

# 「さくらのクラウド」の舞台裏

さくらインターネット研究所

大久保 修一

ohkubo@sakura.ad.jp

# 自己紹介

- 1980年10月生まれ **（ワカモノではありません）**
- 2003/4～ さくらインターネット入社  
バックボーンネットワークの運用
- 2009/7～ さくらインターネット研究所に異動
- 2011/3～ クラウドの開発に携わる  
– 主にネットワーク、ストレージ担当

# Agenda

- さくらのクラウドとは？
- ストレージの話
- 第2ゾーンの話
- DoSアタック対策の話

# さくらのクラウドとは？

## IaaSの基本的なリソースを提供



### サーバ

- 1コア/1GB～12コア/128GB
- 全42種類
- UNIX系OS各種、Windows



### ネットワーク

- 共有グローバルセグメント
- 専用グローバルセグメント
- スタティックルート
- ローカルスイッチ
- ロードバランサ
- パケットフィルタ



### ストレージ

- SSD 20GB, 100GB
- HDD 40GB～4TB
- アーカイブ、ISOイメージ領域

全てAPIで自由に操作可能

これらの組み合わせで  
Software-Defined Data Centerを実現！

# ウェブサイト <http://cloud.sakura.ad.jp/>

**さくらのクラウド**  
SAKURA CLOUD

さくらのクラウドは、サーバーとストレージが1時間単位から使える、高性能で低価格なIaaS型クラウドサービスです

▶ [アカウント開設はこちら](#)

▶ [コントロールパネル](#)

▶ [さくらのクラウドを選ぶ理由](#)

▶ [サービス仕様・料金](#)

▶ [料金シミュレーション](#)

▶ [サービスご利用の流れ](#)

▶ [サポート情報](#)

Simple is beautiful

## さくらのクラウド

CPUとメモリを自在に組み合わせて使える、高性能なサーバーと拡張性の高いネットワーク。まるで手元に実際のサーバーやスイッチがあるような直感的操作が可能なIaaS型クラウド。

高性能なサーバーと拡張性の高いネットワークをインターネット上で自在に構築できるIaaS型パブリッククラウド。性能、安定性、即時性、拡張性、そしてなにより低価格であること。クラウドが持つ本質的価値だけをシンプルに表現したクラウドサービスです。



さくらのクラウド NEWS

▶ [【プレスリリース】さくらインターネット、「さくらのクラウド」で1時間9円からの時間課金をスタート](#)

**月額1,900円、1日95円、1時間9円から使える！**

**ブラウザの中の仮想データセンター**

さくらのクラウドは、サーバとストレージが1時間単位で使えるIaaS型クラウドサービスです。CPUとメモリを自在に組み合わせて使える自由度と、簡単操作のコントロールパネルで、自分だけの仮想データセンターを構築できます。

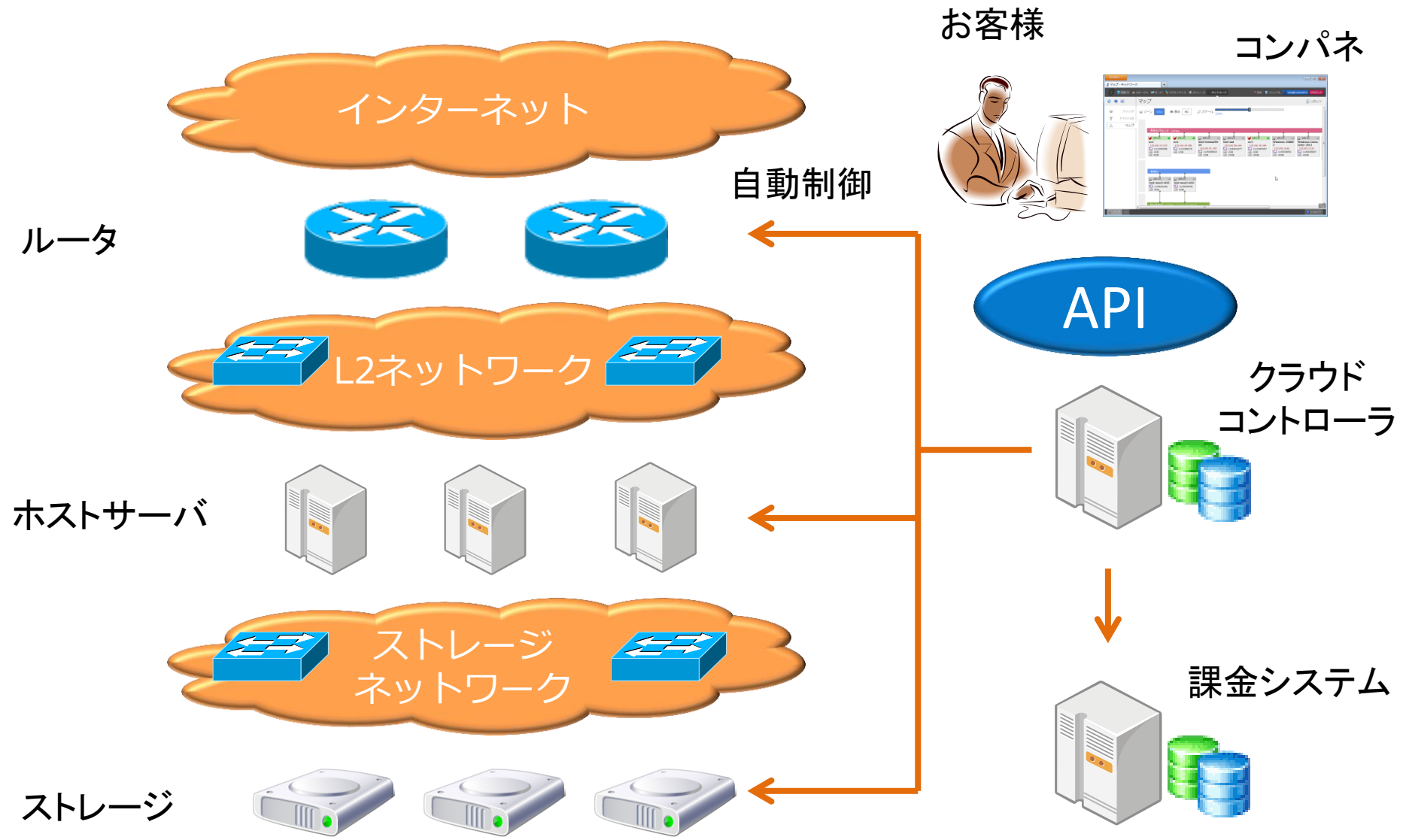
▶ [アカウント開設はこちら](#)

クラウドに関するお知らせ



簡単なデモします

# さくらのクラウド、システム構成



# 2年半いろいろありました

- 2011/3/2～ クラウドの開発本格スタート
- 2011/9/6～ ベータ提供開始
- 2011/11/15～ さくらのクラウド正式リリース
- 2011/12～ ストレージの障害が顕在化
- 2012/3～ サービスの無償化、  
新規申し込み受付停止



