



Internet Week 2009
～v4枯渇時代のシステムインテグレーション～
システムインテグレータにおける
IPv4アドレス在庫枯渇に向けた取り組み

2009年11月25日

株式会社NTTデータ
ビジネスソリューション事業本部
ネットワークソリューションBU
馬場 達也



IPv6への取り組みの必要性



IPv4アドレス枯渇問題の概要

● IPv4グローバルアドレス枯渇の時期

- － IANA (Internet Assigned Numbers Authority) の在庫が枯渇するのは**2011年9月頃**、RIR (地域インターネットレジストリ) の在庫が枯渇するのは**2012年10月頃**と予測されている(2009年11月5日現在) 参考: <http://www.potaroo.net/tools/ipv4/>
- － 根本的な対策は**IPv6への対応**(ただし、IPv4とIPv6は互換性がない！)

● IPv4グローバルアドレス枯渇の影響

- ✓ 自社が提供するサービスのお客様が他社に逃げてしまう
 - ・ 2011年以降、IPv6でしかアクセスできないユーザがでてくる
 - ・ そのユーザは、サービスがIPv6に対応していない場合は利用することができないため、IPv6に対応している他社サービスに乗り換えてしまう
- ✓ インターネット上のサービスが使えなくなる
 - ・ 2011年以降、新規に立ち上げるサービスはIPv6でしかサービスがされない可能性がある
 - ・ 自社ネットワークがIPv6に対応していない場合は、IPv6でしかアクセスできないサービスが利用できなくなる

自社のビジネスを守るためにIPv6を導入しなければならないという認識が必要



IPv4アドレス枯渇問題の影響を受けるシステム

- 影響を受けるシステム(対策の必要があるシステム)
 - － インターネットを介して、不特定多数のユーザに対してサービスを提供しているシステム(ECサイト、インターネットバンキングサイト、ASP/SaaSなど)

- 将来影響を受ける可能性のあるシステム(対策の判断の必要があるシステム)
 - － インターネットを介してサービスを提供しているシステムであって、サービス提供側またはサービス利用側が、今後の拡張に十分なIPv4グローバルアドレスを保有していないシステム

