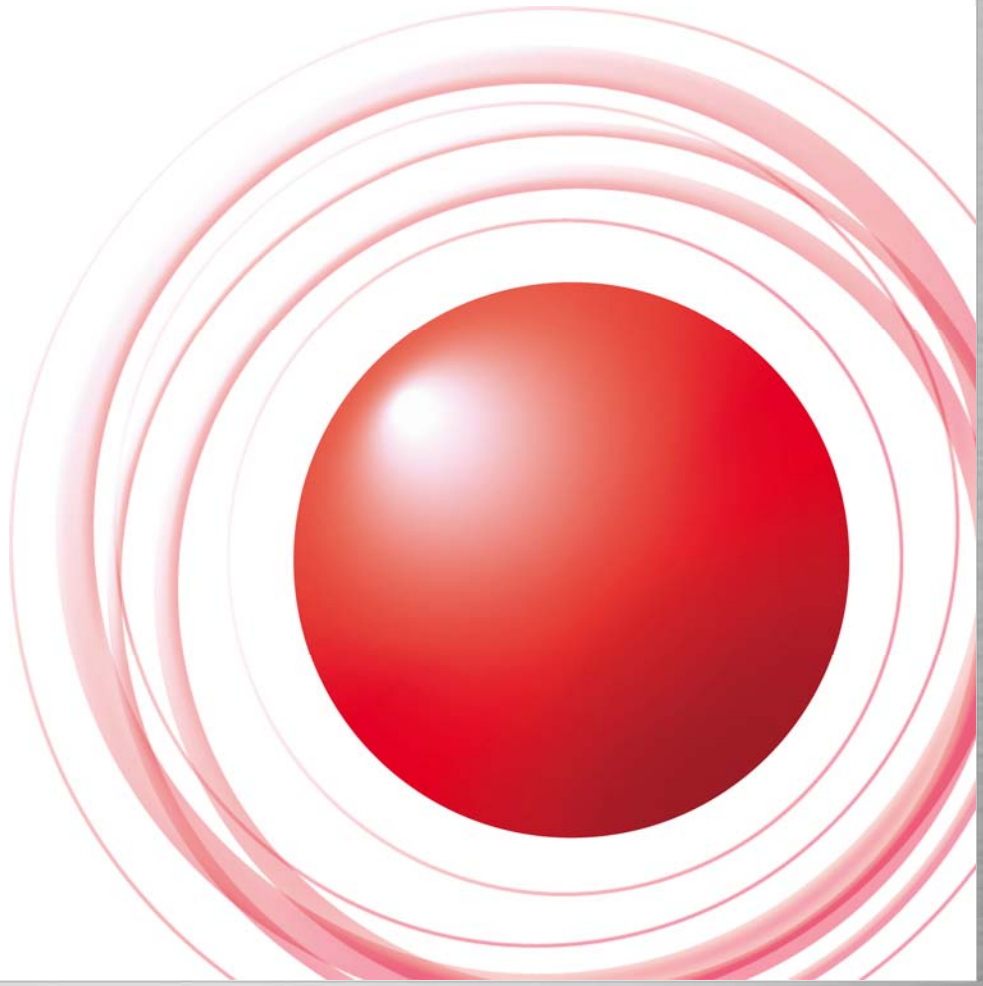
IIJGIOのHaaS基盤について

IIJ Internet Initiative Japan



2011/1/31
IIJ プラットフォームサービス部 サーバプラットフォーム課
牧野 泰光　花高 信哉　阿部 博

Ongoing Innovation

進行

1. NHN(次世代ホストネットワーク)導入以前

・サービス毎に個別設備を構築し、運用を実施していた

2. NHN導入によるサーバ運用の変化・導入効果

・基盤システムを構築し運用効率化を推進
・サーバプール導入による短時間でのリソース増減を可能へ

3. NHNからIIJGIO(顧客提供)に向けた変更

・API制御導入によるデリバリの自動化・運用効率化を推進
・リソースの公平性制御の導入
・マイグレーション運用の導入による保守性向上

4. まとめ

NHNとは？

IIJでは2008年度“NHN”というキーワードで運用の効率化を目指しサービスホスト構成、ホスト運用ネットワークの設計見直しを行った

名称の由来

- Next Host Network: 次世代ホストネットワークの頭文字

NHN導入前の状況

- IIJサービス用に東京都内の複数のデータセンタに分散して200ラック以上、数千台のサーバを利用していた
- サービス単位でラックを確保して、設備を構築していた
  - 構成変更等がある度に、現地作業に出かけていた
  - 将来の需要増を見越して余裕をもたせる必要あり
  - サービスが軌道に乗らなかった時、設備転用が困難
  - 原則機器は保守契約を行い、保守費が高騰
- 物理システムへのアクセスの容易さから東京近郊の立地のよいデータセンタを利用しサーバ運用を最適化していた

規模の増加に伴い人員を増強するのではなく、効率化の検討を始めた

NHN導入の背景

遠隔データセンタの利用を視野に入れ、現時点で最適なものを検討

電力事情

- 都内のデータセンタは**供給電力、空調能力が不足気味**
- 場所に依存しない**サービス用機材は遠隔データセンタに**行くべき？

ランニングコスト削減

- ここ最近、東京近辺のデータセンタは顧客からの引き合いが強い
- **立地のよいデータセンタ**は、可能な限り**顧客に提供**したい
- データセンタを郊外に持っていくことで、スペース費用、人件費、電気代等を削減できる可能性が高い