# 新ストレージ導入の経緯

2012/4

2012/5

2012/6

2012/7

2012/8

2012/9

2012/10

2012/11

2012/4頃  
新ストレージの開発、  
導入をスタート

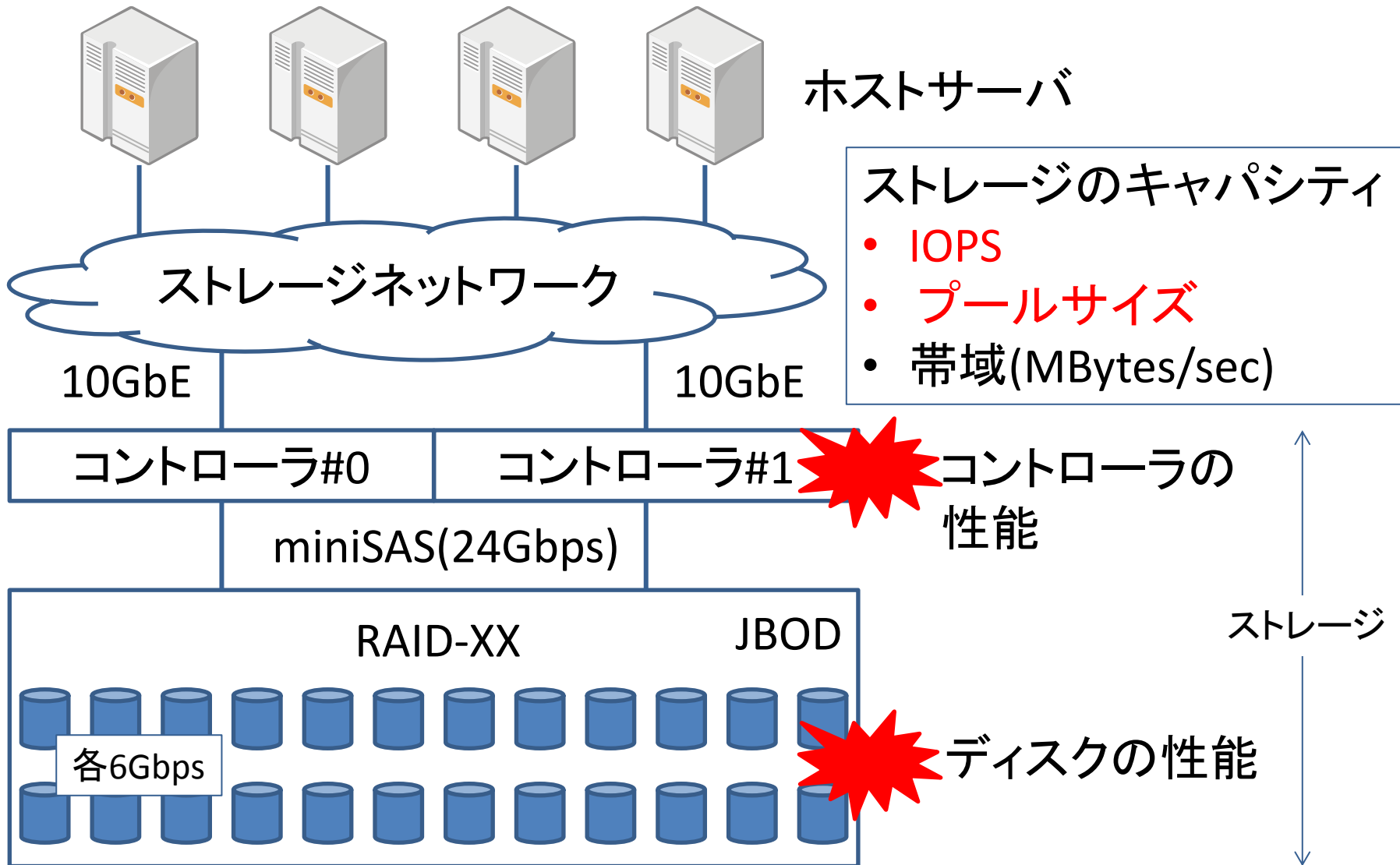
2012/6/25～  
第一期iSCSIストレージの  
β提供開始

2012/8/31～  
第二期iSCSIストレージの  
β提供開始

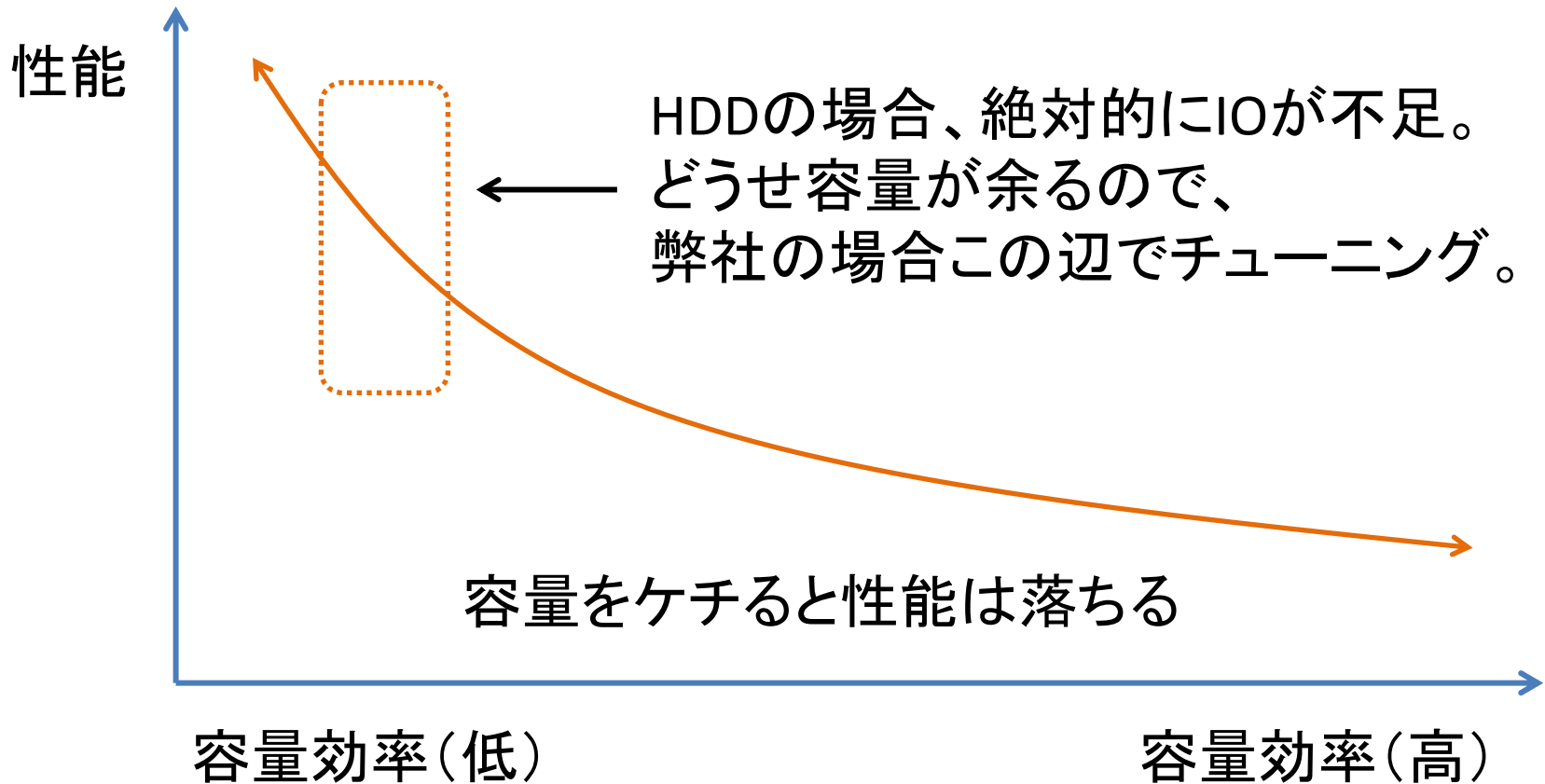
2012/10/1～  
既存ユーザの課金再開

2012/11/1～  
新規ユーザ募集再開  
(サービス正常化)  
SSDプランの提供開始

# ストレージのボトルネック



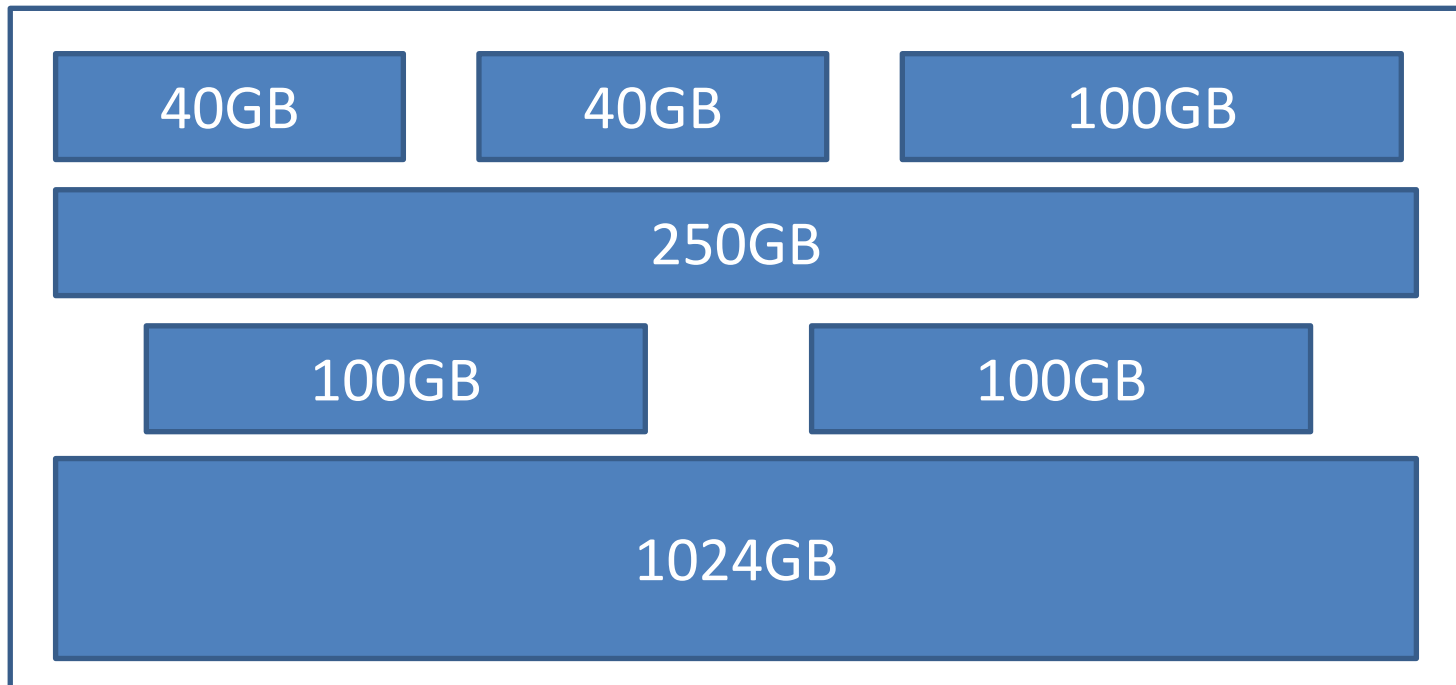
# 性能と容量効率のトレードオフ



シンプロビジョニング、重複排除、  
クローン、コピーオンライト、他

# ストレージ収容設計の難しさ

40～1024GBのユーザを最大N個詰めて、容量が余らないプールを作りたい ⇒ ナップサック問題を解く！

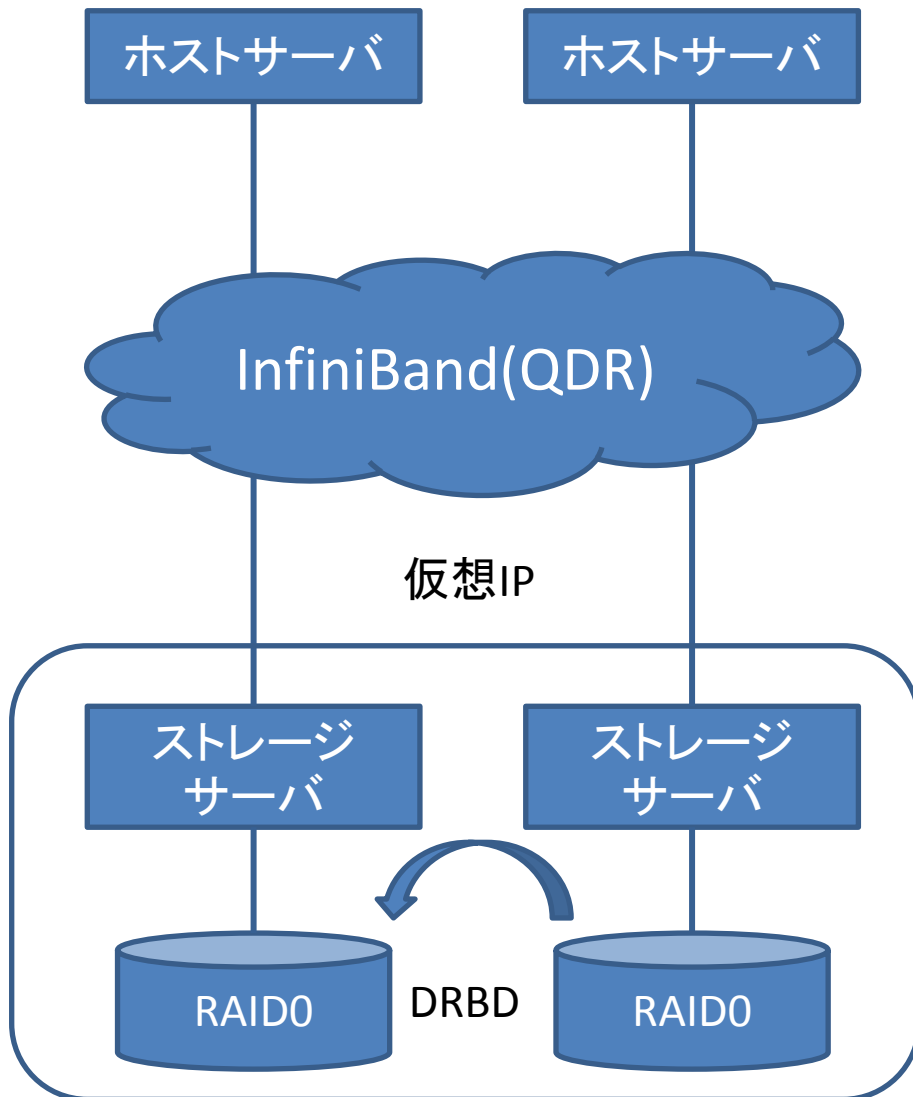


SAS XXXGB 2.5" 10Krpm XX本 RAIDXX  
X,XXXIOPS(最大XXXユーザ)、XX,XXXGiB

会場のみ

現在2種類のストレージを運用

# 20GB, 100GB SSDストレージ



- IPoIBによる接続
- ターゲットは仮想IP
- IB経由でデータのミラー
- 市販サーバにSSD搭載

- OS: CentOS 6.2
- HA: Pacemaker + DRBD
- ボリューム制御: LVM
- iSCSI Target: scsi-target-utils

