IDC Frontier
未来をささえる、Your Innovative Partner

ARPトラブルあるある

株式会社IDCフロンティア
技術開発本部 UX開発部
アーキテクトグループ
井上 一清

> 世の中の様々なネットワーク技術

OSPF

Automation

Fabric

BGP

SDN

Ansible

MPLS

EVPN

Fabric

VPLS

Rest API

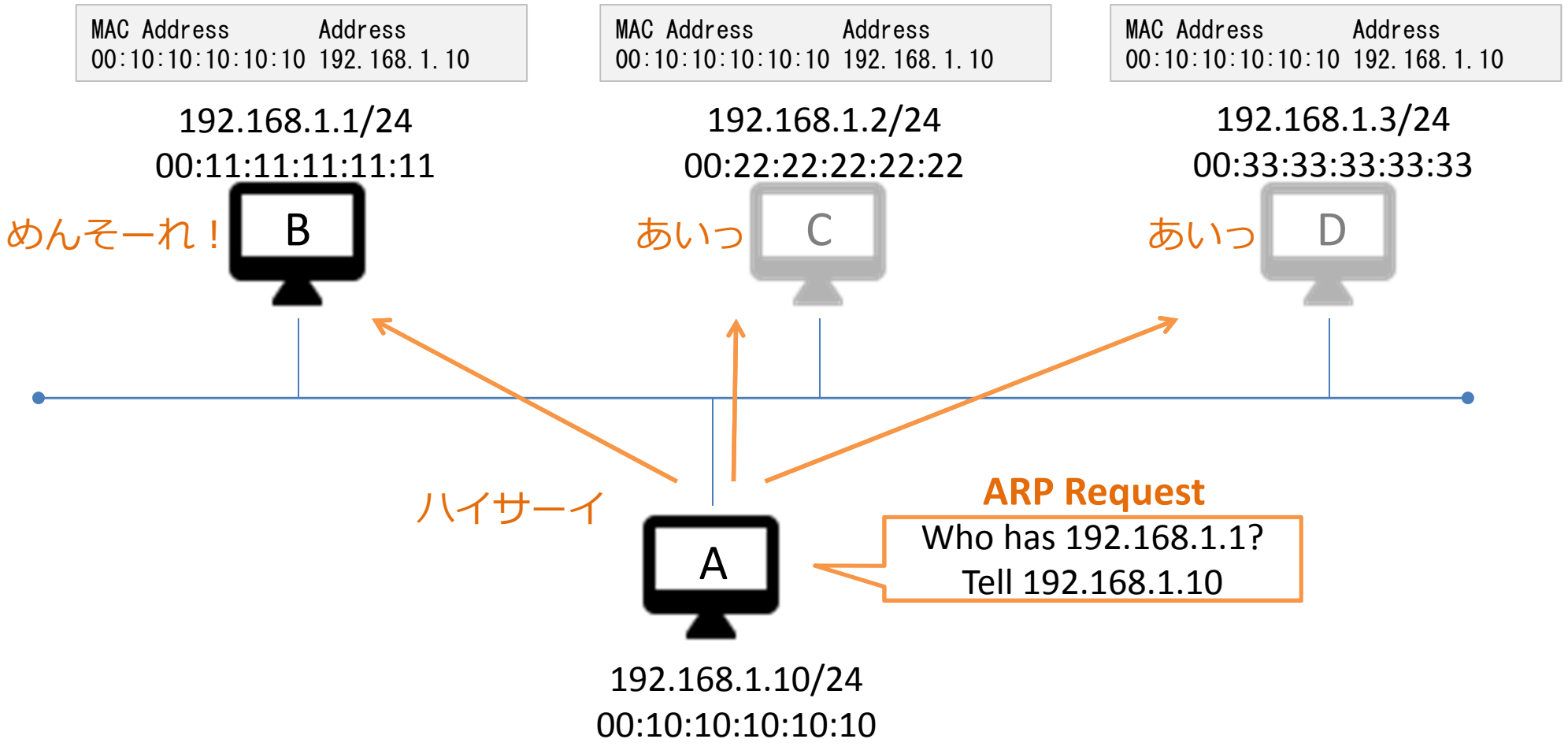
ARP

ARP

今回の話は大体こちらに載ってます・・

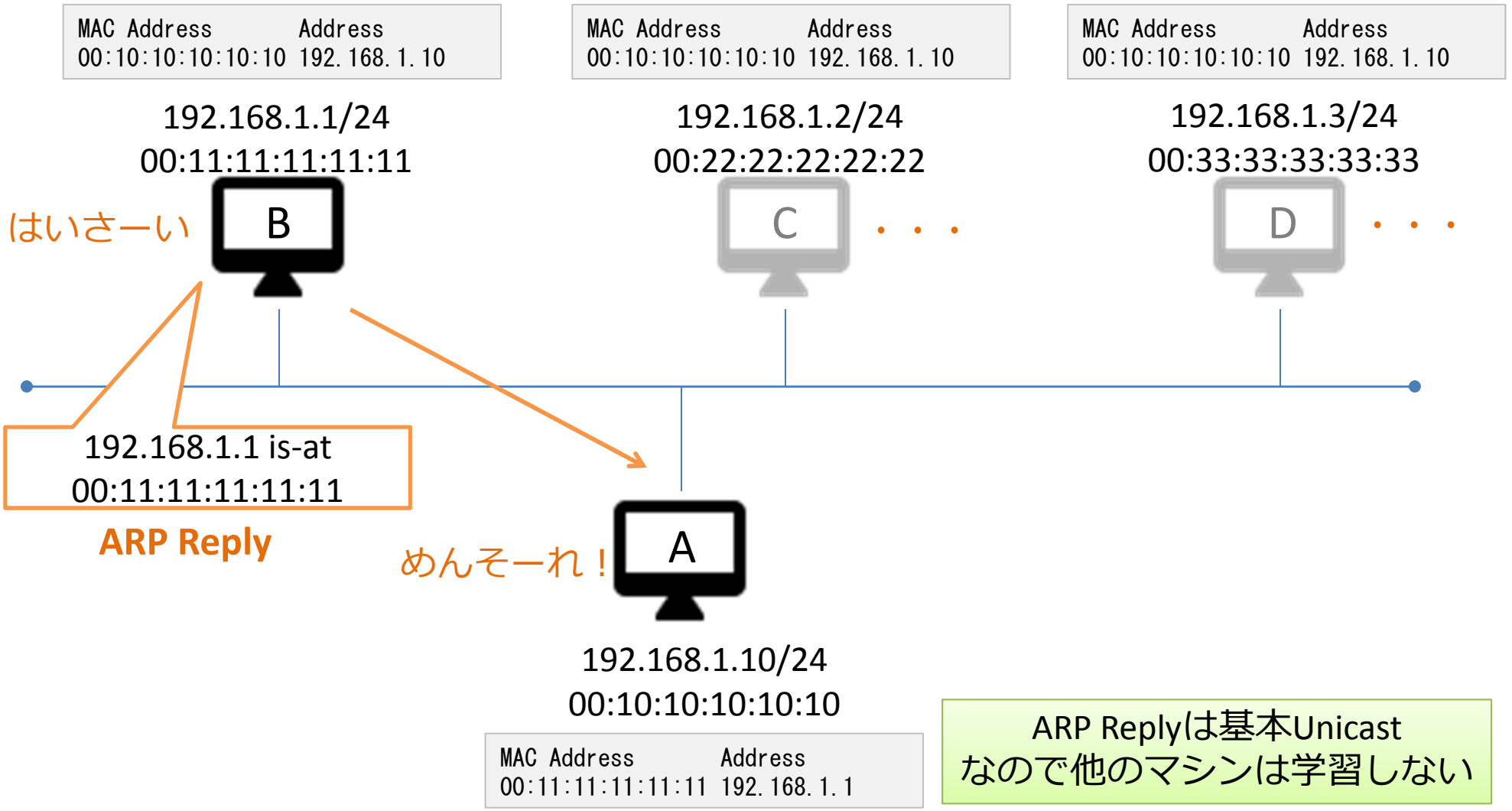
http://qiita.com/inoueissei/items/8d61e675c404ff2ef8d1