ネットワーク事情

- 多重化技術の進化により広帯域でもネットワークコストは抑えられるようになってきており、郊外データセンタの検討を後押し

現状の即時駆けつけ対応前提に最適化した構成から、
遠隔地になってもうまく回せる仕組みを作る必要あり！

遠隔データセンタ利用を視野に入れた機器構成

今までの運用チームの経験を元に、機材選定時に重視する点、許容できる点を検討

サーバ機器

- サーバ機器は時々故障する。故障ポイントは多岐に渡り故障率の削減には限界がある。故障しても影響の少ない構成にすることが望ましい

ストレージ機器

- ストレージ機器に関しては、サービス停止に繋がる故障が起きないよう、今までより信頼性の高い構成にする必要がある

ネットワーク機器

- ホスト収容エッジスイッチの故障は、これまでの経験上さほど多くない。NIC冗長化設定などは必要なときのみ実施しシンプルな構成に

※IIJで取った機材故障の統計情報は技術レポート(IIR)でも公開しています

http://www.iij.ad.jp/development/iir/pdf/iir_vol05_service.pdf

サーバは故障を前提に、ストレージは信頼性の高いもの、ネットワークは複雑なことをせずシンプルに基盤システムとなりうるものを再設計

NHN の設計方針

現地作業を集約し基本は遠隔対応。そして流行の技術を盛り込む

サーバ機器

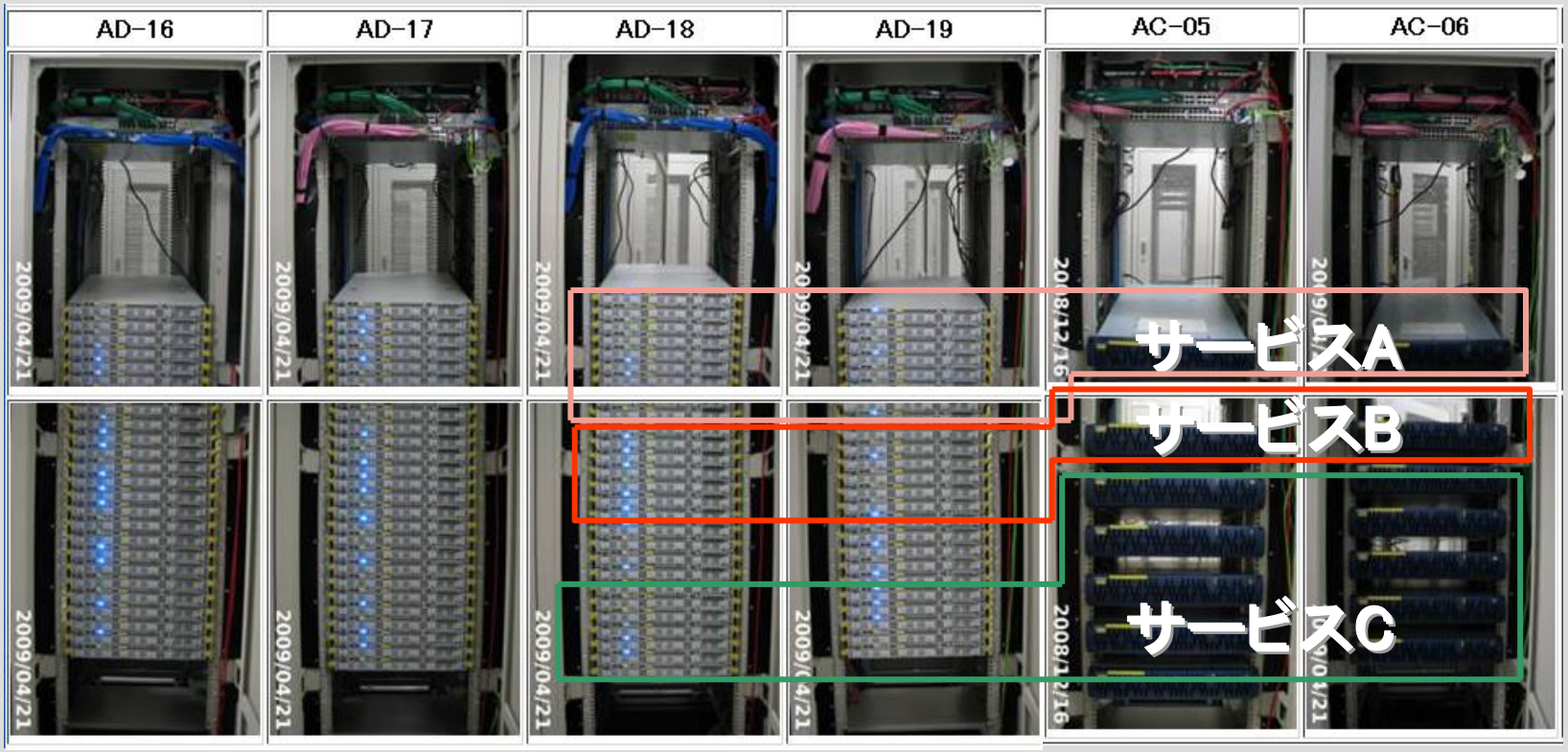
- サーバプール方式の導入による現地作業の集約
- サーバ構成の画一化による構成管理作業の抑制
- Xen,OpenVZ など仮想化技術との組み合わせによる自由度の向上
- 省電力サーバの導入によるラック辺りのサーバ収容数の向上
- サービス単位のラック割りをやめラック辺りのサーバ収容数を向上
- 外部業者に機器設置や交換作業の委託を目指す（シンプルな構成）
- iSCSIを使った安価なSANを構成。コントローラ、パスの二重化を必須
- VLAN を使い、現地での配線作業なしに論理ネットワーク構成を可能
- サーバはディスクレス構成にして、故障時には切り替えで一次対応
- OSのインストールを含め設置以降の作業をリモートから実行可能に

プール機材から利用することに慣れるなど利用者側の意識改革が必要

NHN利用イメージ

需要に応じてサーバ数、ストレージ容量を柔軟に変更可能

- リモートから設定を投入することで**数時間～数日程度**でサーバが利用可能
- iSCSIストレージ一台を複数の物理サーバで共用して利用している
- 仮想化(Xen,OpenVZ)と組み合わせ物理サーバを分割して利用可能