# 40GB～4TB HDDストレージ

- NEC iStorage M300とM100
- 1年がかりでようやく導入事例出ました



質実剛健なローエンドストレージが商用クラウドを支える

性能と安定性でさくらが選んだNECの「iStorage M300」

<http://ascii.jp/elem/000/000/819/819520/>

詳しくは・・・

Google

さくらのクラウド iStorage

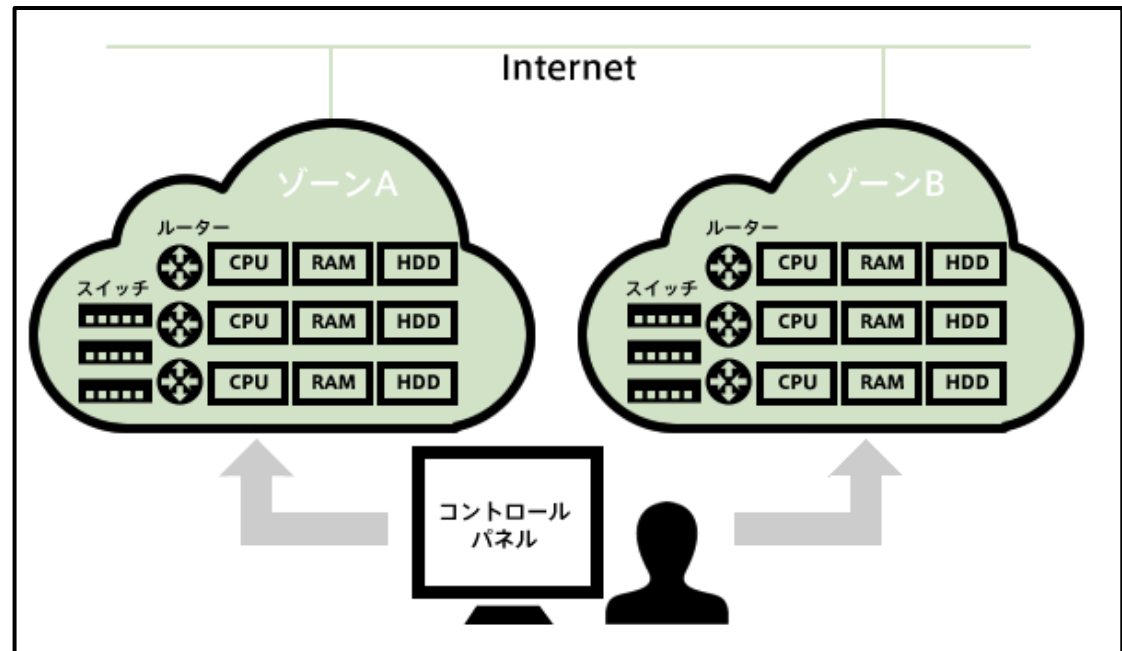


石狩第2ゾーン始めました  
(2013/10/8より)



# ゾーンとは？

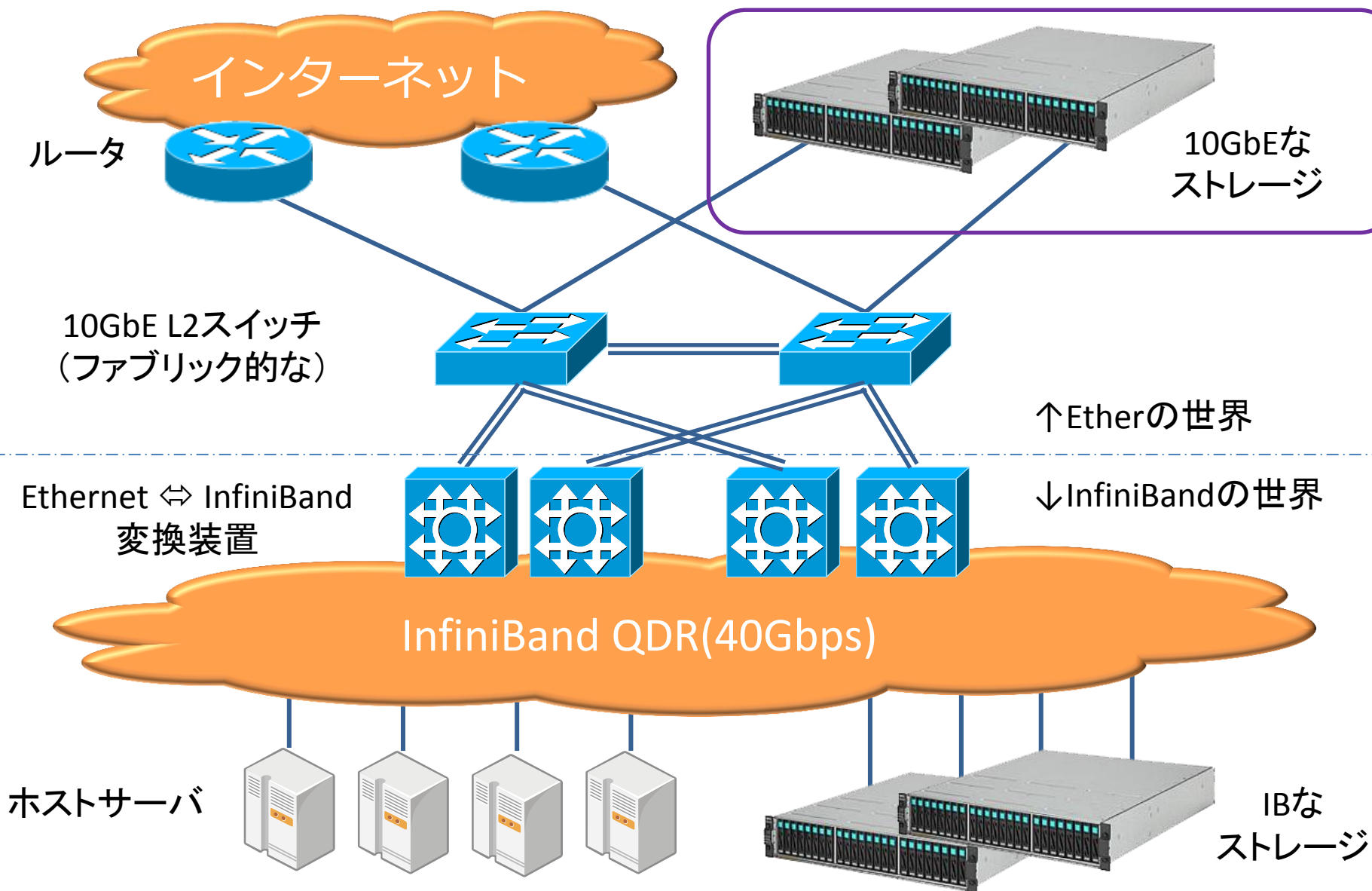
- AWSさんのアベイラビリティゾーン的な
- システムを完全に分離、障害が波及しない
- APIサーバ(クラウドコントローラ)とコンパネサーバも分離
- GSLB等と組み合わせた冗長システムの構築が可能



折角なので...

第2ゾーンは  
ネットワーク構成を一新しました

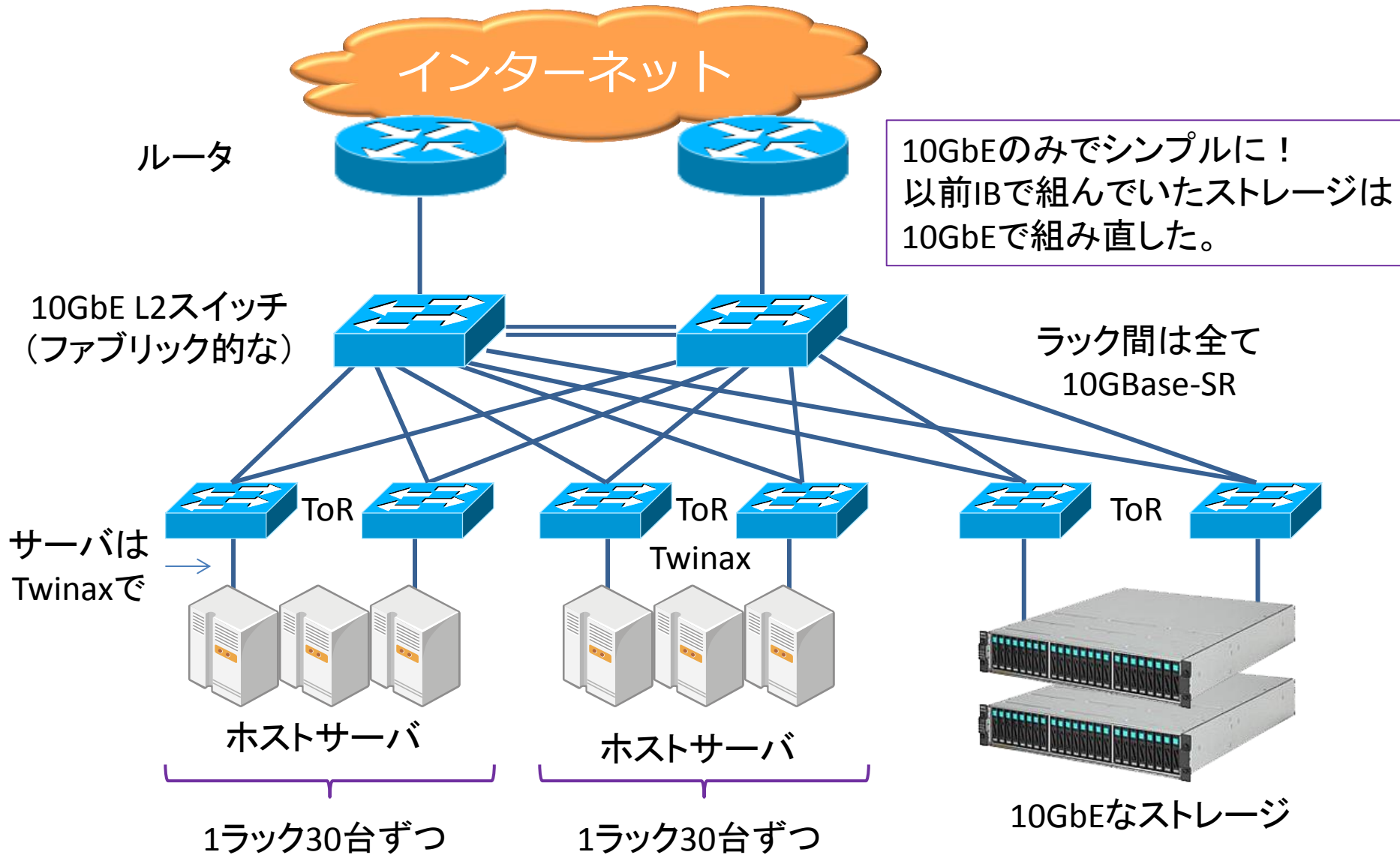
# 第1ゾーンの構成は・・・



## 第2ゾーンでは・・・

- 10年使えるアーキテクチャにしたい
- InfiniBandやめました
- 普通の10GbEベースのネットワーク
- VMの通信とストレージ通信を1面のネットワークに収容
- SDNはまだ入ってません。。。

