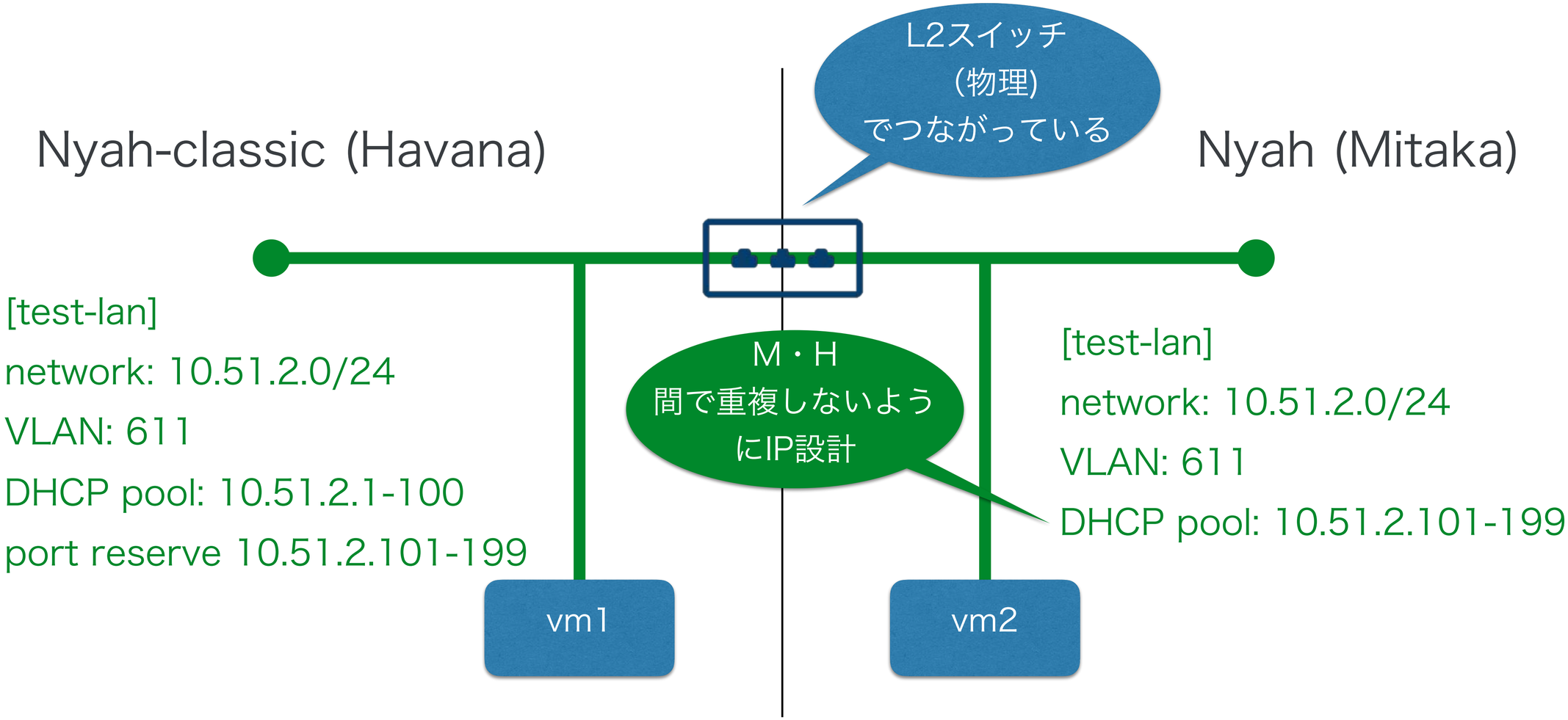
2バージョン間マイグレーション

マイグレーション？

- 既存のHavana環境をアップグレード出来ない
  - OSが異なる (CentOS → Ubuntu)
  - Havana to Mitakaは一度ではアップグレード不可
- 新規ラック・サーバにてMitaka環境を別途構築
- 同DC内別ラックなのでネットワークは疎通出来る