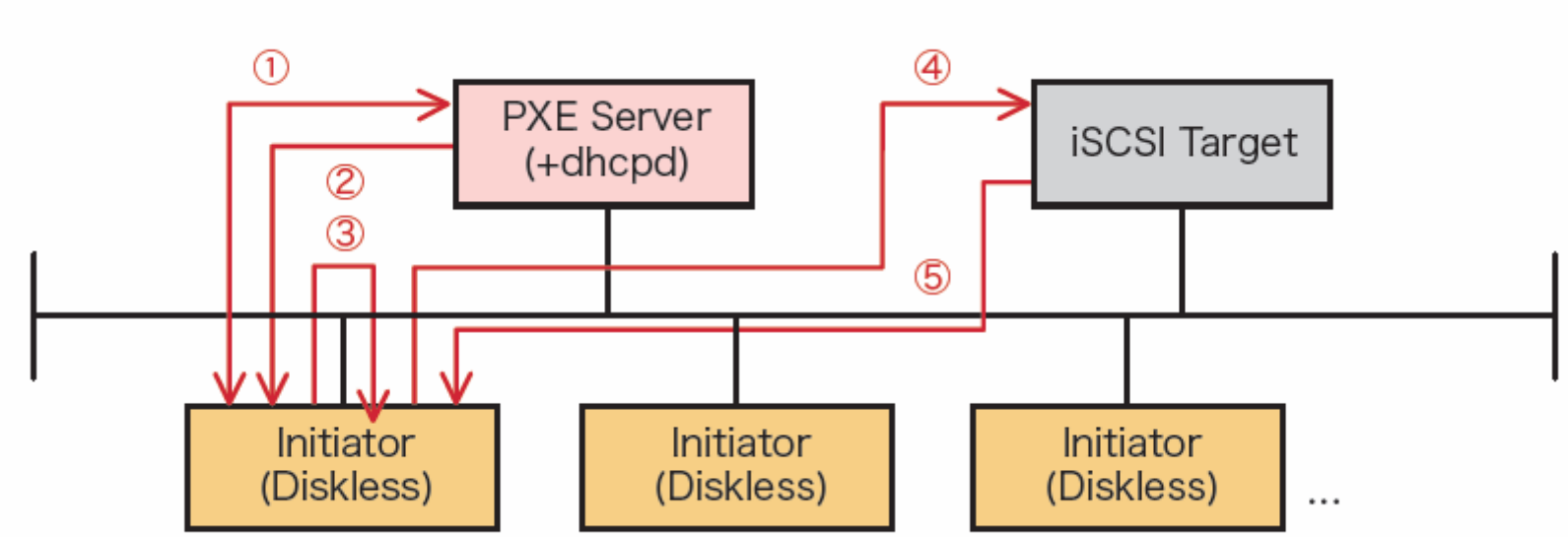


PXE boot と iSCSI ストレージを利用したディスクレスサーバの実現

安価かつサーバ障害時に内容を簡単に他のサーバに切り替え可能

ディスクレスサーバ実現に向けた技術的な取り組み

- **PXEブート後、iSCSI領域内のOSを起動する仕組み**を作成。サーバ標準のNICで起動可能にして iSCSI HBA 等の追加は不要
  1. DHCP でアドレスとパラメータを取得する
  2. TFTP から kernel と initrd イメージを取得する
  3. Initrd の ramfs root 環境で kernel を起動する
  4. DHCP パラメータを元に iSCSI イニシエータを初期化する
  5. ブロックデバイスとしてマウントし、マウントした場所へ switchroot する

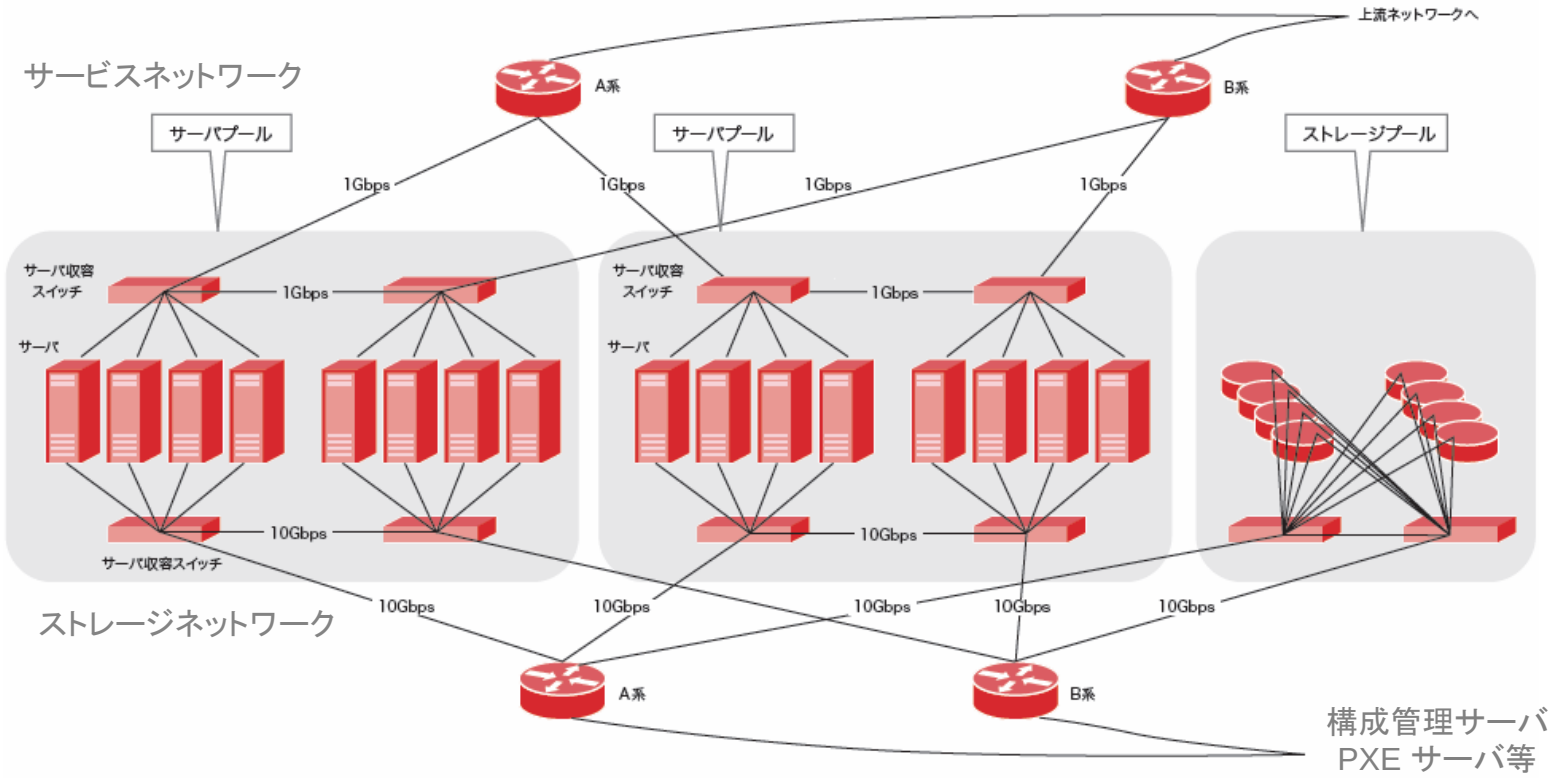


VLANを使った配線変更不要な仮想ネットワークの実現

現地での物理配線変更なしに柔軟なネットワーク構成が可能

構成情報とネットワーク機器のVLAN設定を連動

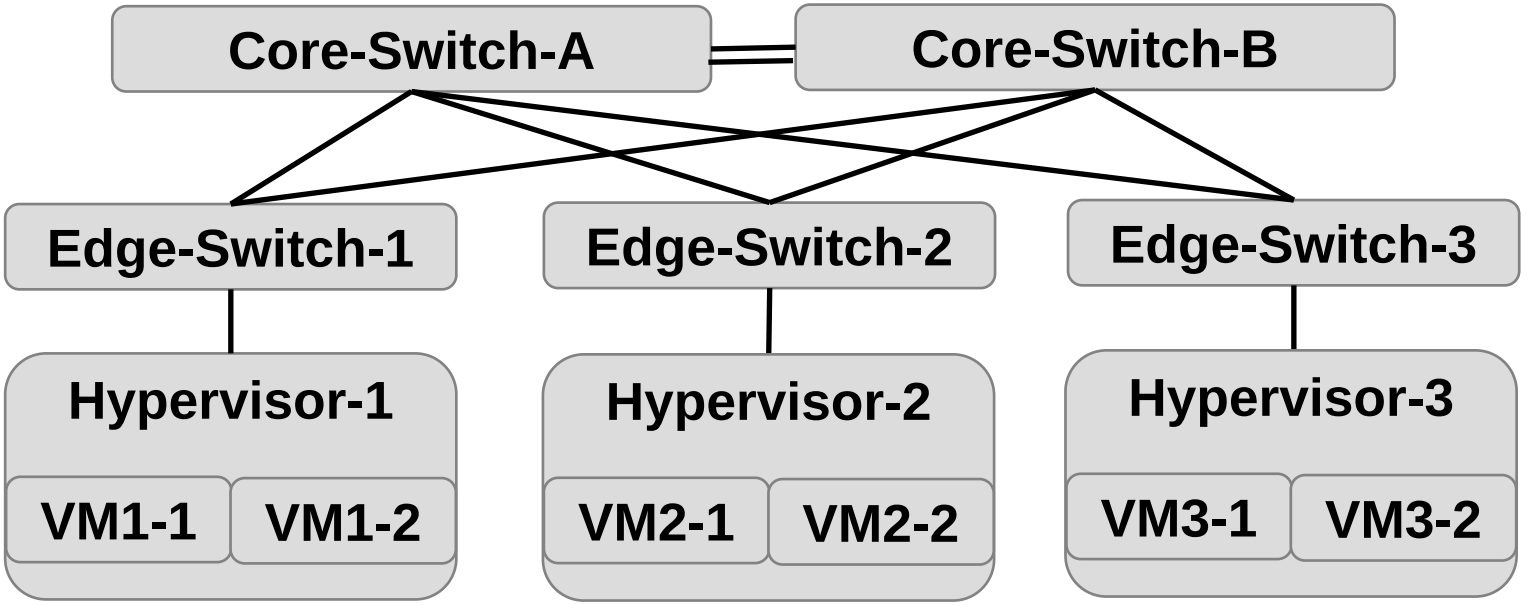
- IIJが管理する**構成情報と連動して自動的にスイッチのVLAN設定を書き換える**仕組みを作成。設定ミスを減らし運用コストを削減した