# こんな感じ



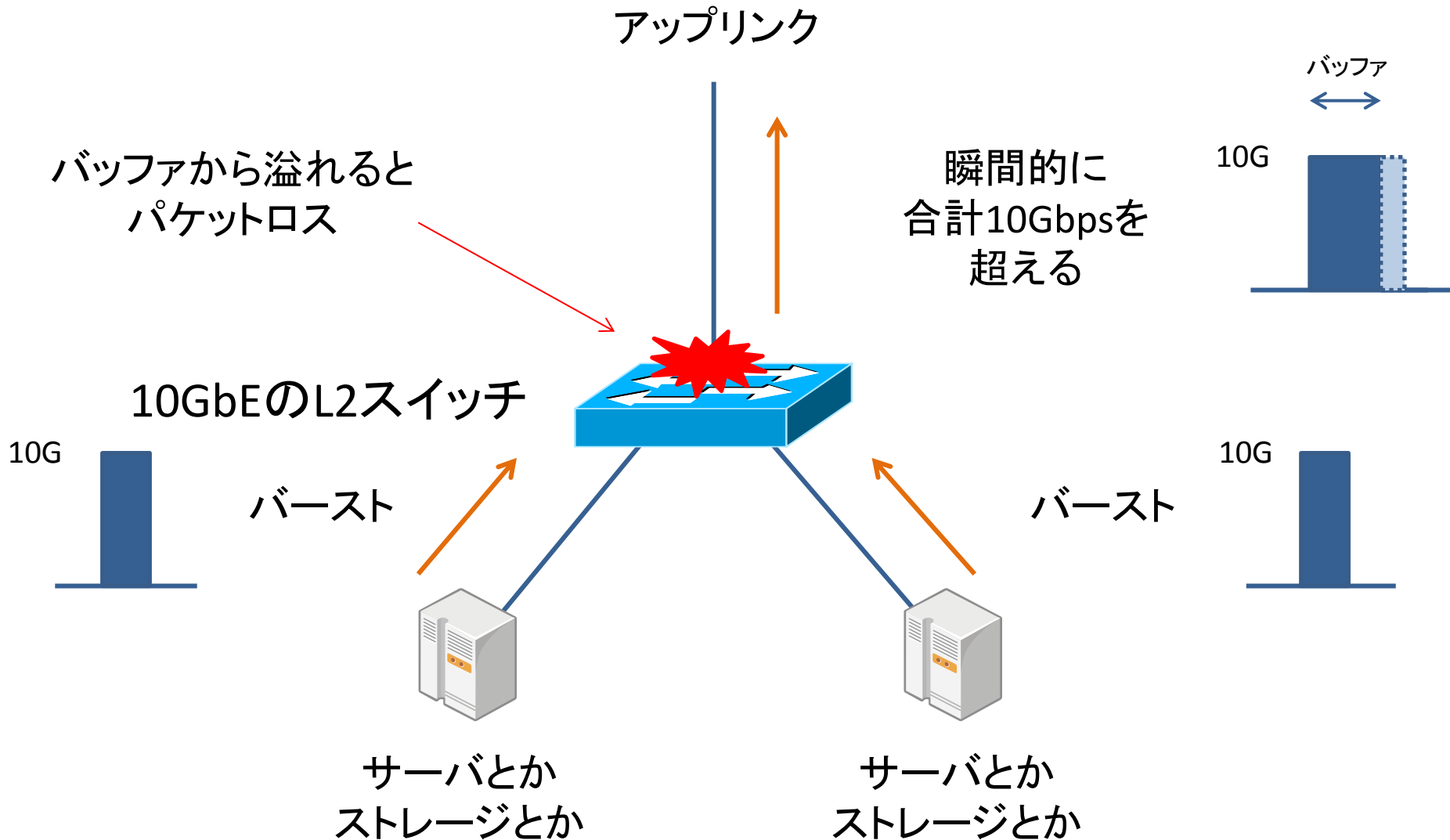
# IBをやめて10GbEにした理由

- 10GbEインターフェイスのストレージが増えた
- Ether over InfiniBandの部分がボトルネックに
- 10GbE(NIC、スイッチ)も安くなってきた
- 帯域は減るが、10G × Nでも足りる
- IBドライバのインストールが大変 😞
  - いろんなドライバが入っていると障害解析も大変 😞

# IBの良かったところ

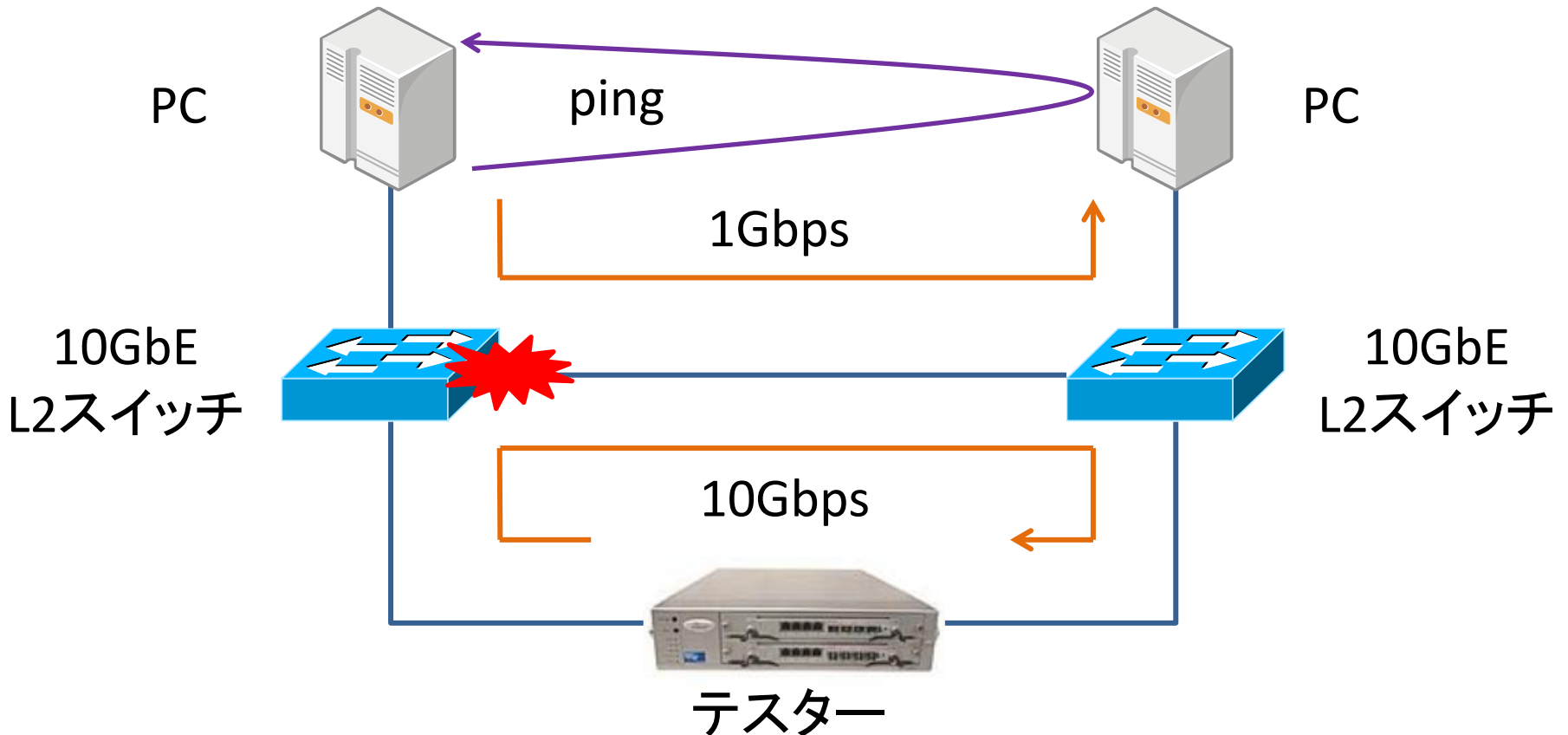
- 「IB自体はすごく良いメディアだと思います」
- ループフリー
  - サブネットマネージャがノード間の通信経路を制御してくれる
  - IBスイッチはどんなトポロジでつないでも良い
  - ECMP、Shortest Path普通に動く
- マネジメントフリー
  - IBスイッチは個別にマネジメントしていない
  - サブネットマネージャのトポロジ情報で監視可能
- バーストラフィックでもパケットロスしない
  - End to Endのフロー制御がある

# バーストによるパケロスが心配





# 参考：パケットバッファの測定



ポートを溢れさせ、何 $\mu$ secのパケットを貯められるかをpingのレイテンシで測定し、バッファサイズを逆算する

# 参考：パケットバッファの測定

- 無輻輳時の遅延

105 packets transmitted, 105 received, 0% packet loss, time 5213ms  
rtt min/avg/max/mdev = 0.031/0.035/0.176/0.016 ms

- 輻輳時の遅延

224 packets transmitted, 216 received, 3% packet loss, time 11369ms  
rtt min/avg/max/mdev = 4.369/4.374/4.404/0.039 ms

- バッファサイズ

$10 \times 1000^3 / 8 \times (0.004374 - 0.000035) = 5,423,750$  (約5.4MB)

※ 検証中に試した、とあるスイッチ

- 輻輳時の遅延

148 packets transmitted, 148 received, 0% packet loss, time 7372ms  
rtt min/avg/max/mdev = 0.078/0.084/0.231/0.015 ms

- バッファサイズ

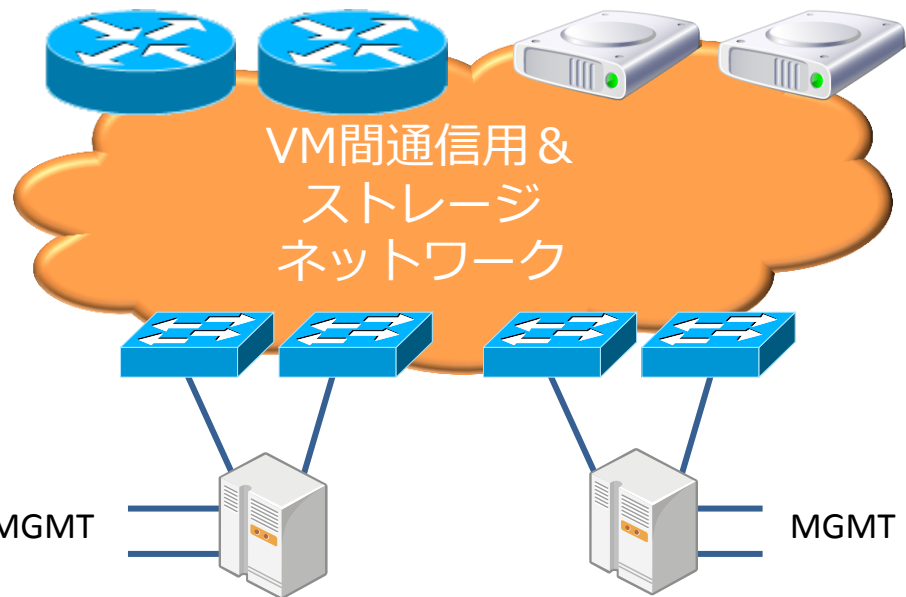
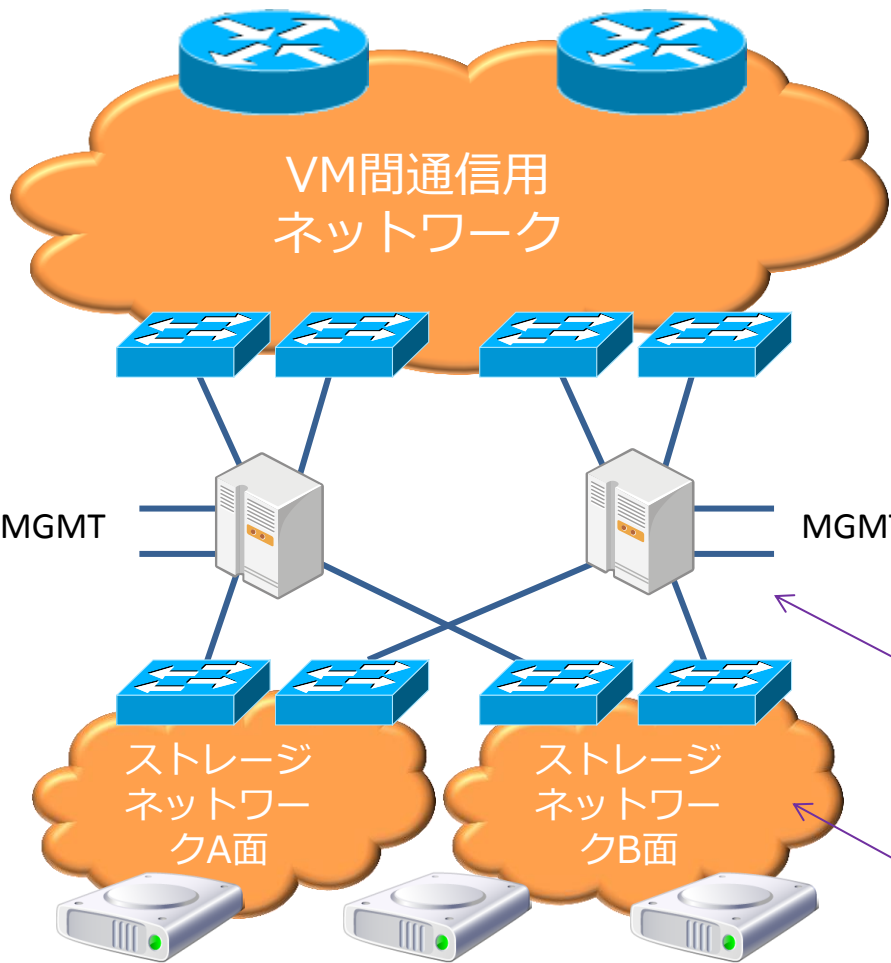
$10 \times 1000^3 / 8 \times (0.000084 - 0.000035) = 61,250$  (約61KB)

※ 実はカタログスペックで64KB

# ホストの配線数を増やしたくない

普通に安全に組むなら...