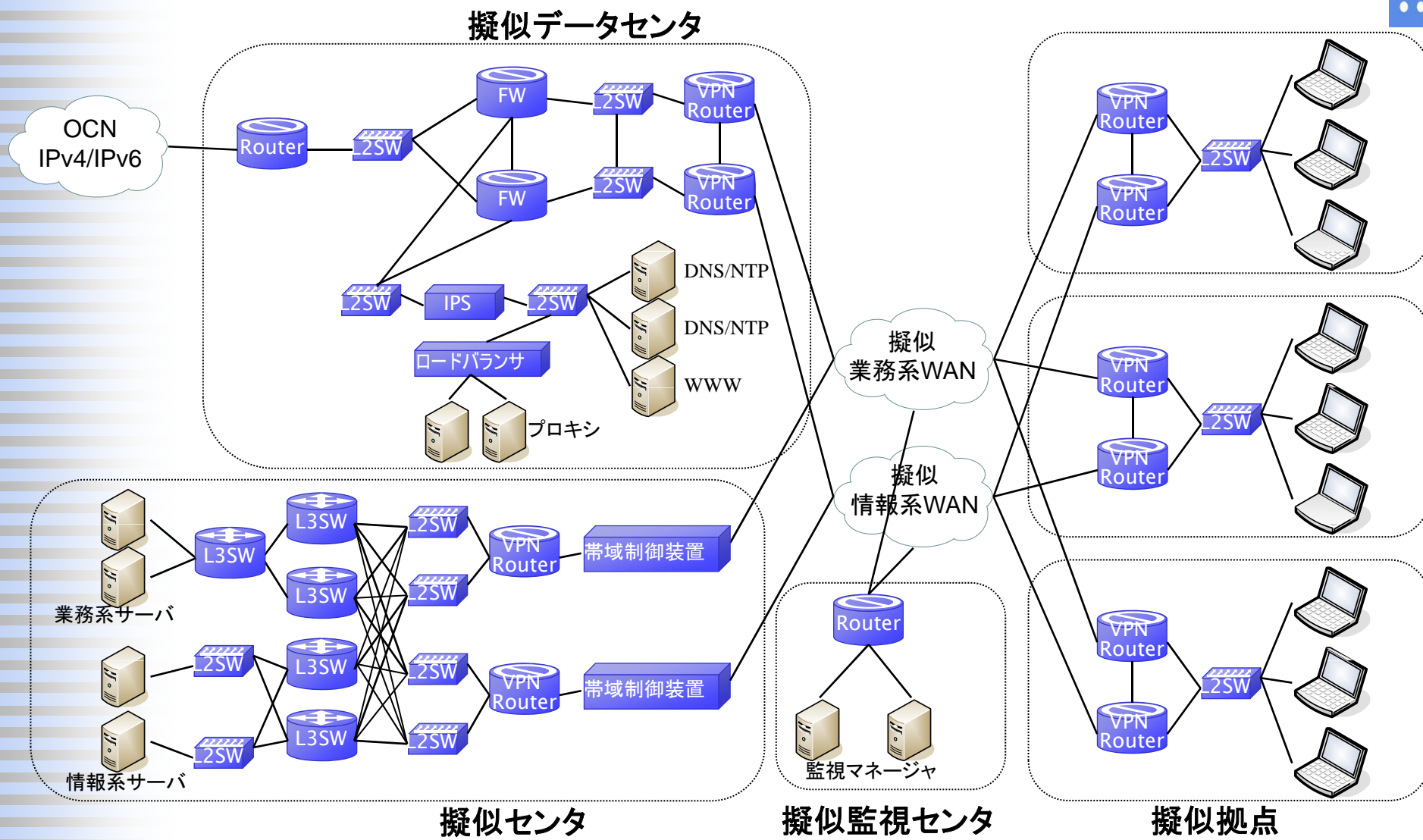
- 影響を受けないシステム(対策の必要がないシステム)
 - － 直接インターネットを介して通信していないシステム
 - － インターネットを利用していても、既にIPv4グローバルアドレスを十分確保している組織のみに対してサービスを提供しているシステム



製品/サービスのIPv6対応状況



NTTデータにおけるIPv4/IPv6デュアルスタック環境の検証





OS/ミドルウェアのIPv6対応

● OS

- Windows XP SP1/Windows Server 2003 ⇒ 一部対応(デフォルト無効)
- Windows Vista/Windows Server 2008 ⇒ 完全対応(デフォルト有効)
- Windows 7/Windows Server 2008 R2 ⇒ 完全対応(デフォルト有効)
- Red Hat Enterprise Linux 4以降
- Mac OS X 10.2以降
- Solaris 8以降
- HP-UX 11i以降

● アプリケーションサーバ

- IBM WebSphere Application Server 6.0以降
- Oracle WebLogic Server(旧 BEA WebLogic Server) 9.0以降
- Oracle Application Server 10g以降
- その他