2環境のネットワーク接続

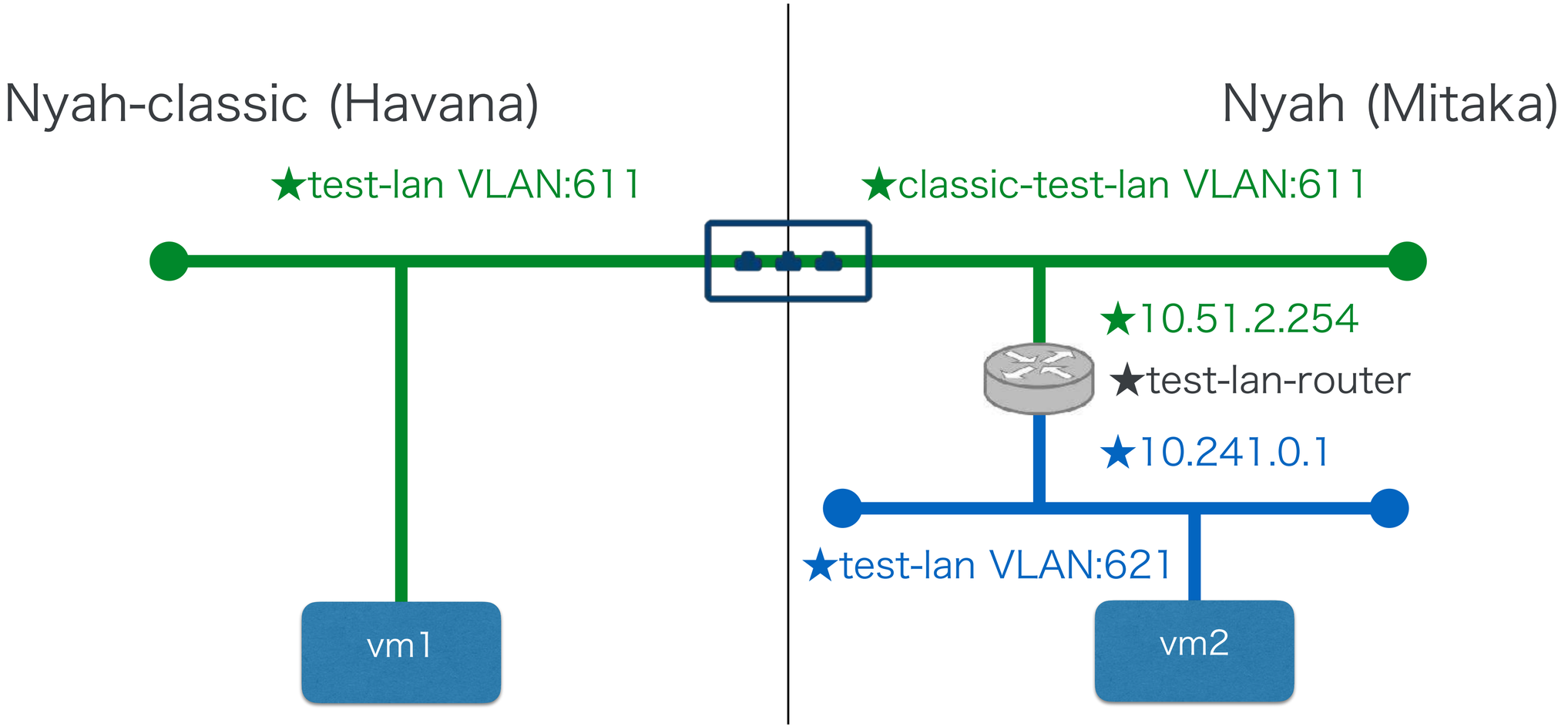
ネットワーク接続 例1


L2スイッチ
(物理)
でつながっている
Nyah-classic (Havana)
Nyah (Mitaka)
[test-lan]
network: 10.51.2.0/24
VLAN: 611
DHCP pool: 10.51.2.1-100
port reserve 10.51.2.101-199
M・H
間で重複しないよう
にIP設計
[test-lan]
network: 10.51.2.0/24
VLAN: 611
DHCP pool: 10.51.2.101-199
vm1
vm2

ネットワーク接続 例1

- ルーターを利用しない構成
- M(itaka)・H(avana)間で同じVLAN ID, サブネットを利用
- 互いの環境で管理IPが重複しないように管理
    - Havana: reserved port (DHCPレンジを後から変更できない…)
- シンプルで分かりやすい
- projectの規模が大きいとIPが足らなくなる可能性がある