VLAN制御の例 (物理構成)

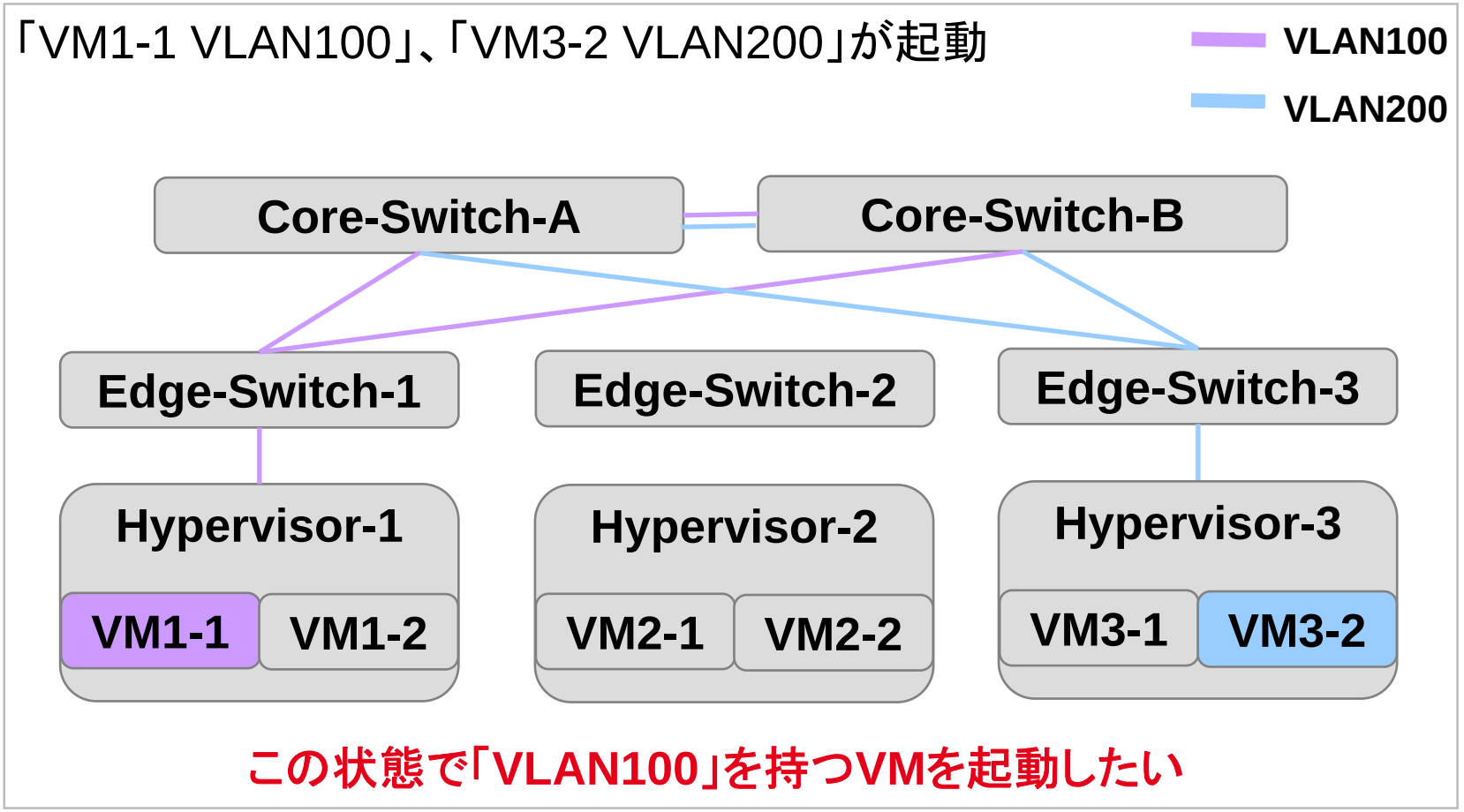
人間ではとても対応できない巨大L2ネットワークの自動制御の例

物理ネットワーク構成　(これ以降物理接続は省略)