OSや代表的なアプリケーションサーバでの対応はOK



各種サーバアプリケーションのIPv6対応状況

- Webサーバ
 - － Apache 2.0以降
 - － Microsoft IIS 6.0以降

- DNSサーバ
 - － BIND 8.4.0以降
 - － Windows Server 2003以降のDNSサーバ

- メールサーバ
 - － Sendmail 8.10以降
 - － Postfix 2.2以降

- プロキシサーバ
 - － Squid 3.1以降
 - － Apache 2.0以降 + mod_proxy



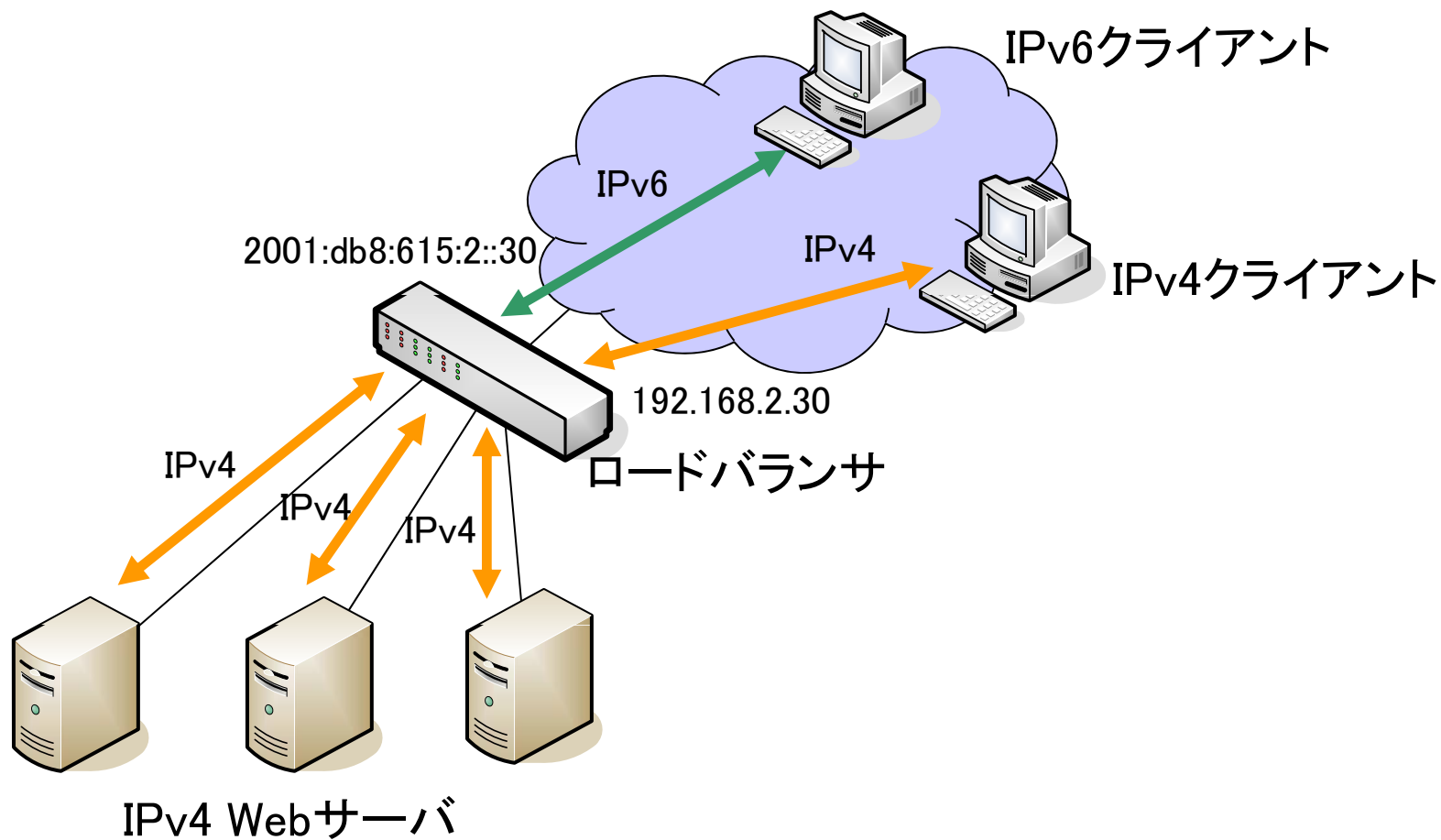
ネットワーク機器のIPv6対応状況

- ルータ/レイヤ3スイッチ
 - ・ 多くのルータで対応済み
 - ・ ただし、IPv4で実装されているすべての機能がIPv6に対応しているわけではない
- ファイアウォール
 - ・ Checkpoint IPシリーズ、Fortinet FortiGateシリーズなどが対応
 - ・ 冗長化構成時のIPv6のステート同期に対応していないものが多く、障害発生時にセッションが切断されるなどの問題がある
- IPS (Intrusion Prevention System)
 - ・ IBM Proventia Network IPS、McAfee Network Security Platform (旧 IntruShield)、Cisco IPS 4200シリーズなどが対応
- ロードバランサ
 - ・ F5 Networks BIG-IP Local Traffic Manager、A10 Networks AXシリーズ、Citrix NetScalerなどが対応
- リモートアクセスVPNサーバ
 - ・ SSL-VPN製品やIPsec-VPN製品など、対応している製品が少ない
 - ・ Windows Server 2008 R2 + Windows 7のDirectAccessはIPv6で動作！