ネットワーク接続 例2


Nyah-classic (Havana)
Nyah (Mitaka)
★test-lan VLAN:611
★classic-test-lan VLAN:611
★10.51.2.254
★test-lan-router
★10.241.0.1
★test-lan VLAN:621
vm1
vm2

ネットワーク接続 例2

- ルーターを利用する構成
- M・H環境での管理IP重複を気にしなくとも良い
- トポロジは若干複雑にはなるが理解が出来ないほどではない
- 利用port数の増加が予想される場合、別サブネットになるのでレンジを広く取ることが出来る
- **現在はこちらを利用する方針にしている**

問題1: パケット重複問題

1. vm2→vm1: routerから出るパケットのSRC MACはDVRが独自に付与するMAC
2. vm1→vm2: **パケットのDSTがrouterのMACアドレス**
   - DSTはDVRが独自に付与するMACにして欲しい
3. DVRで構成されたrouterは各compute node上に存在し、M・H間はL2でつながっているため、すべてのrouterにパケットが届く
4. vm2に届くパケットが重複し通信が不安定になる

問題1: パケット重複問題

- Mitakaでの旧プロジェクトネットワーク（前述 VLAN:611）は**外部(external)ネットワークとして定義する必要がある**
  - 外部ネットワークとして設定した場合パケットのDUPは発生しない
  - 内部ネットワーク -> 外部ネットワークへの通信がSNATルーターを経由する
  - つまりDVRを使用しないため

Terraform

Terraform

- Mitakaより本格利用開始
- OpenStack Provider
- instance以外にも様々なリソースに対して、宣言的に記述・管理が行えるようになった
  - port, network, volume 等々
- 2環境のネットワーク接続用のrouterも作成出来るようになった

問題2: 不正なルーターが作成される

- SNAT HA構成でrouter作成時にexternal_gateway_infoを指定するとネットワーク通信が出来ないrouterが作成されてしまう
- neutronノード上でエラーがループしログが肥大化する
- 対処方法としてはneutron-l3-agentの再起動のみ
- Horizonからのrouter作成時には問題にならなかった
- Terraformを利用するようになってから問題が発覚
- TFがextra_gatewayを設定するタイミングが早すぎることが原因

問題2: 不正なルーターが作成される

- 対処方法としてはTFでのrouter作成時にはexternal_gatewayは指定しない
- 7月頭、backport fixで根本対応となる修正が行われた
- https://bugs.launchpad.net/neutron/+bug/1662804
- neutronノードは冗長構成が取られている為、1台ずつフェイルオーバーさせてパッケージをアップデート

問題3: SNAT VRRP port DOWN

- SNAT HAの手動フェイルオーバー時に発覚
- neutronノード間でkeepalivedによる冗長化が取られている
- VRRPを送受信するportが起動しているのが前提だが、何故か1台でそのportがダウンしていた
- portがダウンしているため、手動フェイルオーバーが行えない
- 結果インスタンスから外部ネットワークへの通信が行えなくなる