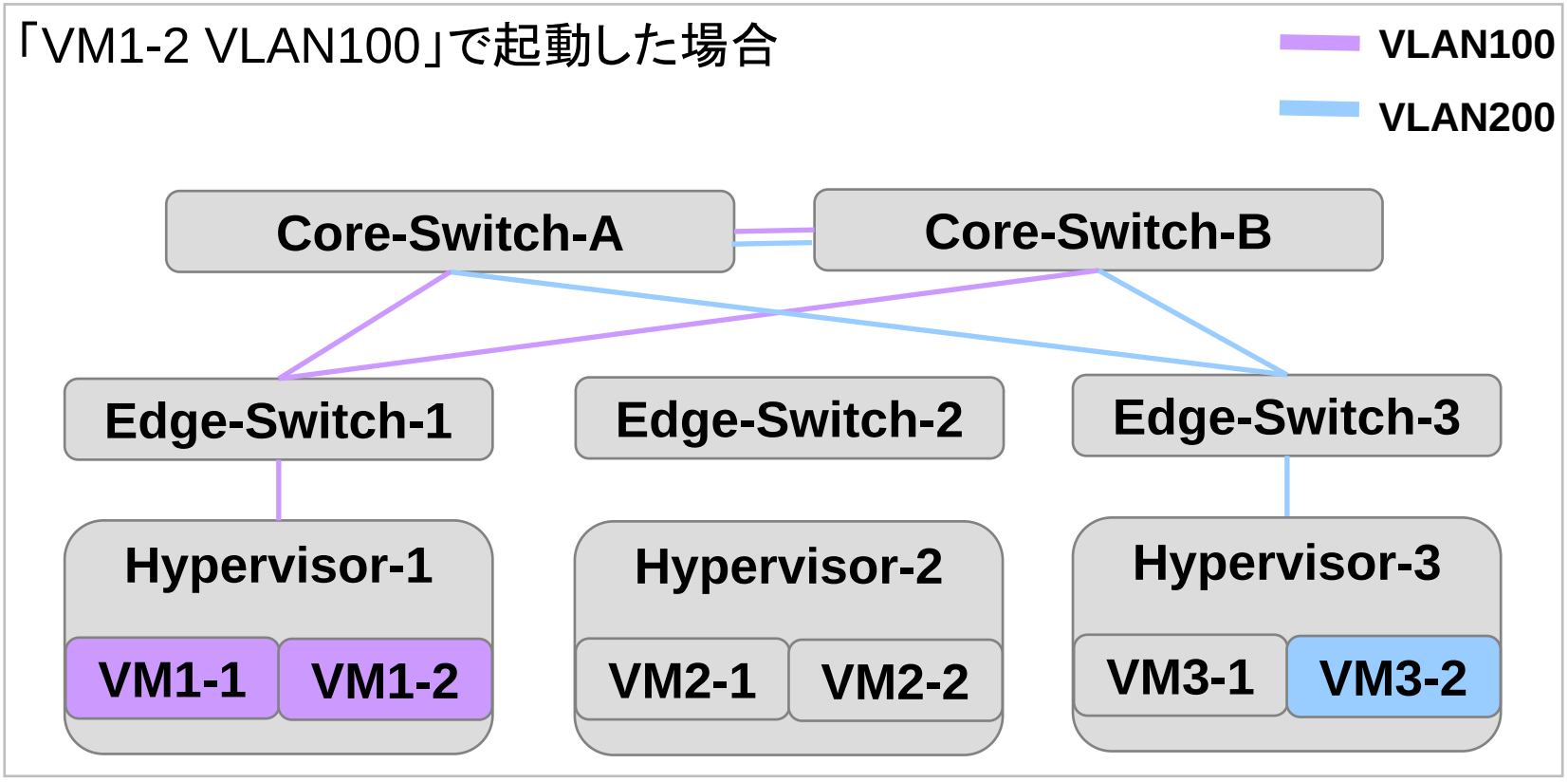
VLAN制御の例（1）

エッジNW機器は最大MAC数に余裕がなく消費を抑える工夫を実施



VLAN制御の例（2）

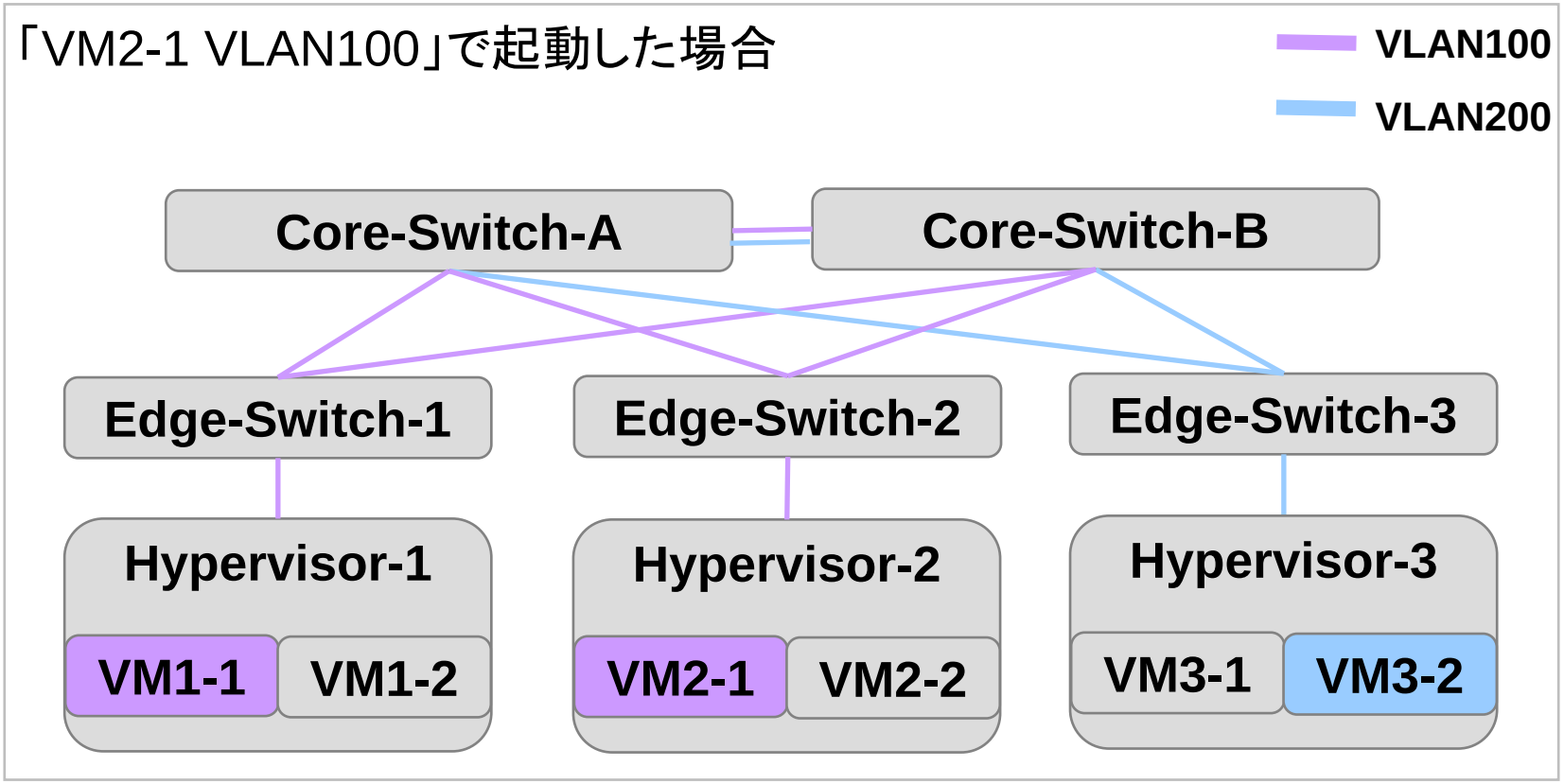
コアNWは全疎通、エッジNWは疎通が必要なもののみ自動設定



同じ VLAN を持つものはなるべく同じ物理ホストに収容する

VLAN制御の例（3）

HaaS制御基盤で自動計算して、ネットワーク機器に自動で設定投入



ライブマイグレーション時や機器故障時でも人手で機器の設定変更は不要

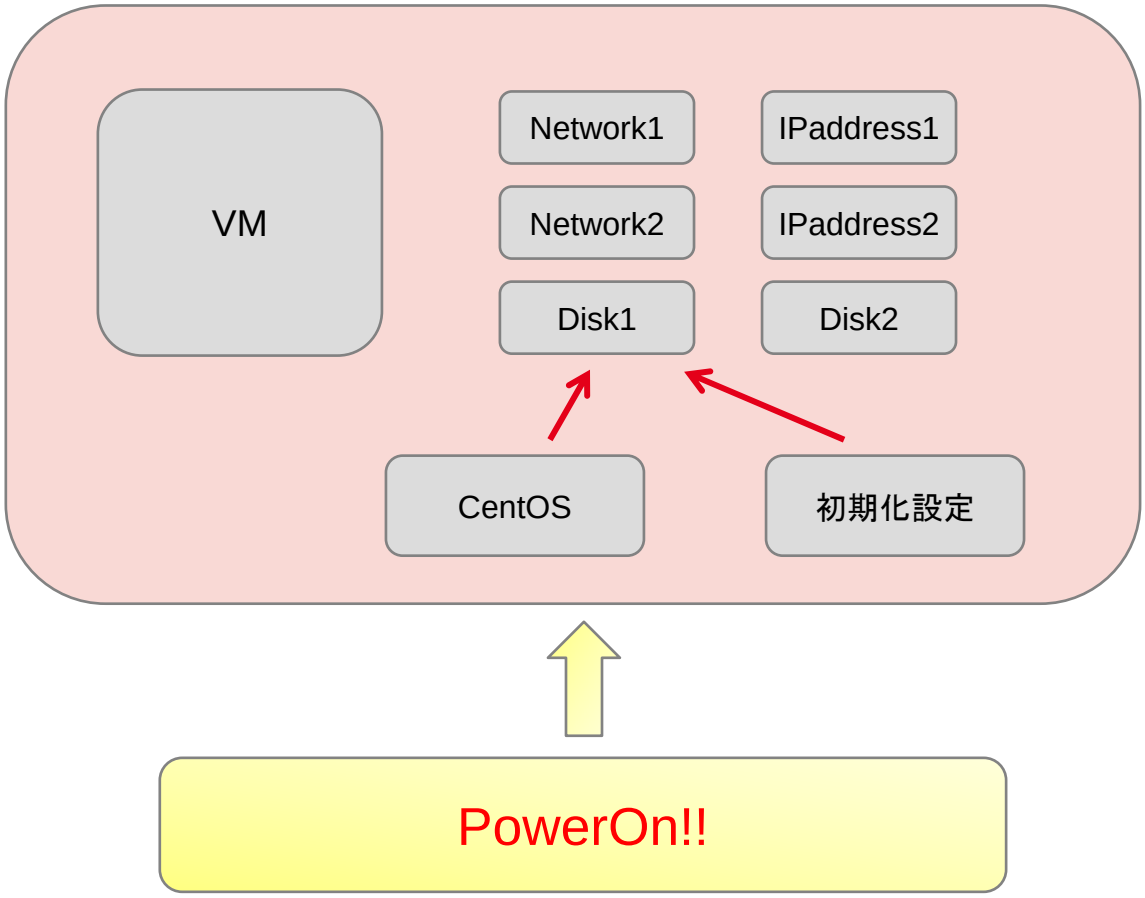
NHNからIIJGIO(顧客提供)に向けた変更

API制御導入によるデリバリの自動化・運用効率化を実現

- リソース確保からPowerOnまで全てAPI経由で行う
  - GIO HaaSシステムを利用するための唯一の手段
- 外部からXML-RPCリクエストを受け付ける
  - 受け口なので落ちるとまずい
  - リクエストをそれなりにさばけないといけない
  - 一部APIでVMの状態監視などを行う人もいる
- XML-RPCの内容を精査し、必要な処理を行う
- 論理データの管理
  - サーバ構成情報管理、リソース管理
- ファシリティ層からのコールバックを受けつける
  - 非同期処理の確認
  - サービス層は末端クライアントに対してコールバックは行わない
    - API利用者はポーリングにて非同期処理の状態を確認する
  - 非同期処理用に終了状態をサービス層は知らないといけない

HaaS システムイメージ

仮想ハードウエアにNICやディスクを付け、組み立てるイメージ



API種別（VM管理、ネットワーク管理）

HaaS制御基盤で制御対象を自動計算して、機器に設定投入

VM管理API	同期/非同期	解説
getVmList	同期	VMリスト取得
getVmInfo	同期	VMの詳細情報取得
getVmStatus	同期	VMステータス取得（on/off, コマンドステータス）
allocateVm	同期	VM確保
deallocateVm	同期	VM解放
powerOnVm	非同期	VMパワーオン
powerOffVm	非同期	VMパワーオフ
resetVm	非同期	VMリセット
restoreImage	非同期	イメージインストール
changeVmType	同期	VM品目変更
changeVmArchitecture	同期	32bit <-> 64bit変更