ロードバランサでのIPv6対応

ロードバランサがIPv4 ⇔ IPv6の変換を行うため、
サーバがIPv6対応していなくてもOK

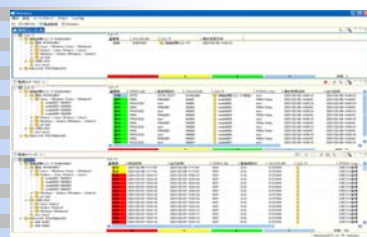




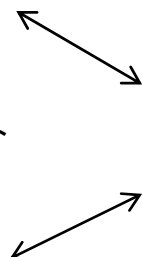
運用管理ソフトウェアのIPv6対応状況

日立製作所のJP1/Cm2/Network Node Manager、
横河電機のLMaT-AR、HinemosなどがIPv6ネットワークの監視に対応

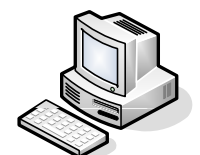
Hinemos



Hinemos
クライアント

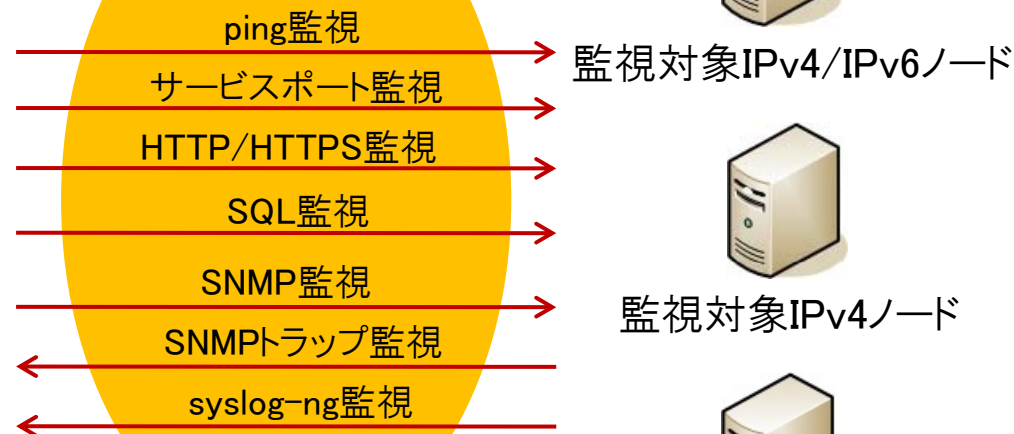


Hinemos
マネージャ



Hinemos
クライアント

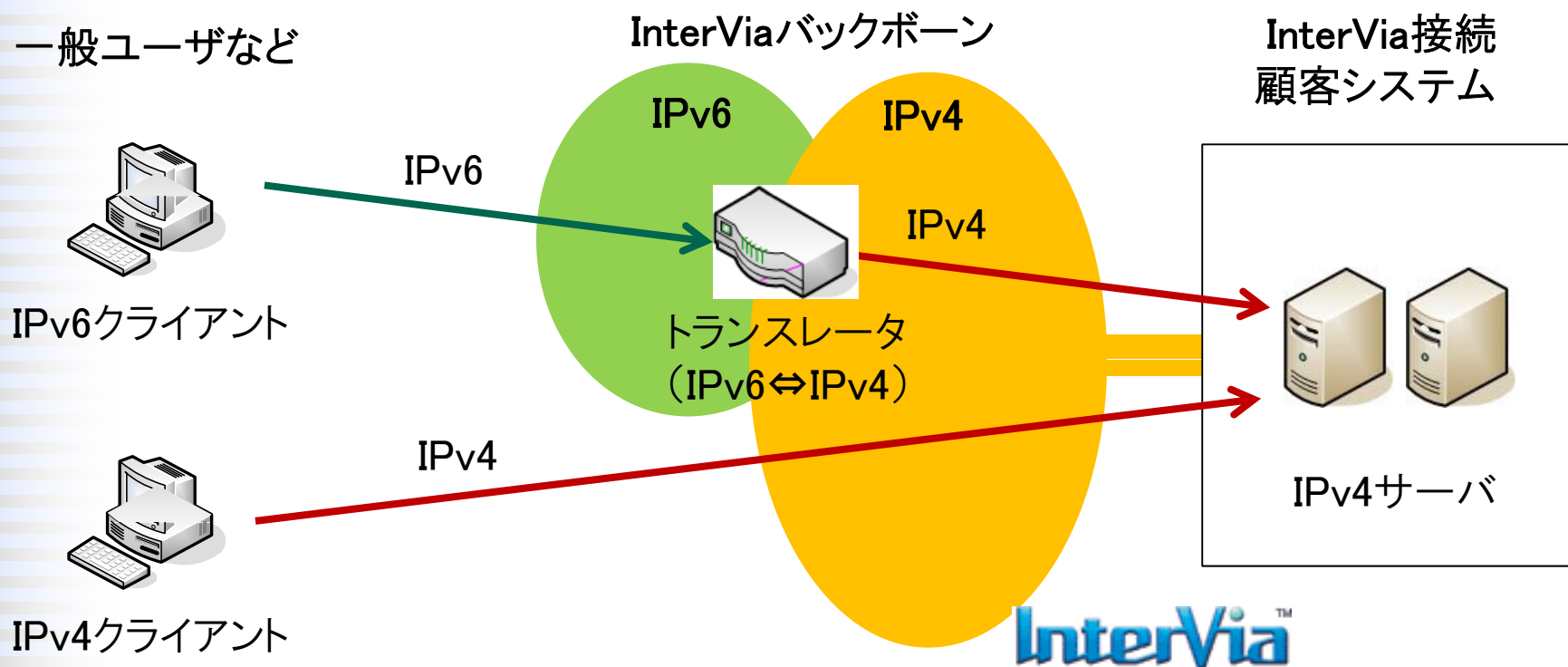
IPv4/IPv6混在ネットワーク





InterVia「IPv6/IPv4トランスレートサービス」

InterVia (ISP) 内でIPv6をIPv4に変換するサービス
(システム側がIPv6に対応できない場合でもIPv6ユーザからのアクセスを受けることが可能)





WANサービスのIPv6対応状況

- 広域イーサネット
 - イーサフレームで処理されるため、IPv4/IPv6パケットを問題なく利用可能
- IP-VPN
 - IPレベルでルーティングを行っているため、IPv6に対応したサービスを選択する必要がある
 - NTTコミュニケーションズの「Arcstar IP-VPN」IPv6デュアル
 - 対応していない場合は、IPv6 over IPv4トンネリングで対応
- フレッツVPN
 - IPv6に対応していない
 - IPv6 over IPv4トンネリングで対応
- ISDNなど
 - PPPでカプセル化するため、接続機器がIPv6 over PPPに対応していれば問題なく利用可能