> ARPは一往復で両機器のARP tableが更新される



ARP RequestはBroadcastで送出され、その時点で
他のマシンもマシンAのARPテーブルを学習する

> ARPは一往復で両機器のARP tableが更新される

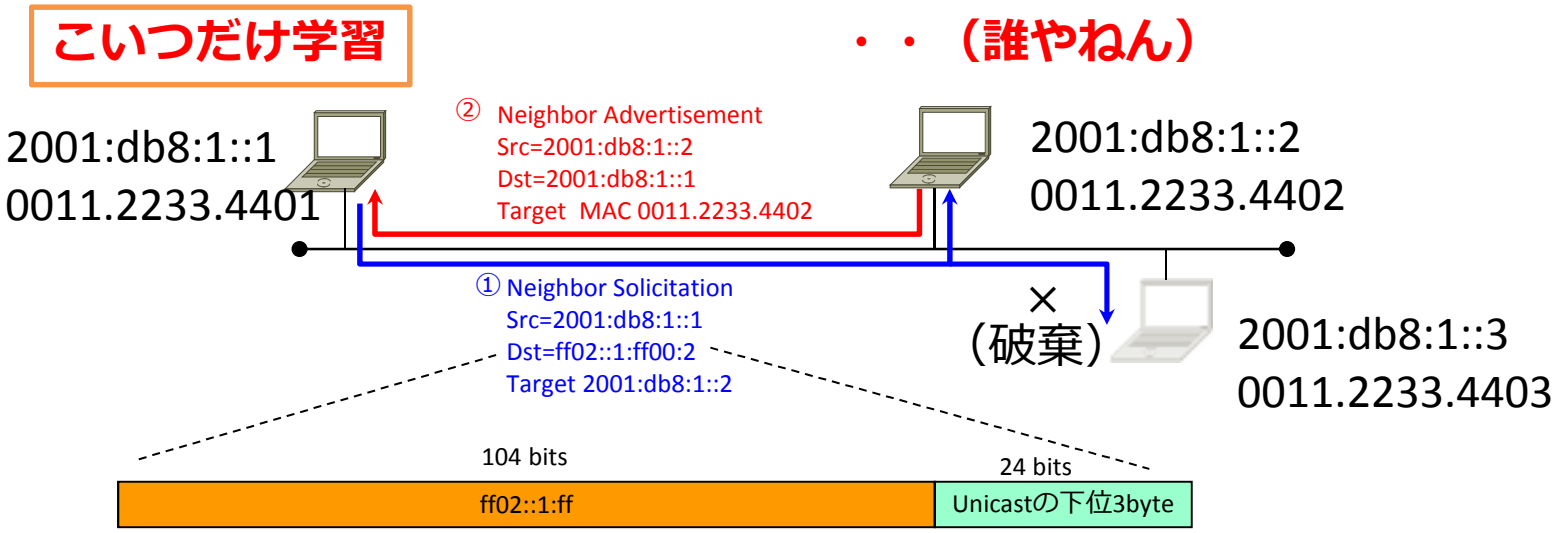


はぁーや
(へー、まぁそんな感じだよね)

でもNDPだと違う

> Neighbor Solicitation/Neighbor Advertisement の仕組

- ARPと異なり双方向で行われる必要がある
- 要請ノードマルチキャストアドレスを上手く利用
  （ff02::1:ff00:0000～ff02::1:FFff:ffff）



2001:db8:1::0000:0002 (2001:db8:1::2)

↓

ff02::1:ff00:0002 (ff02::1:ff00:2)

要請ノードマルチキャストアドレス

Multicast AddressとEthernet Addressの関係

ff02::1:ff00:0002 (ff02::1:ff00:2)