ネットワーク管理API	同期/非同期	解説
getNetworkList	同期	ネットワークリスト取得
getNetworkInfo	同期	ネットワークの詳細情報取得
allocateNetwork	同期	ネットワークリソース確保
deallocateNetwork	同期	ネットワークリソース解放
allocateIpAddress	同期	IPアドレス確保（共用L3ネットワークのみ）
deallocateIpAddress	同期	IPアドレス解放（共用L3ネットワークのみ）
attachNetwork	同期	VMへのネットワークアタッチ
detachNetwork	同期	VMからのネットワークデタッチ
assignIpAddress	同期	IPアドレス割当（共用L3ネットワークのみ）
removeIpAddress	同期	IPアドレス割当解除（共用L3ネットワークのみ）

※より詳しいことはThinkIT 第2回 クラウドサービスにおける自動制御基盤（HaaS API）の記事でも公開しています。http://thinkit.co.jp/story/2010/06/10/1605

API種別（ストレージ管理、初期化、イメージ管理、その他）

API のみの範囲でサービスを組めば、オンラインサービスも可能

ストレージ管理API	同期/非同期	解説
getVolumeList	同期	ストレージリスト取得
getVolumeInfo	同期	ストレージの詳細情報取得
allocateVolume	同期	ストレージリソース確保
deallocateVolume	同期	ストレージリソース解放
attachVolume	同期	VMへのストレージアタッチ
detachVolume	同期	VMからのストレージデタッチ

初期化API	同期/非同期	解説
setIpAddress	非同期	NICへのIPアドレス設定
unsetIpAddress	非同期	IPアドレス設定解除
setNetwork	非同期	ネットワーク設定
setPassword	非同期	パスワード設定
setSshKey	非同期	ssh key設定
setIptables	非同期	iptables設定
setSshdConfig	非同期	sshd config設定
writeFile	非同期	ホワイトリストで指定するファイル設定

イメージ管理＆その他API	同期/非同期	解説
getImageList	同期	VMイメージリスト取得
getImageInfo	同期	VMイメージ情報取得
getRequestIdInfo	同期	リクエストID情報取得
getResourceComment	同期	リソースへのコメント追加