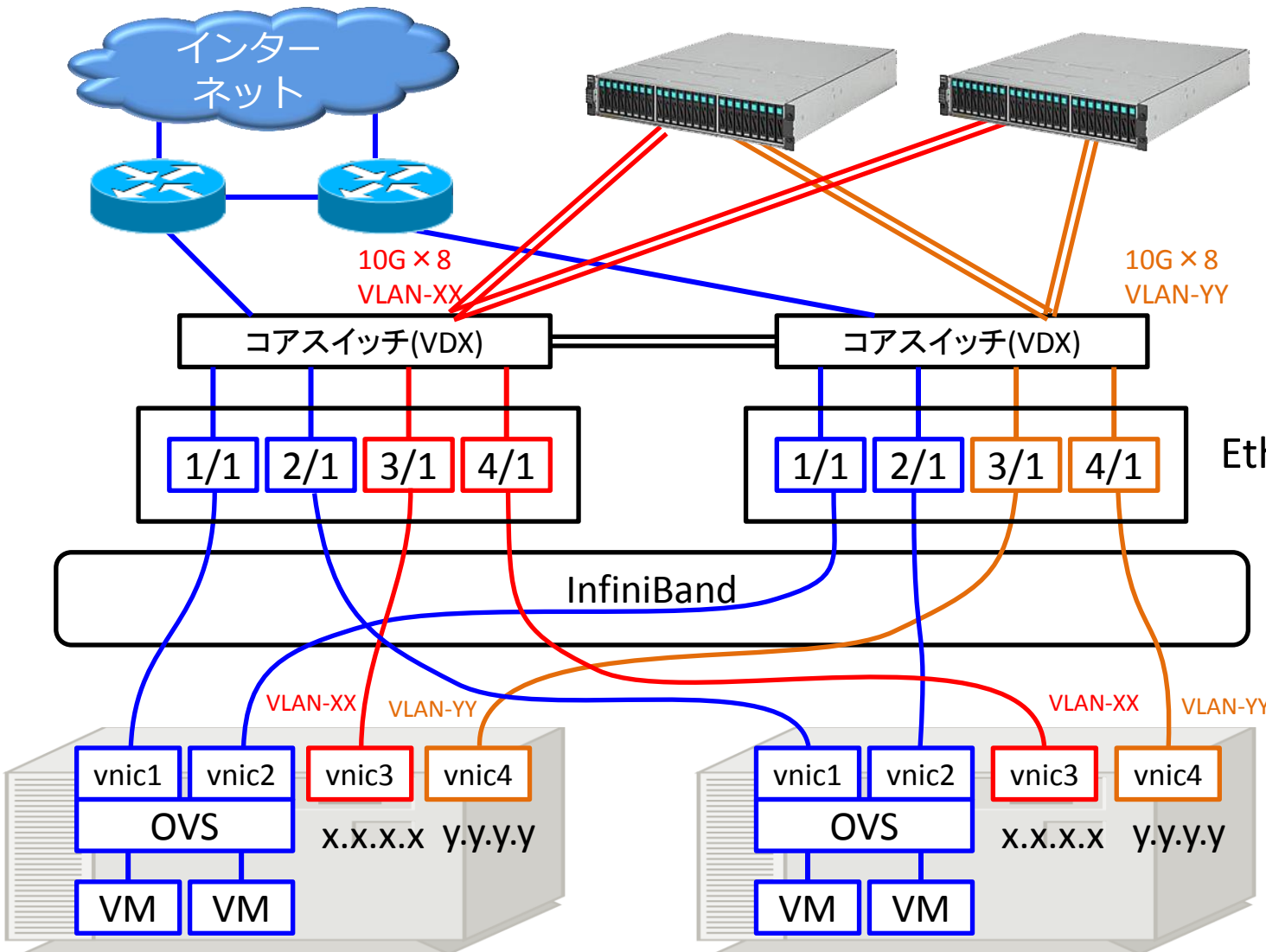
できれば一緒にしたい



配線もスイッチも  
たくさん必要

文化の違う  
ネットワーク

# InfiniBandの時は



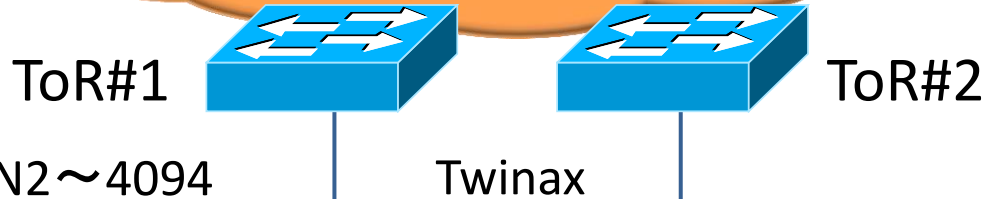
物理ポートは  
分離する

Ether over InfiniBand

ストレージ用の  
vnicを別に張る。  
OVSを経由しない

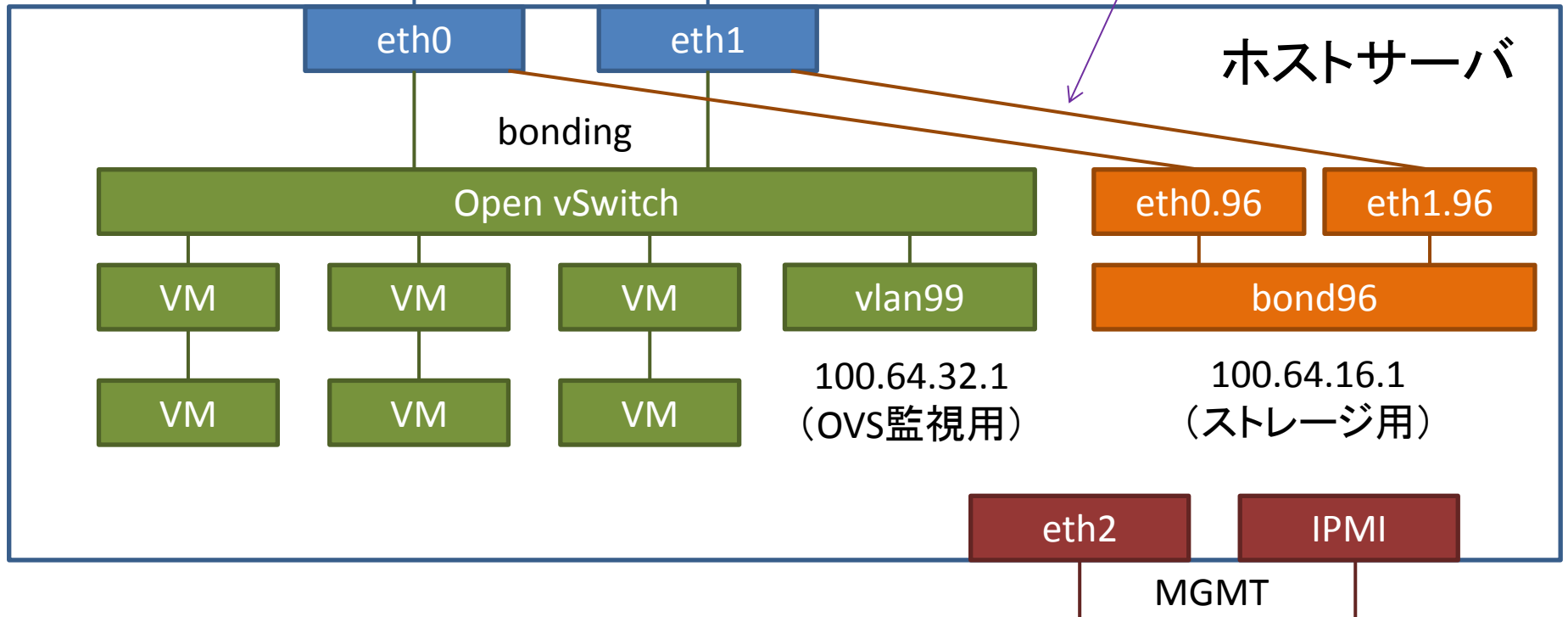
# 10GbEでどうしたか

文化の違いは覚悟を決める



VLAN2~4094

ストレージ通信をOVSから分離することで安定化。  
VLANタグの付与、除去はNICへHWオフロード。



# 配線数 Before After

- 第1ゾーン (InfiniBand)
  - 電源 × 2
  - InfiniBand QSFPケーブル × 2
  - IPMI
  - マネジメント
- 第2ゾーン (10GbE)
  - 電源 × 2
  - 10GbE Twinaxケーブル × 2
  - IPMI
  - マネジメント

サーバあたり6本  
ラックあたり180本

変わらず！

サーバあたり6本  
ラックあたり180本

# ブリッジ接続機能（近日提供予定）



第1ゾーン

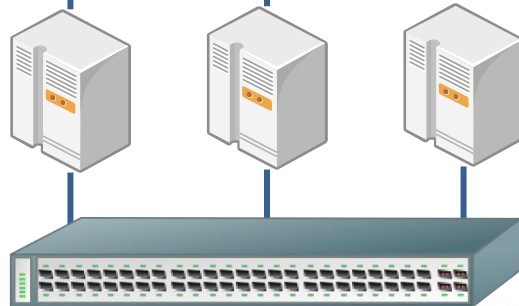
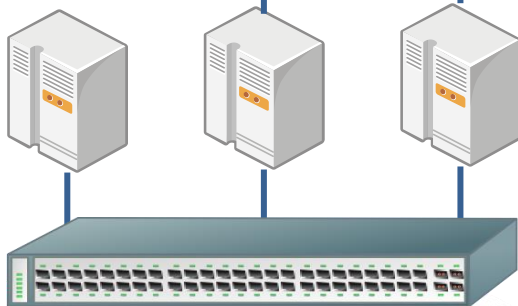
第2ゾーン