(Dst Ethernet Address) 33:33:ff:00:00:02

※”33:33”にMulticastの下位4byteを連結

- IPv6だと双方でNS/NAをやり取りする必要がある
  - /64とかのアドレス空間で大量のデバイスが接続される可能性があるため
  - 実際通信しない機器のMACまで覚えてられない

- 要請ノードマルチキャストアドレスのDst MACを見れば、破棄できる
  - 機器は自分の要請ノードマルチキャストアドレスを知っている。
  - 他の要請ノードマルチキャストアドレスは破棄

> ARPはCPUを食う

APP Requestのパケットキャプチャ

```
Frame 147: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0
Ethernet II, Src: 12:94:e1:38:48:d7 (12:94:e1:38:48:d7), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 12:94:e1:38:48:d7 (12:94:e1:38:48:d7)
    Sender IP address: 10.157.29.15 (10.157.29.15)
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 10.157.28.46 (10.157.28.46)
```

これを”CPUが”見てようやく
自分ではないと判断して破棄

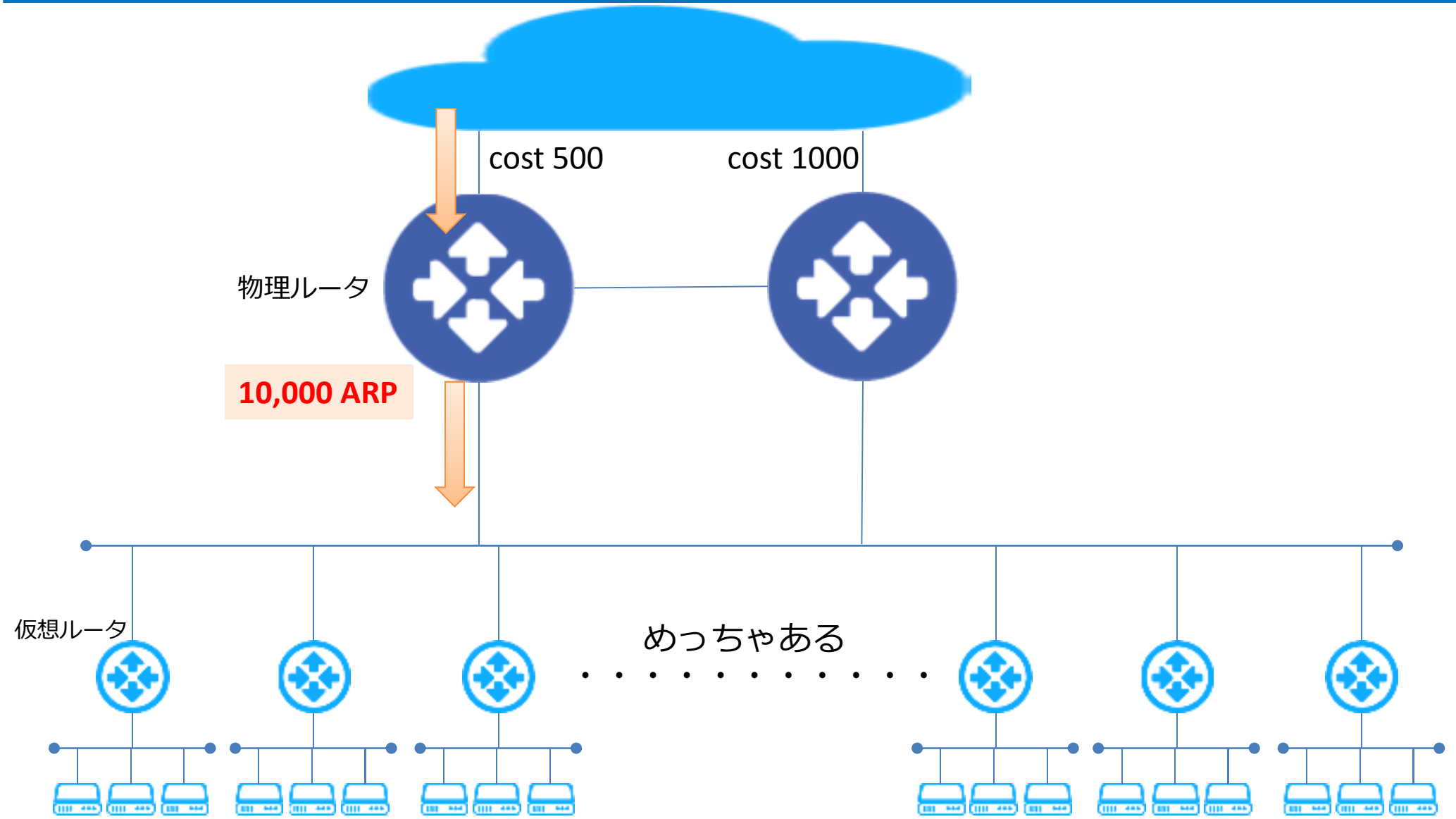
Broadcastだからこの段階では
（NICは）破棄できない

※あと、ARPはIPパケット（Ethernetタイプ番号0800）ではありません。

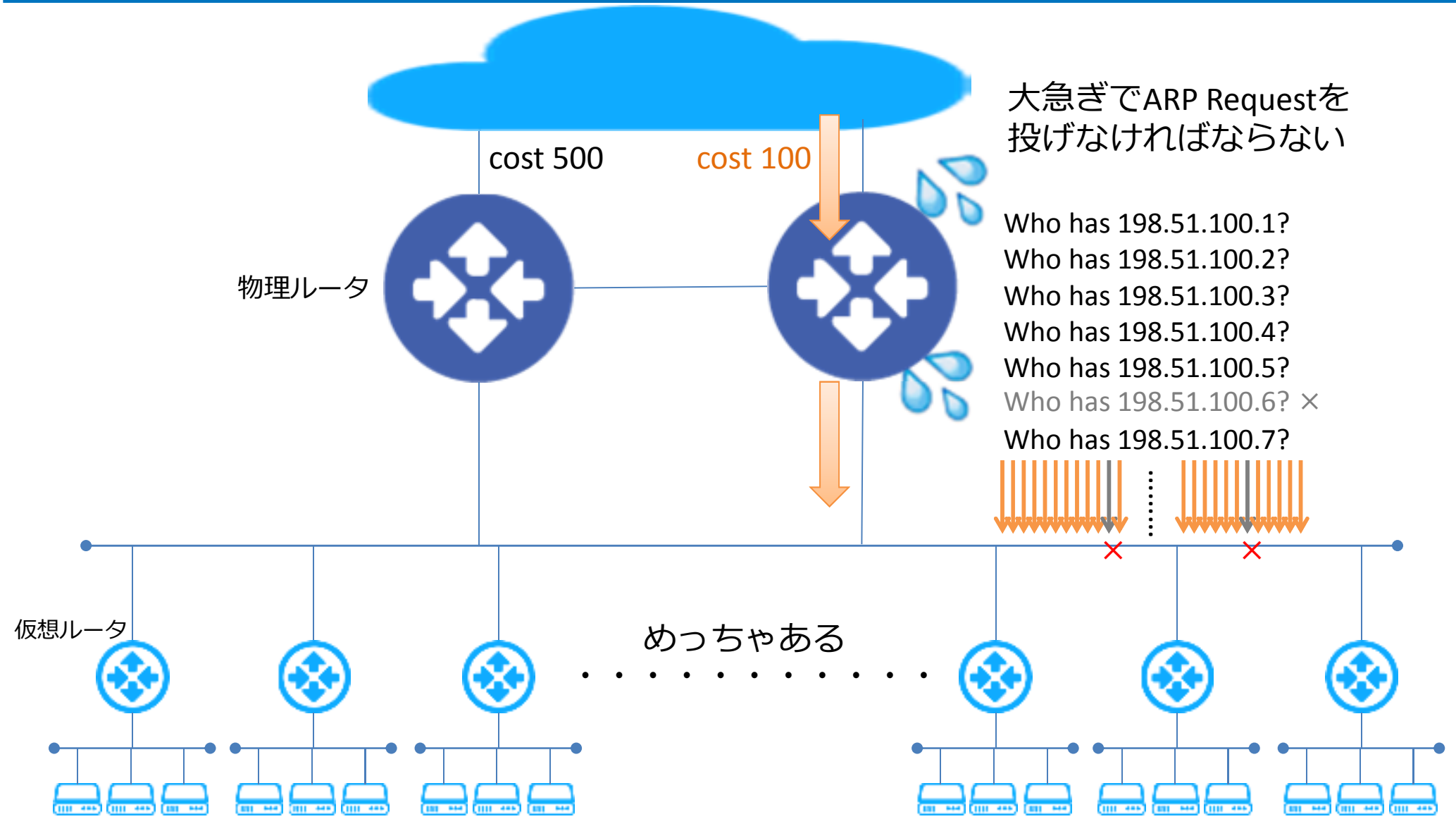
ARPのBroadcast Stormが発生した場合、
帯域が食われるだけでなく、マシンのCPUも100%に張りつく