GIOホスティングパッケージはAPI範囲のみで実現。オンラインで利用可能

XML-RPC API仕様例（getVmInfo）

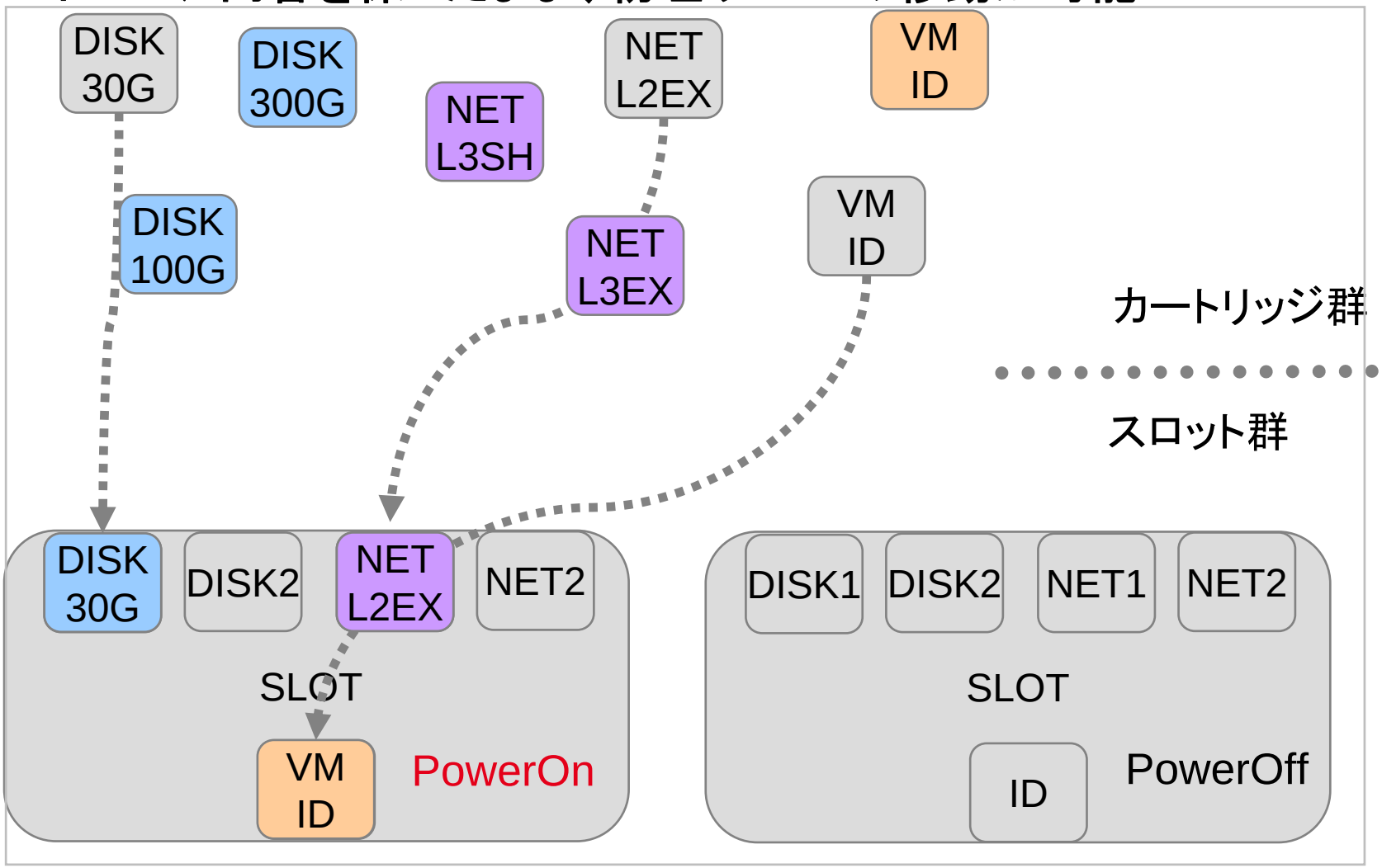
VM情報取得(getVmInfo / APIメニューID: 002)

- リクエスト種別: 同期
- 前提条件: 特になし

```
□ 入力
HRB.getVmInfo {
    IaasId,                             # IaaS事業者ID(必須)
    VmIds (VmId, ....)                  # VM IDリスト(必須)
}


□ 出力
$RESULT {
    StatusCode,
    ErrorCode,
    RequestId,
    Result {
        VmInfoList ( {
            VmId,                       # VM ID
            VmMenuId,                   # VMの品目ID
            VmLocation,                 # VM 収容先 (L/R)
            Architecture,               # null/32bit/64bit
            InterfaceInfos ( {          # インタフェース情報
                InterfaceId,            # インタフェース番号(0/1/2/3/4/5/6/7 :
                                        #                    0=eth0, 1=eth1, 2=eth2, 3=eth3, 4=eth4, 5=eth5, 6=eth6, 7=eth7
                                        #                    ただし、attachNetworkされているインタフェース番号のみ戻値として返す)
                NetworkId,              # Network ID
                AssignIpAddress(IpAddress, ...),  # VMにアサインされたIpAddress
            }, ....),
            VolumeInfos ( {
                SlotId,                 # ディスクスロット番号(0/1/2 : 0=基本ディスク, 1=拡張ディスク1つ目, 2=拡張ディスク2つ目)
                VolumeId,               # ストレージID
            }, ...),
            Comment,                    # VMに対するコメント
            AllocateDate                # 作成日(YYYY/MM/DD HH:MM:SS)
        }, ....)
    }
}


□ エラーコード
- INVALID_ID                            # 不正なIDの利用

$RESULT {
    StatusCode,
    ErrorCode,
    RequestId,
    Result {
        ErrorList (VmId, ...)           # エラーになったVM ID
    }
}
```