問題3: SNAT VRRP port DOWN

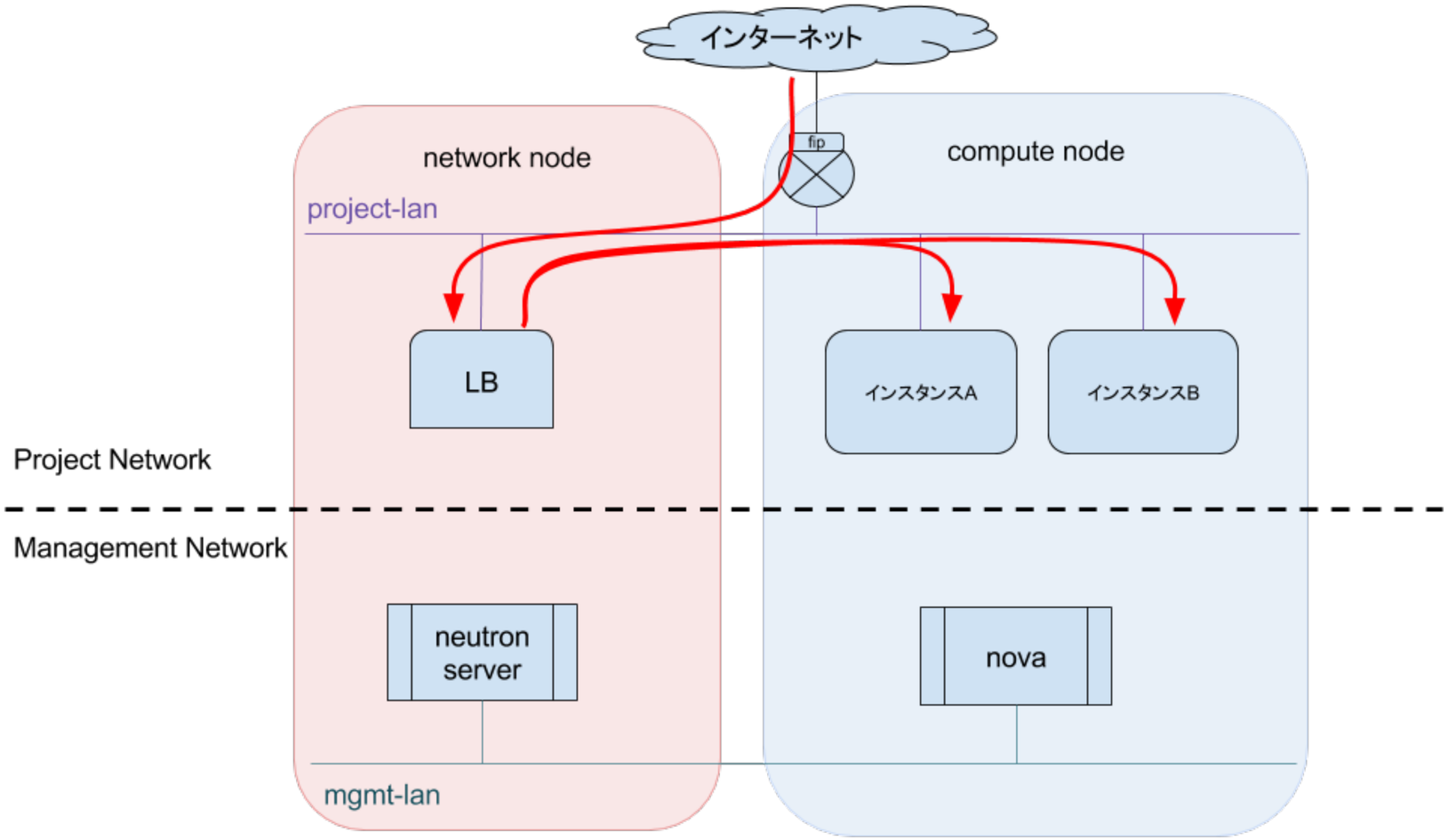
- 今のところ原因は不明
- 対応を検討中
  - portがDOWNしていることが検知出来るように監視を行う
  - 検知されたら自動的に起動する

Octavia (LBaaS)

Octavia

- LBaaS v2の実装, OpenStackコンポーネントの一つ
- Mitakaから標準でサポートされている
- 概要、挙動については弊社メンバーによるブログ記事参照
  - http://buty4649.hatenablog.com/entry/2017/06/19/141206
- Havana環境では各projectや要件ごとにロードバランサ用のインスタンスを作成していた
- これをOpenStackコンポーネントとして解決したい