お客様  
仮想サーバ

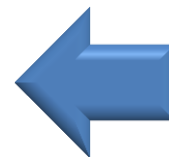
お客様  
仮想サーバ

ローカル  
スイッチ

ローカル  
スイッチ



ブリッジ接続



ゾーンを超えて  
ローカル接続

# ハイブリッド接続サービスとの違い？

ハイブリッド接続は2012/10/3より提供開始



各サービスの特長を生かしたシステム構築が可能

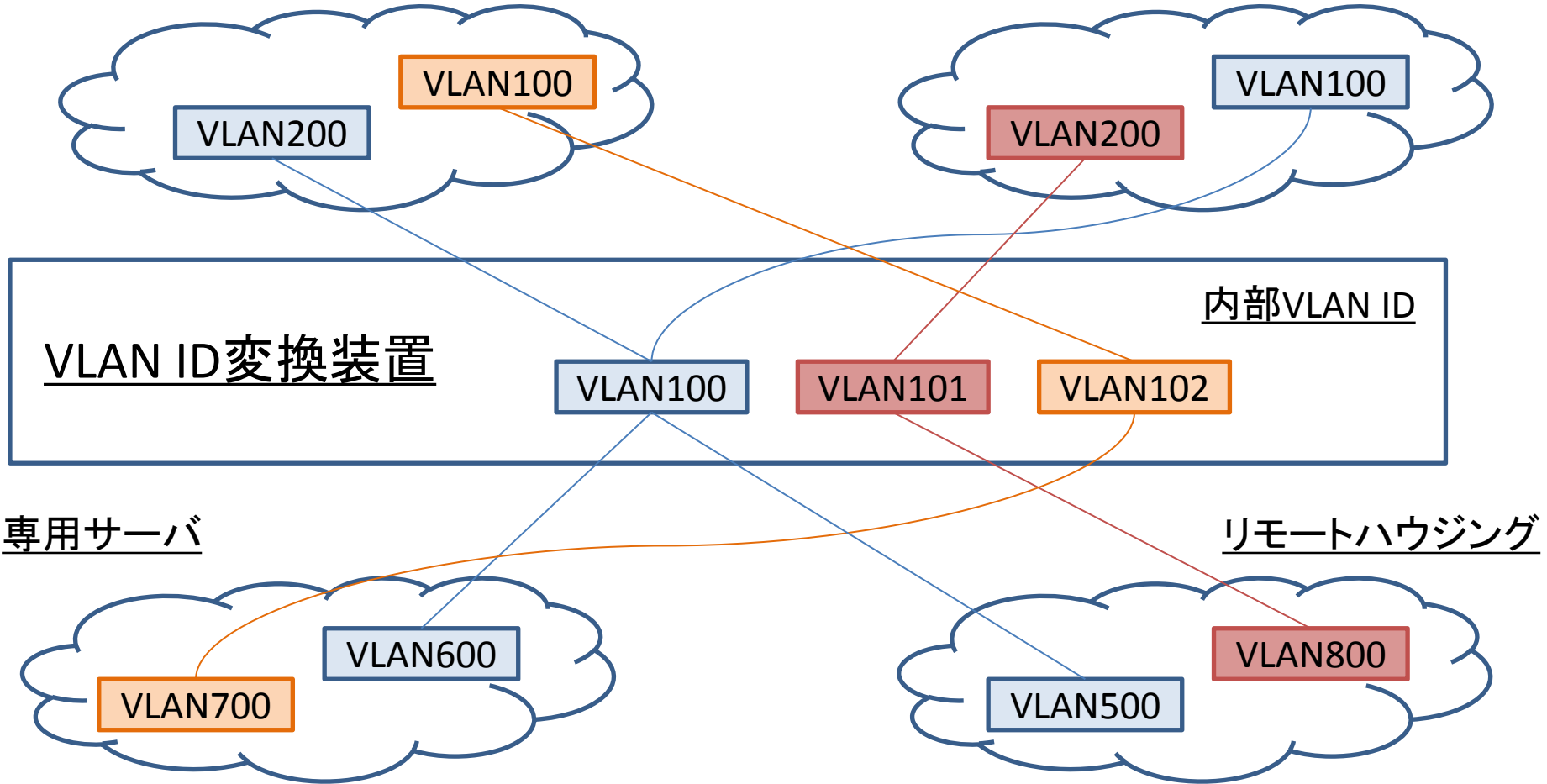




# 仕組み

クラウド第1ゾーン

クラウド第2ゾーン



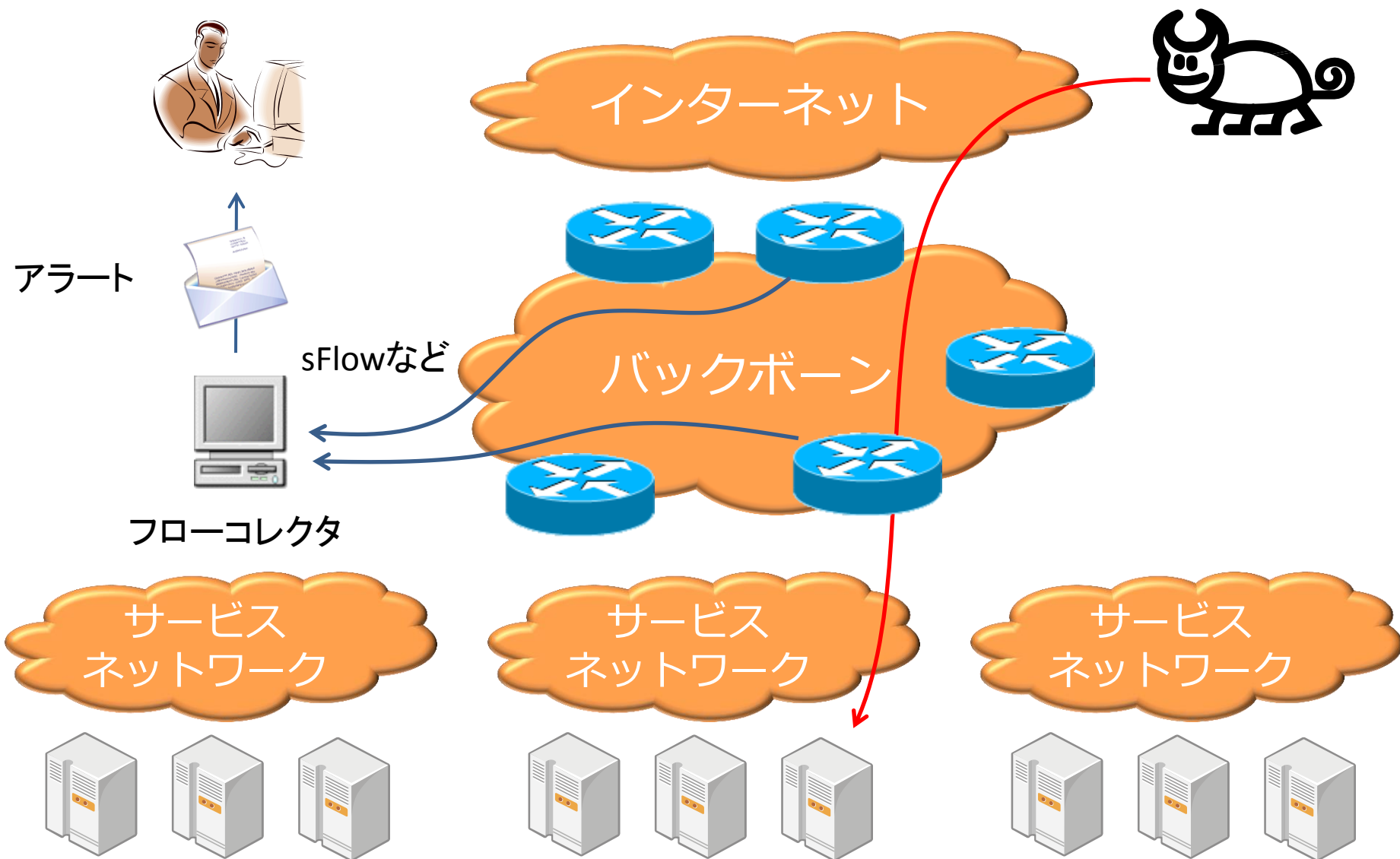
ハイブリッド接続の仕組みをそのまま利用

# DoSアタック対策の話

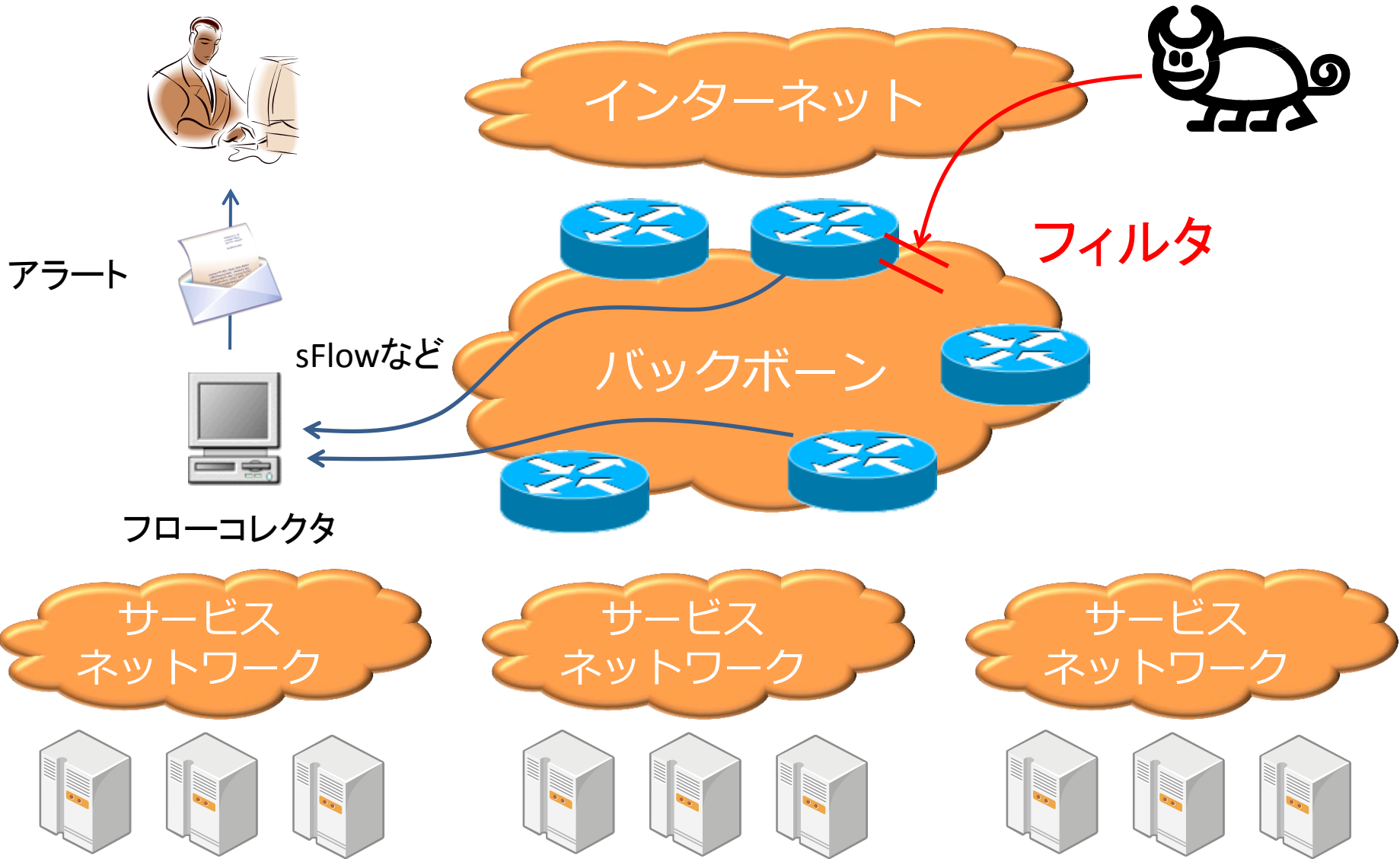
# DoSアタック対策

- バックボーン全体で見ると、DoSの発生頻度は日常茶飯事
- ただ、何をDoSとみなすかは難しい
- 弊社では帯域とppsの閾値で、他ユーザに影響が発生する場合はバックボーンでフィルタ