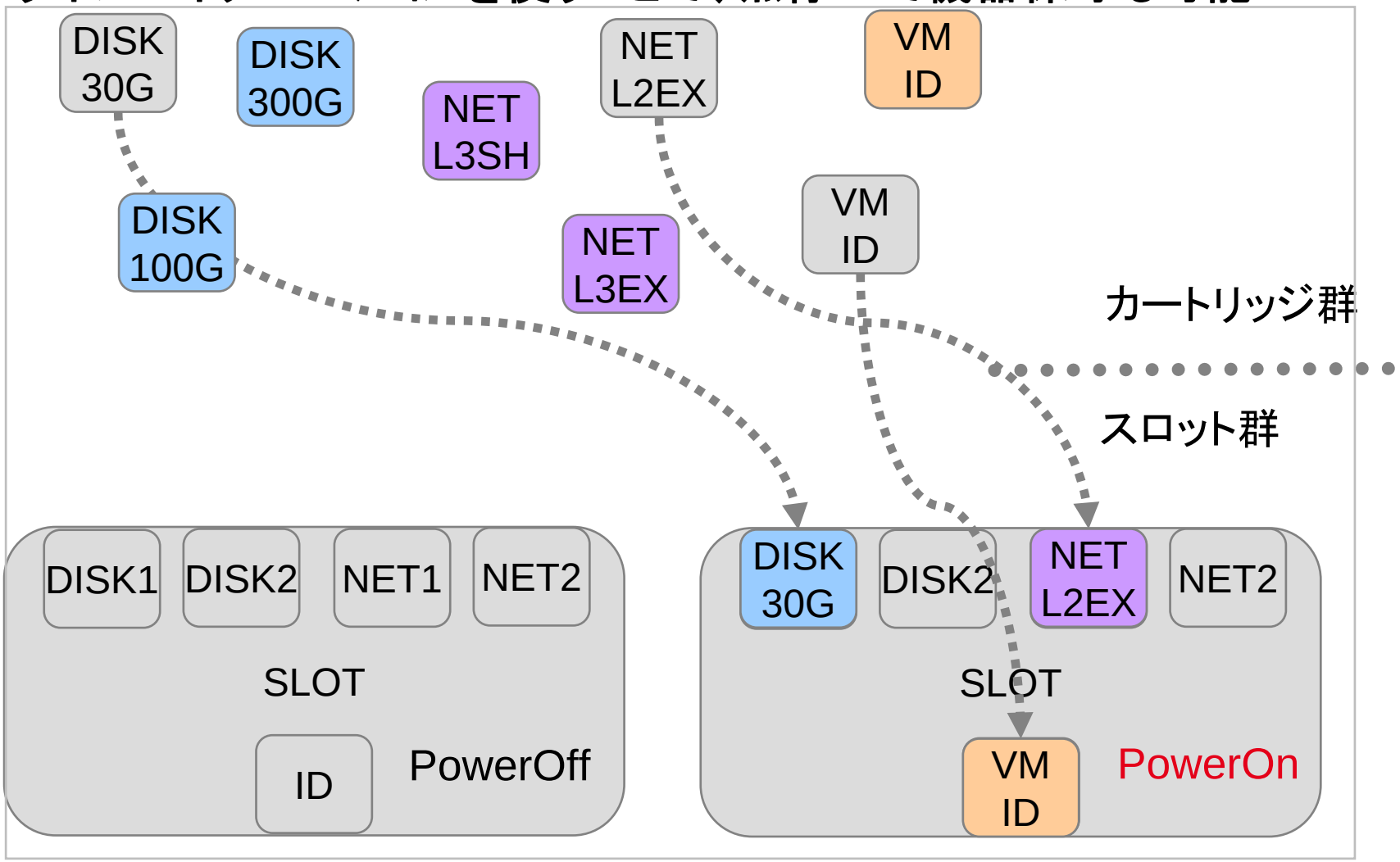
マイグレーション運用のイメージ（1）

ストレージ内容を保ったまま、物理サーバの移動が可能



マイグレーション運用のイメージ（2）

ライブマイグレーションを使うことで、無停止で機器保守も可能



機器構成についてよく聞かれること

仮想化運用を前提に、単純な構成でも充分な品質を確保

よく聞かれる事項

- サーバの電源ユニットが冗長構成でないが大丈夫？

→ **電源故障時は、別のサーバで SAN ブート**することで短時間で対応可
（電源単体の故障率は低く、冗長構成にすることで消費電力も上がる）

- サーバ収容ネットワーク（エッジ）が冗長化されていないが大丈夫？

→ エッジスイッチ故障時には、エッジスイッチ収容サーバ全てを他のサーバに移動して対応。**計画作業の場合は、ライブマイグレーションを前提**

- サーバの保守契約が24h365d対応でなくて大丈夫？

→ **リモートから操作して一次対応**を行い、営業日にまとめて保守対応

- ローカルディスクやSSDを積むことはできないの?

→ **SANブートやマイグレーションができなくなる**ため対応できません

構成を簡単、安価なものにしても運用方法の変更で充分な品質に！

NHN および IIJGIO HaaS 基盤の展開（主なもの）

2008年上期(4月～9月)

- 設計開始。技術検証および機器の選定を実施

2008年10月

- 新規サービス等の開発環境兼NHN実証環境として稼動開始

2008年12月

- NHN本番機材の構築完了。IIJセキュアWebゲートウェイサービス、IIJサービスオンライン機材の一部をNHN上に構築開始

2009年1月末

- IIJセキュアWebゲートウェイサービス顧客提供開始。その後の新規サービスおよび既存サービスの移行先として増強しながら継続稼動

2009年度

- 顧客提供に向け、ネットワーク設計および制御システム部分の再設計・開発および物理構築作業、試験等を実施

2010年4月

- IIJGIO の仮想化サーバサービス(Vシリーズ)の基盤システムとして稼動開始

NHNの導入効果

当初は今までとの違いが浸透しておらずとまどいもあったが、現在では社内に浸透し、短時間で利用できるメリットが理解された

IIJ内部から導入当初聞こえた意見

- 仮想化サーバでは不安。物理サーバにして欲しい
- これほどの性能は必要ないため、もっと安価なサーバにして欲しい
- メモリ量が多すぎるので減らして欲しい
- ローカルディスクに比べてディスク単価が高い

NHNの導入効果

- 仮想化の導入による集約、ファシリティ、運用コストまで含めたコストを考えれば安価と理解された
- サーバを要求して数日以内に利用可能になるなど利点が浸透
- 同様に計画変更などで不要になった場合でも即座に返却可能
- DCは遠隔地になったが、うまく構成すれば十分信頼性が確保できる

サービス開発フロー、設備増強の需要予測の仕組みが変わりつつある

まとめ

- IIJでは2008年度サービスホスト構成の見直しを実施(“**NHN**”と命名)
- NHNでは**遠隔DC利用を視野**に十分な品質を保ちつつコスト削減を実施
  - 一括構築およびサーバプール方式の導入
  - 基本的な**対応はすべてリモート**から可能にする
  - サーバは**ディスクレスで運用**
  - 仮想化の導入、ラック辺りの収容数向上により**DCコストを圧縮**
- IIJでは運用視点で**いろいろなものを自作して運用効率化**に励んでいる
  - PXE と iSCSI ストレージを使った IP SAN の実現やネットワーク機器のVLAN動的変更の仕組みなど**自分達でサーバ管理や制御に必要な仕組みを作って**運用コストの削減を行っている
- NHNの成果を元に IIJGIO を設計・開発し2010/4からサービス開始した
  - **API制御の導入によるデリバリの自動化・運用効率化**を推進した
  - 顧客利用を前提に**リソースの公平性制御を導入**
  - マイグレーション運用の導入により保守性を向上させた

IIJ Internet Initiative Japan

ご清聴ありがとうございました

お問い合わせ先　IIJインフォメーションセンター
TEL：03-5205-4466（9：30～17：30 土/日/祝日除く）
info@iij.ad.jp
http://www.iij.ad.jp/

Ongoing Innovation

IIJ Internet Initiative Japan

インターネットの先にいます。

IIJはこれまで、日本のインターネットはどうあるべきかを考え、
つねに先駆者として、インターネットの可能性を切り拓いてきました。
インターネットの未来を想い、イノベーションに挑戦し続けることで、世界を塗り変えていく。
それは、これからも変わることのない姿勢です。
IIJの真ん中のIはイニシアティブ ―――――― IIJはいつもはじまりであり、未来です。

Ongoing Innovation

お問い合わせ先　IIJインフォメーションセンター
TEL：03-5205-4466（9：30～17：30 土/日/祝日除く）
info@iij.ad.jp
http://www.iij.ad.jp/