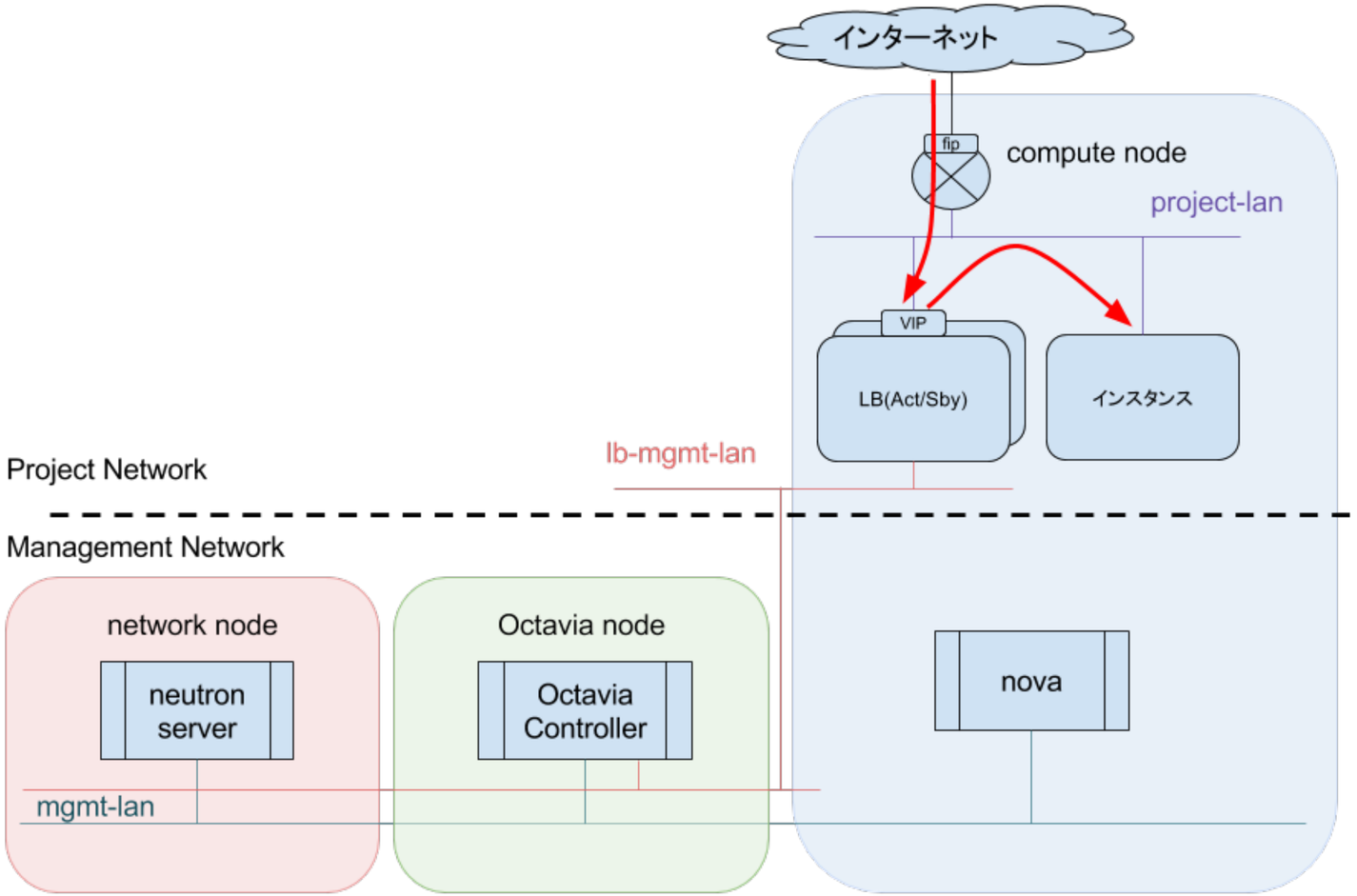
比較 (haproxy backend)



比較 (haproxy backend)

- Floating IP (fip) はLBのVIPに紐付いている
- インターネットからのパケットはLBを通しインスタンスへロードバランスされる
- **network nodeがシングルポイントになりやすい**

比較 (Octavia)


インターネット
fip
compute node
project-lan
VIP
LB(Act/Sby)
インスタンス
lb-mgmt-lan
Project Network
Management Network
network node
neutron server
Octavia node
Octavia Controller
nova
mgmt-lan

比較 (Octavia)

- **LBがhypervisor(comute node)上にある**
- LBを通常のインスタンスとして起動することでnetwork nodeへのトラフィック集中を避けれる
- network nodeのSPOFも解消出来る

問題4: Octavia v0.8.1

- MitakaのOctaviaバージョンが 0.8.1
  - Newton v0.9.1, Ocata v1.0.0
- lbaas-loadbalancer-create時にAmphoraインスタンスが作成されるがPROVISIONING_STATUSがERROR
- **LBやpoolの削除が失敗し削除出来ない**
- (厳密にはトラブルではない) PROXY Protocolが利用出来ない
  - HTTPSをインスタンスで処理したい場合利便性が低下する

問題4: Octavia v0.8.1

- 現時点で最新の1.0.0.0b2を利用
- Neutron LBaaSとの連携は諦めた (APIバージョンが合致しない)
- Octaviaを簡易に扱えるクライアントを作成しようと検討中
  - 現状だとcurlでOctavia APIを操作することでLBを作っている
  - そもそもPROXY Protocolを使用するにはOctavia APIを直接叩くしかない