トラフィック切替時のARP学習に注意しよう

> 例えば、クラウド/仮想環境で良くある構成

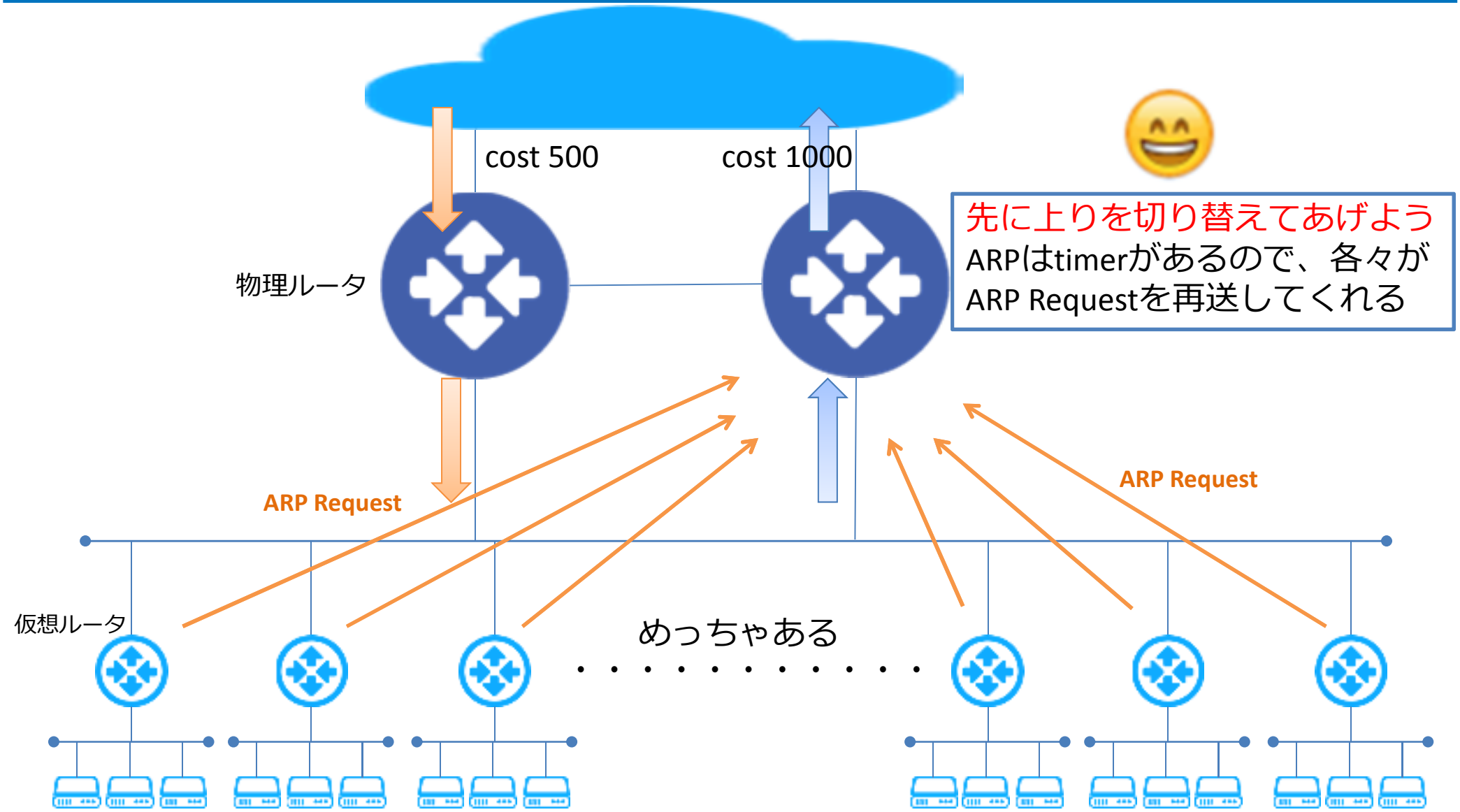


> 例えば、クラウド/仮想環境で良くある構成

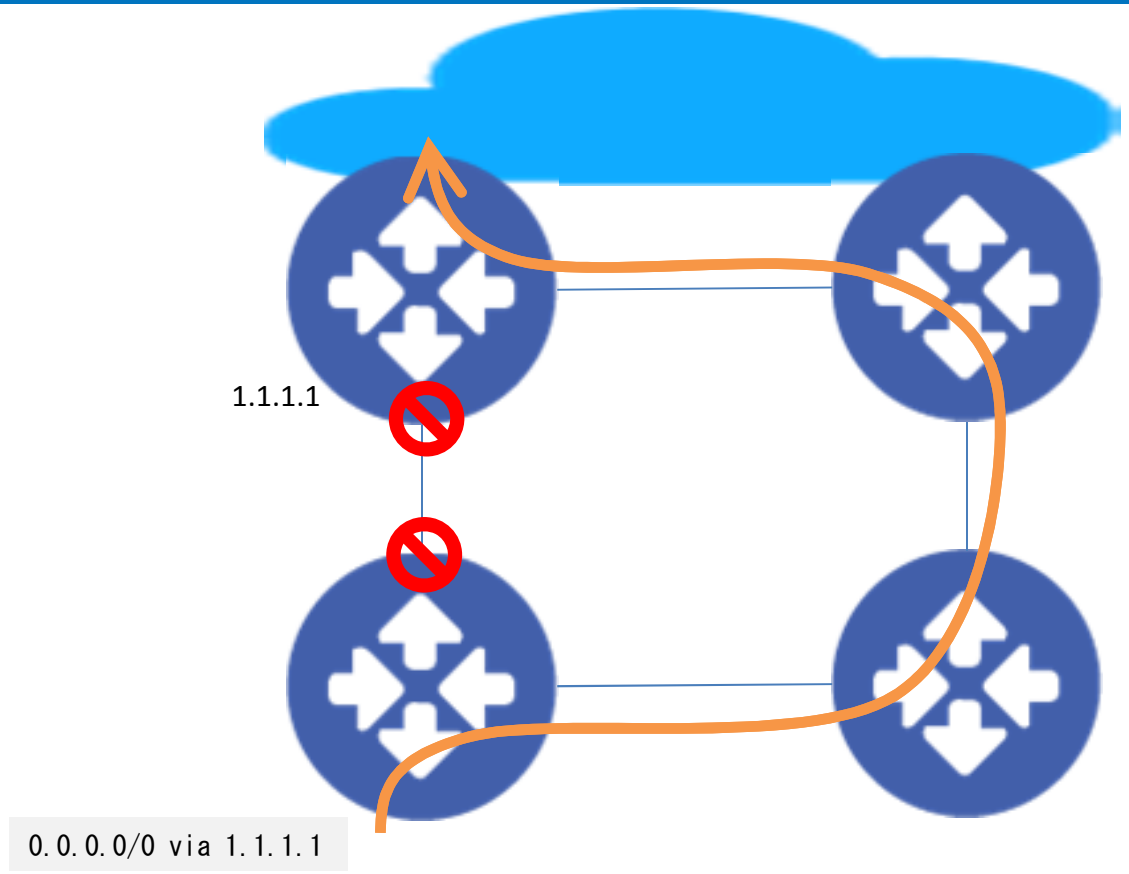


機器によるがこぼした場合は再送に20秒程度かかる

IDC Frontier

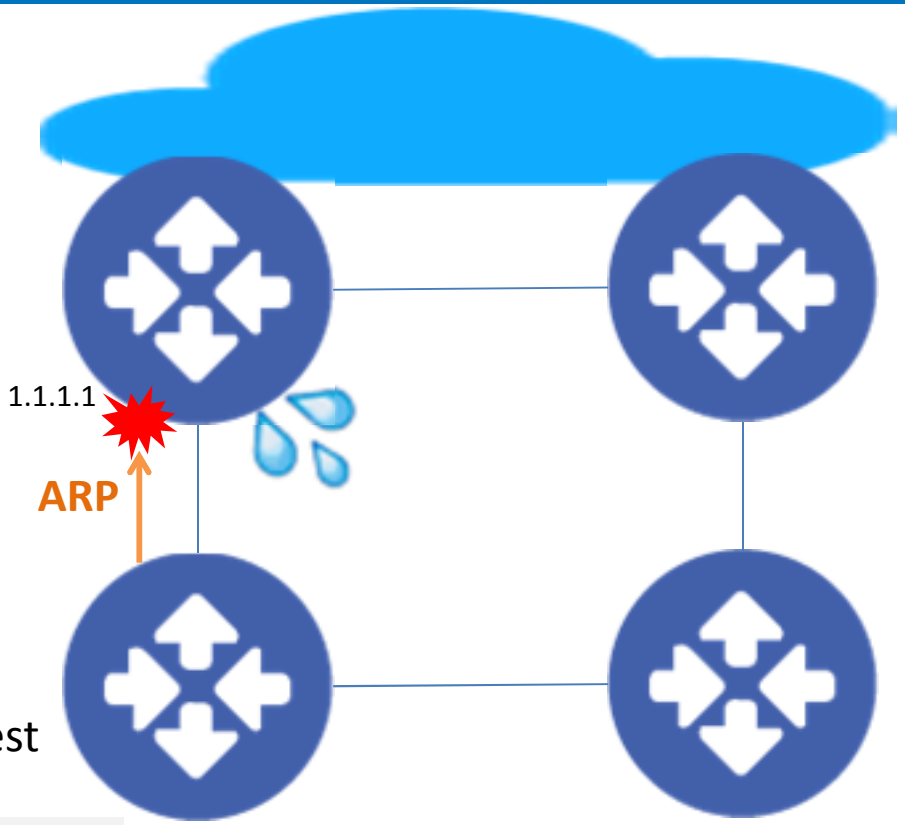


> 例えば2、ルータ間のトラフィック切り戻し



メンテナンス等で、インタフェースを落として迂回している状態

00:00:00.02 Link Up



00:00:00.00 Link Up
00:00:00.01 ARP Request

0.0.0.0/0 via 1.1.1.1

下位機器がLink UpしてARP Requestを投げても
上位機器がReadyな状態になっていないと
ARP Requestが受信されない

大量のARP更新には
落とし穴があることも

ARP Timerを5分に設定
1分間隔にExpireが近いものを
まとめてARP更新

Who has 198.51.100.1?
Who has 198.51.100.2?
Who has 198.51.100.3?
Who has 198.51.100.4?
Who has 198.51.100.5?
Who has 198.51.100.6? ×
Who has 198.51.100.7?

2000個くらい

10,000 ARP

自分が送ったARP Requestの
Replyを捌けないことも・・

仮想ルータ

以上、ARPトラブルあるあるでした。
他にもこんなのあるぜ！という方は
Qiita等にコメント下さい。

ご清聴ありがとうございました！！