インターネット接続サービスのIPv6対応

- 大手ISPではIPv6対応済み
- IPv6インターネットへの接続方法は3種類

IPv6トンネル接続サービス

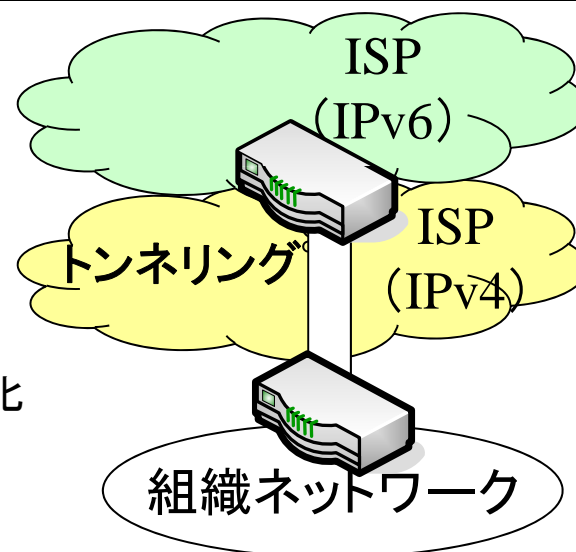
- IPv6 over IPv4トンネリングにより接続
- トンネリングのオーバーヘッドにより性能が劣化
- アクセス線がIPv6に対応している必要がない
- 接続料金が安い

IPv6ネイティブ接続サービス

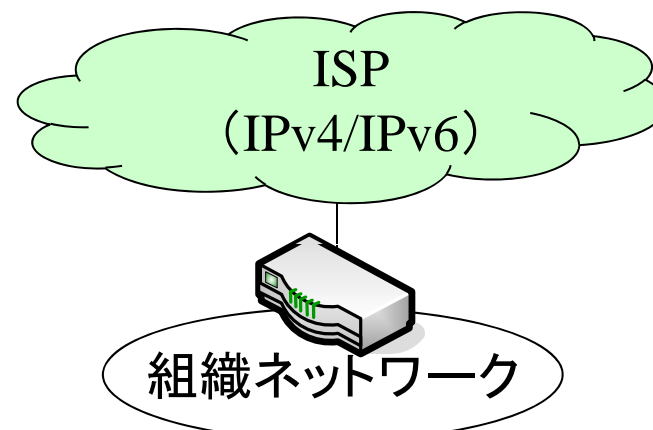
- IPv6専用のため、IPv4接続用は別に必要
- IPv6接続のオーバーヘッドはない
- 接続料金が安い

IPv6デュアル接続サービス

- IPv4およびIPv6の両方に対応
- IPv6接続のオーバーヘッドはない
- 接続料金が安い



IPv6トンネル接続



IPv6デュアル接続



IPv6導入検証の結論

- IPv4/IPv6デュアルスタックネットワークの構築は可能
- ただし、以下のような問題がある
 - ロードバランサ、IPS/IDS、リモートアクセスVPNサーバなどは、IPv6に対応していない製品がまだ多い ⇒ **製品の調査が必要！**
 - IPv6に対応している製品でも、IPv4で実装されている機能がIPv6では実装されていないことが多い ⇒ **機能ごとの確認が必要！**
 - VPN機能など、IPv6はハードウェア処理しないものがあり、性能面での問題が発生する可能性がある ⇒ **事前検証が必要！**
 - ファイアウォールの設定や障害の切り分けなどは、IPv6のプロトコルの挙動を知らないと難しい ⇒ **教育が必要！**

問題はあるが、経験とスキル、運用ノウハウでカバー可能



自社開発ソフトウェアのIPv6対応



ソフトウェアをIPv6対応にするうえで注意するポイント

IPv4とIPv6はアドレスの長さや表記方法が異なります！

- GUIや設定ファイルにおいて、IPv4アドレスの入力が想定されていないか？
 - ・ 例えば、下記のような入力画面を用意していないか？

IPアドレス:

...