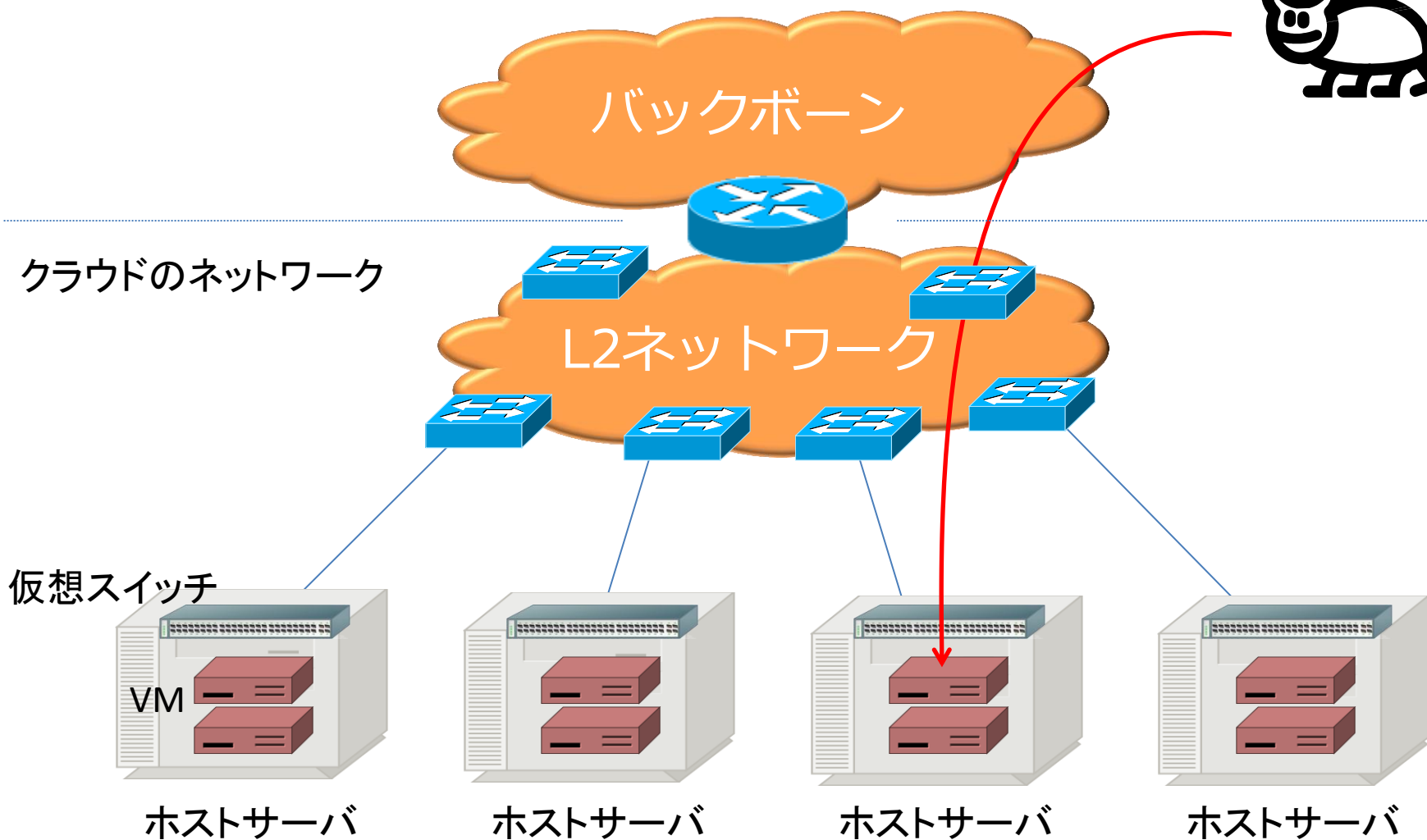
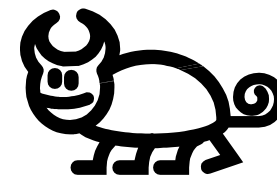
# バックボーンでの検出とフィルタ



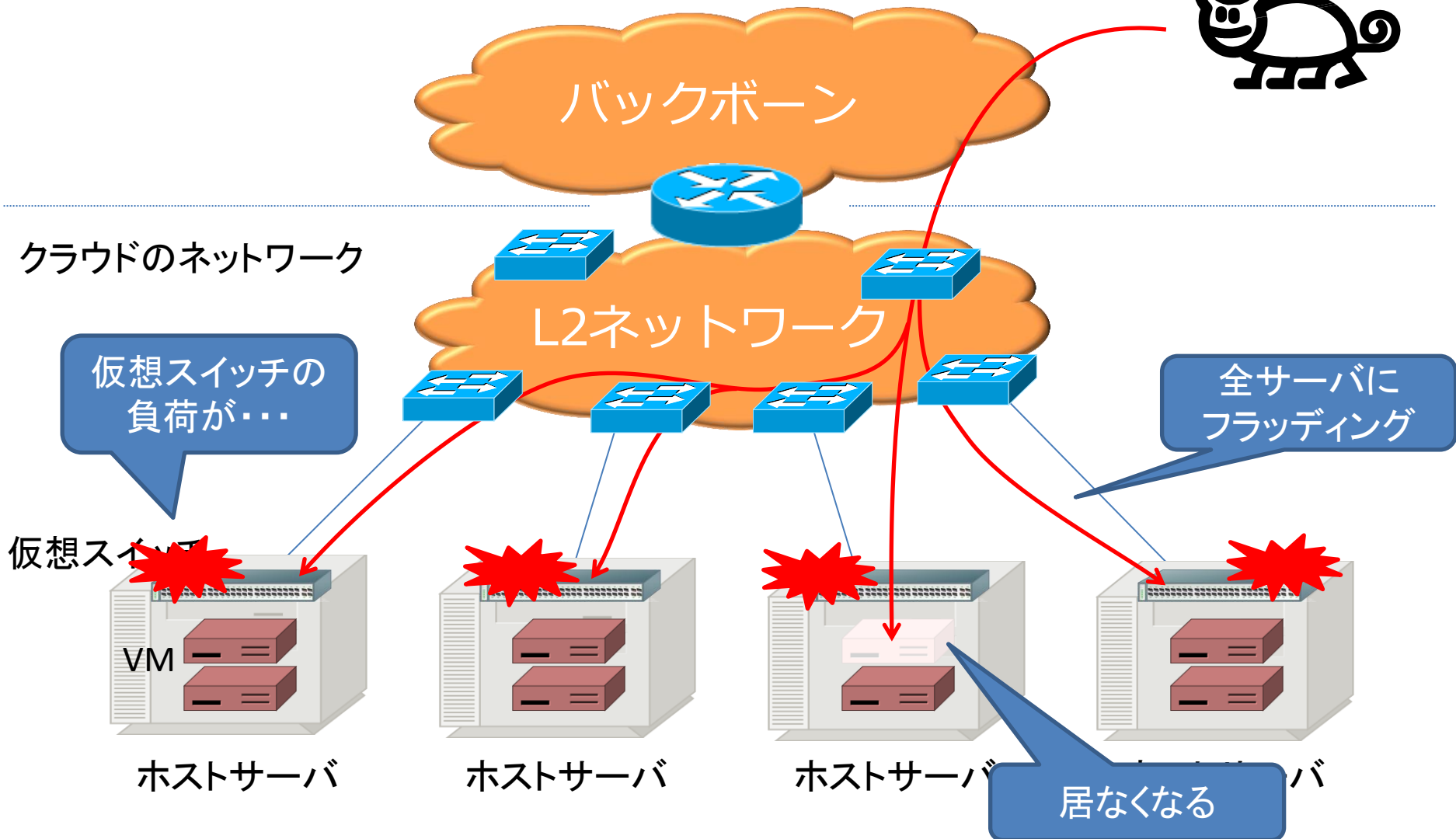
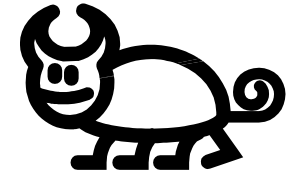
# バックボーンでの検出とフィルタ