マイグレーション状況

マイグレーション進捗

- 2環境間の通信が行えるようになった
- 状態やデータを保持していないロールから順次移行
  - WebサーバやAPIサーバ
- APIサーバがMitakaにあり、データベースサーバがHavanaにある、というサービスも増えてきた

マイグレーション方法

- M・H間で直接インスタンスを移動させる方法は今のところない
  - そのため、最初は状態を持っていないロールから取り組んでいる
- HavanaのインスタンスをGlanceイメージ化してMitakaでたてる、というツールの作成を検討
  - この場合はインスタンスの停止が伴う
- データベースサーバの移行など今後の課題

規模感

- Nyah-classic (Havana)
  - compute node : 154
  - instance : 616

- Nyah (Mitaka)
  - compute node : 36
  - instance : 426

その他改善状況

Live-Migration (block based)

- 無停止のインスタンスマイグレーション
- Cinder Volumeのような共有ストレージを利用しないパターン
- インスタンスのディスクイメージやメモリ情報がqemuを介して転送され切り替わる
- 検証では切り替わり時は5パケットほどで、通信に大きな影響は見られない
  - sshdのセッションやデータベースのレプリケーションが切れない

問題5: コンソールポート衝突

- マイグレーション時にインスタンスに紐付いているSerial consoleポートがcompute node間で衝突する場合がある
- ポートが衝突するとマイグレーションに失敗する
- https://bugs.launchpad.net/nova/+bug/1455252 にてバグ報告がされている

問題5: コンソールポート衝突

- Novaの設定でインスタンスに払い出されるconsole portの範囲指定が可能
- 各compute nodeで範囲が重複しないように少しずつスライドさせて設定することで回避出来た
- Newtonからはfixされている

Cinder

- Cinderが利用可能になった
- 現在のバックエンドはDell EMC ScaleIO
- 弊社GitHub Enterpriseのインスタンスは起動イメージ・データ領域両方ともこのCinder Volumeに配置し稼働させている
- データベースのデータなどIOPSが必要なシーンで利用していく予定

HCI (Hyper-Converged Infrastructure)

- compute nodeがScaleIOのstorage nodeも兼ねる
- 分散データを置くデバイスにはNVMeを利用している
  - hdparmの簡易速度チェック平均1140MB/sec=9120Mbps
  - SATA3.0の6Gbpsを超えるスペック
- ScaleIOの分散で10Gbaseのストレージ用ネットワークの方が先に帯域が足らなくなりそうだが、まだそこまでヘビーには使われていない

まとめ

自前構築・2環境並列運用を経て

OpenStackのコードが読めないとツラい

- バグを踏んだ時、新しい機能を使おうとした時、共に
- 後者の場合ドキュメントされていない、Blueprintだけも多い
- 今回特にNeutron含めネットワークのトラブルシュートに時間を割いた
- ログの調査に加えてOpenStackのコードを読む時間が増えた
- 自前で構築・運用するという場合には覚悟が必要

継続的にバージョンアップしないとツラい

- 紹介した問題や機能で多かった「新しいバージョンだったら直っている・使える」に対応していきたい
- ユーザ管理やセキュリティ対応など2環境を並行運用する手間はかかる。1環境を集中して運用したい。
- 2環境間マイグレーションにはパワーが必要。年内いっぱいは続いていきそうな見通しなので、毎度は出来ない。
- **今ある環境を段階的にアップグレードしたい**

楽しいこともある

- “クラウドのプロバイダ”としてのやりがい
- クラウドの作り方が学べる
- クラウドの中の人の気持ちが分かる
- 問題解決のレイヤが増やせる
- SDS, SDNなどソフトウェアでの実装を知ることが出来る
- 総合力が試される

今後の予定

- OpenStackアップグレードしやすい仕組みづくり
  - コンポーネントのコンテナ化 (Kolla, Kubernetes)
- ScaleIO以外のSDSを導入し全台HCIにする
  - 現在のところDRBD9を検討中
- HCIに伴ったVM-HA
- インスタンスのよりよいスケジューリング方法

プライベートクラウド運用も
”もっとおもしろくできる”

最新の採用情報をチェック→ @pb_recruit