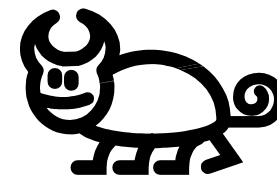
# ある日発生した事象



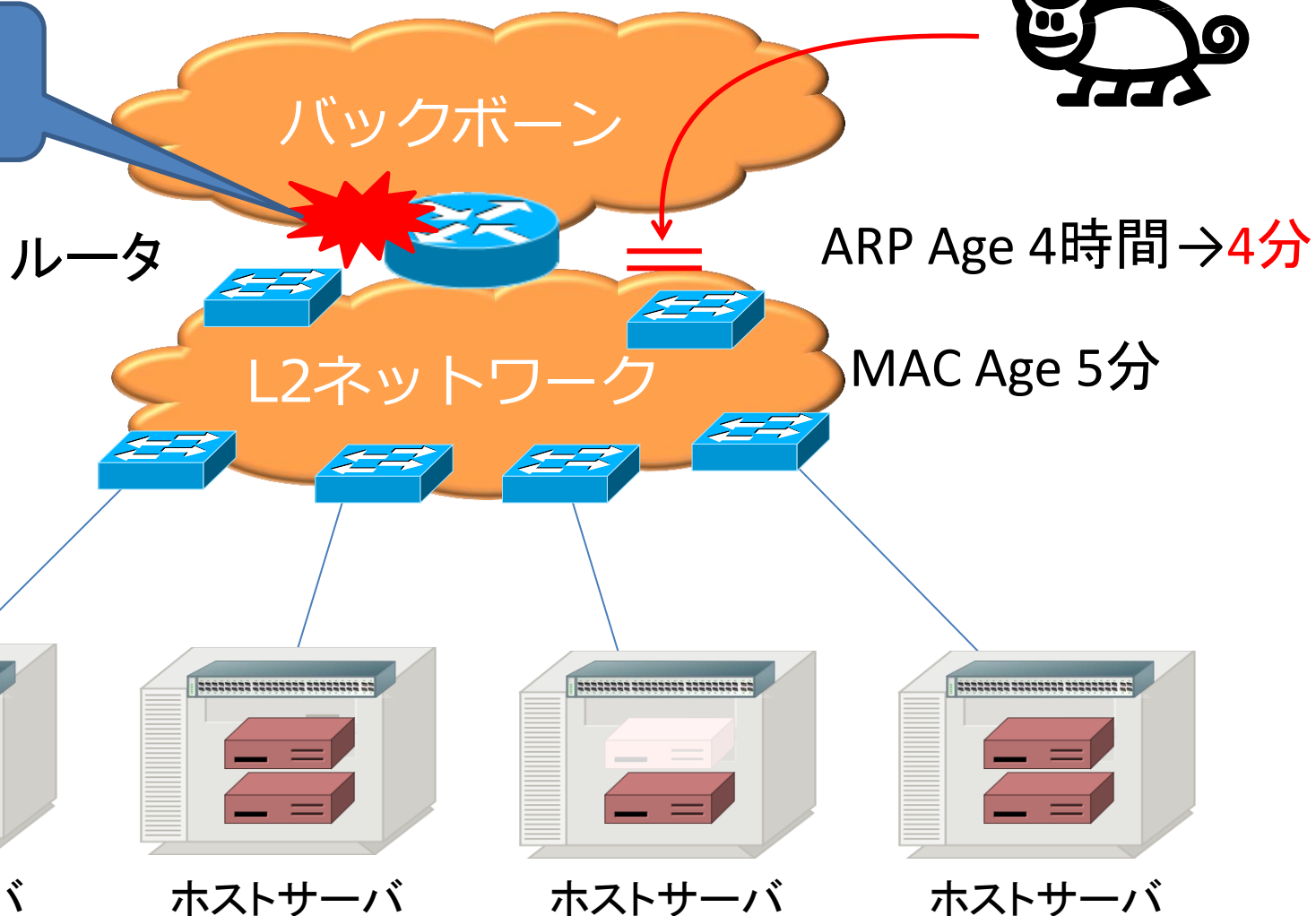
# ある日発生した事象



# ルータのARPタイムを短縮してみた

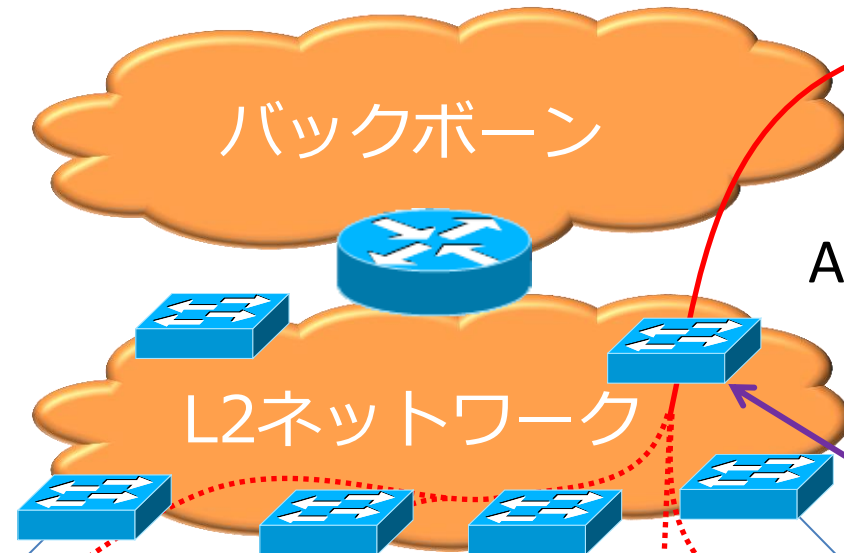
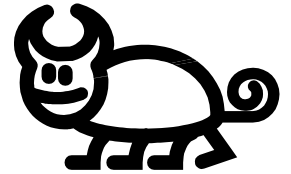


ルータの  
CPU負荷が...





# 結局どうしたか？

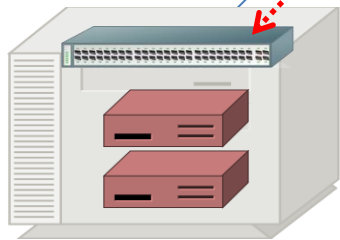


ARP Age 4分 → 4時間

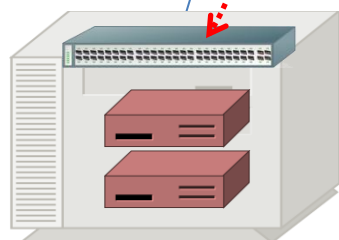
MAC Age 5分

フラッディングするけど、問題ないレベル

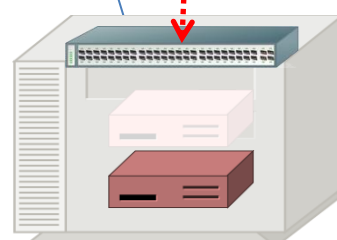
unknown unicastのフラッディング制限



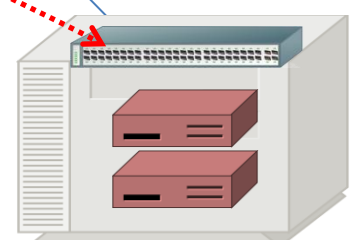
ホストサーバ



ホストサーバ



ホストサーバ



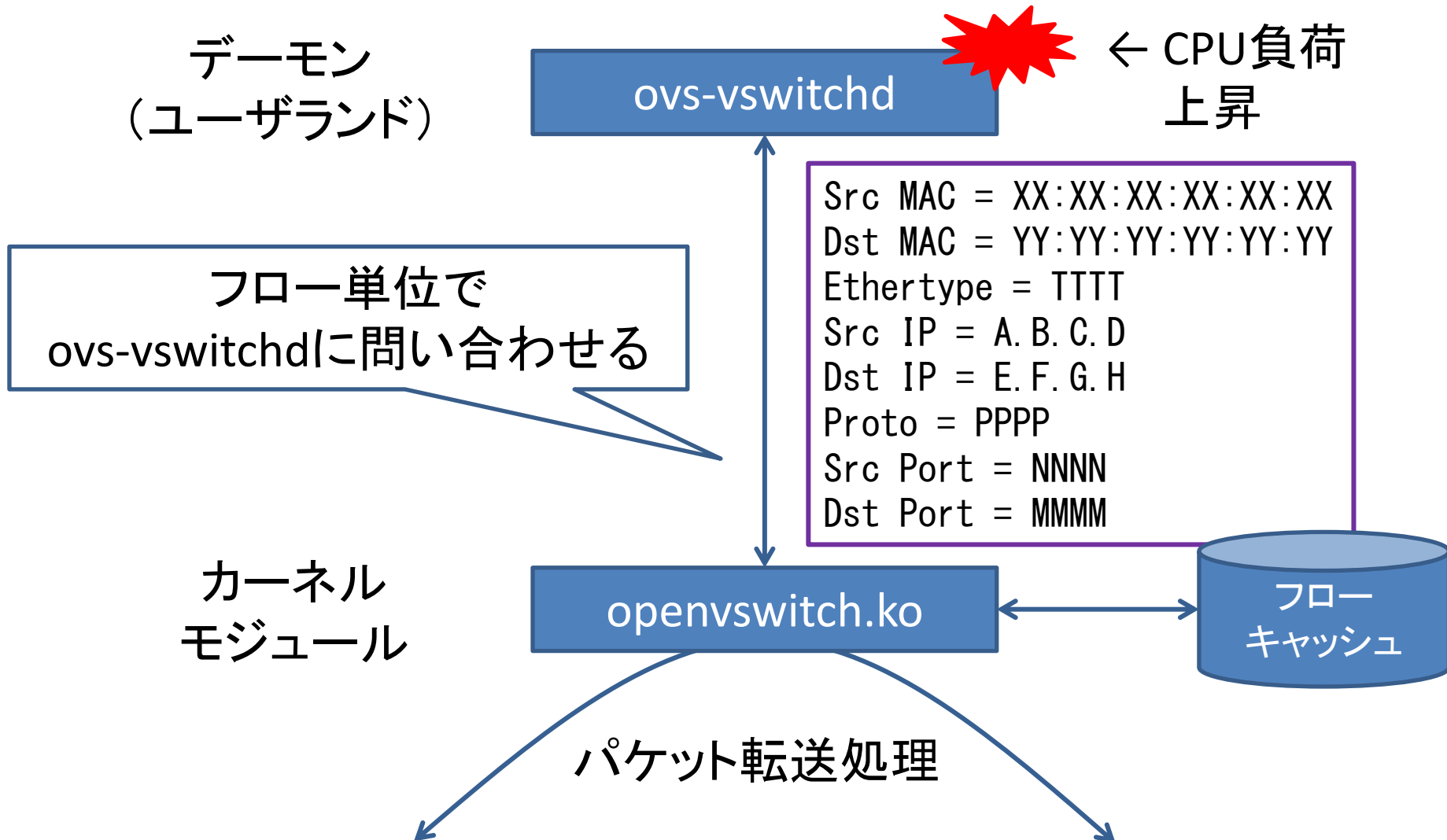
ホストサーバ

# Open vSwitchはDoSに弱い

- Open vSwitchは、フローベースのアーキテクチャ
- DoSアタックなど、大量のフローが発生すると、転送性能が極端に低下する
- フローキャッシュを見る方法

```
% ovs-dpctl dump-flows br0
in_port(2), eth(src=52:54:00:15:3e:a3, dst=52:54:00:b3:d4:57), eth_type(
0x0800), ipv4(src=220.205.27.17, dst=61.12.213.224, proto=17, tos=0), udp(
src=49766, dst=19968), packets:0, bytes:0, used:never, actions:1
in_port(2), eth(src=52:54:00:15:3e:a3, dst=52:54:00:b3:d4:57), eth_type(
0x0800), ipv4(src=200.254.249.181, dst=14.232.213.232, proto=17, tos=0), u
dp(src=54759, dst=10904), packets:0, bytes:0, used:never, actions:1
```

# Open vSwitchのアーキテクチャ



# これまでの対策

- フロー数の監視  
閾値を超えたら通知して対処
- 新しいバージョンを使う  
フローのセットアップ性能が向上  
これまで使ってきたバージョン: 1.1→1.2→1.4
- パラメータの調整

```
% ovs-vsctl set bridge br0 ¥  
other_config=flow-eviction-threshold=120000
```

※ ガベコレ処理開始の閾値を上げる

# Open vSwitch 1.11.0を入れてみた

- 2013/8/28にリリースされた
- 第2ゾーンのホストサーバに適用
- カーネルモジュールのフローキャッシュが、  
ワイルドカードマッチに対応！！

<http://openvswitch.org/pipermail/announce/2013-August/000054.html> より、

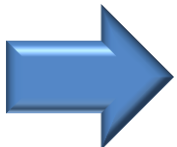
Support for megafloWS, which allows wildcarding in the kernel (and any dpif implementation that supports wildcards). Depending on the flow table and switch configuration, flow set up rates are close to the Linux bridge.

# 効果の程は・・・？

```
host % ovs-ofctl add-flow br0 priority=39990,ip,in_port=4,nw_src=1.0.1.0/24,actions=drop
```

```
guest % mz -c0 -t ip -A 1.0.0.0/23 -B 10.0.0.0/24 -t udp "sp=1,dp=1-10"
```

```
host % ovs-dpctl dump-flows
skb_priority(0), in_port(5), skb_mark(0/0),
eth(src=52:54:00:07:63:77, dst=2e:ba:27:b5:de:d1),
eth_type(0x0800),
ipv4(src=1.0.0.1/255.255.255.0, dst=10.0.0.2/0.0.0.0,
proto=17/0, tos=0/0, ttl=255/0, frag=no/0xff),
udp(src=1/0, dst=3/0),
packets:31994986, bytes:1343789412, used:0.001s,
actions:4
```



マスクが表示されるようになった。

don't care bitが異なるフローは、1エントリに集約される。

# でも...

- フローエントリが参照するフィールドの和集合(or)でマスクが生成される。
- 参照するフィールドが増えるほど、完全一致に近くなる。ワイルドカードが活かせない。。



フロー数の問題が完全に解決するわけではないが、ちょっとはマシかなというレベル

# まとめ(欲しいもの)

- HDDではIOが絶対的に不足  
→ 既に小容量ディスクはSSDに移行済み
- 大容量、高性能、高可用なフラッシュストレージ希望
- 超高速にARPのリフレッシュができるルータ
- フロー数が爆発しないOpen vSwitch
- 気持ちよく使えるSDN  
→ VLAN数MAC数を気にせず、心配事の少ない  
まるっと全部が繋がれるネットワークインフラを
- End to Endでフロー制御できるEthernet製品群  
(802.1Qauとか)