サブネットマスク:

...

- ログ(DB)や画面などへのIPv4アドレスの出力を想定していないか？
- アクセス制御など、プログラム内部でIPv4アドレスが使用されていないか？
 - ・ 4バイトの変数にIPアドレスをバイナリ形式で代入
- プログラム中にIPv4アドレスが埋め込まれていないか？
 - ・ サーバアドレス/ループバックアドレス埋め込み



言語特有の問題も存在(C言語の例)

- IPv4/IPv6デュアルスタック環境で問題の発生する型や関数が使われていないか？
 - ・ 「in_addr」, 「sockaddr_in」, 「inet_addr」, 「inet_aton」, 「inet_lnaof」, 「inet_makeaddr」, 「inet_netof」, 「inet_network」, 「inet_ntoa」, 「inet_ntop」, 「inet_pton」, 「getservbyport」, 「gethostbyname」, 「gethostbyname2」, 「gethostbyaddr」, 「getservbyname」が使われている場合は問題が発生する
 - ➡ 「sockaddr_storage」, 「getaddrinfo」, 「addrinfo」, 「if_laddrreq」, 「if_laddrconf」などのIPv6に対応した型や関数に書き換える
- ソケットの設定がIPv4を使用するようになっていないか？
 - ・ ソケットの設定値に「AF_INET」と直接記述されている場合
 - s = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
 - ➡ getaddrinfo関数又はgetnameinfo関数を用いて、端末がサポートするプロトコルファミリーを取得し、ソケットの確立を行うよう書き換える



IPv4に依存したソースコードの例(C言語)

```
int i, s;
struct hostent *hp;
struct sockaddr_in sin;

s = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
hp = gethostbyname("www.nttdata.co.jp");
for (i = 0; hp->h_addr_list[i]; i++) {
    memset(&sin, 0, sizeof(sin));
    sin.sin_family = AF_INET;
    sin.sin_len = sizeof(sin);
    sin.sin_port = htons(80);
    memcpy(&sin.sin_addr, hp->h_addr_list[i], hp->h_length);
    if (connect(s, &sin, sizeof(sin)) < 0){
        continue;
    }
    break;
}
```

// gethostbyname関数の結果を格納する構造体
// 接続情報を格納する構造体

// IPv4のTCPソケットを作成
// 接続先サーバのIPv4アドレスを取得

// アドレスファミリーとしてIPv4を設定

// ポート番号として80番を設定
//取得した IPv4アドレスを設定
// 上記の条件で接続

sockaddr_in型、gethostbyname関数、
AF_INETをハードコーディングしたソケット設定がIPv4に依存している



IPv4に依存しないソースコードの例(C言語)

```
int s;
struct addrinfo hints;
struct addrinfo *res, *res0;

memset(&hints, 0, sizeof(hints));
hints.ai_family = PF_UNSPEC;
hints.ai_socktype = SOCK_STREAM;
getaddrinfo("www.nttdata.co.jp", "http", &hints, &res0);
for (res = res0; res; res = res->ai_next) {
    s = socket(res->ai_family, res->ai_socktype, res->ai_protocol);
    if (connect(s, res->ai_addr, res->ai_addrlen) < 0) {
        close(s);
        continue;
    }
    break;
}
freeaddrinfo(res0);
```

// 接続条件を格納する構造体
// getaddrinfo関数の結果を格納する構造体

// プロトコルを指定しない(IPv4とIPv6に対応)
// TCPを指定
// 接続先サーバのIPアドレスを取得

// ソケットを作成
// 上記の条件で接続

- sockaddr_in型の代わりにaddrinfo型を使用
- gethostbyname関数の代わりにgetaddrinfo関数を使用
- アドレスファミリをハードコーディングしないソケット設定に変更

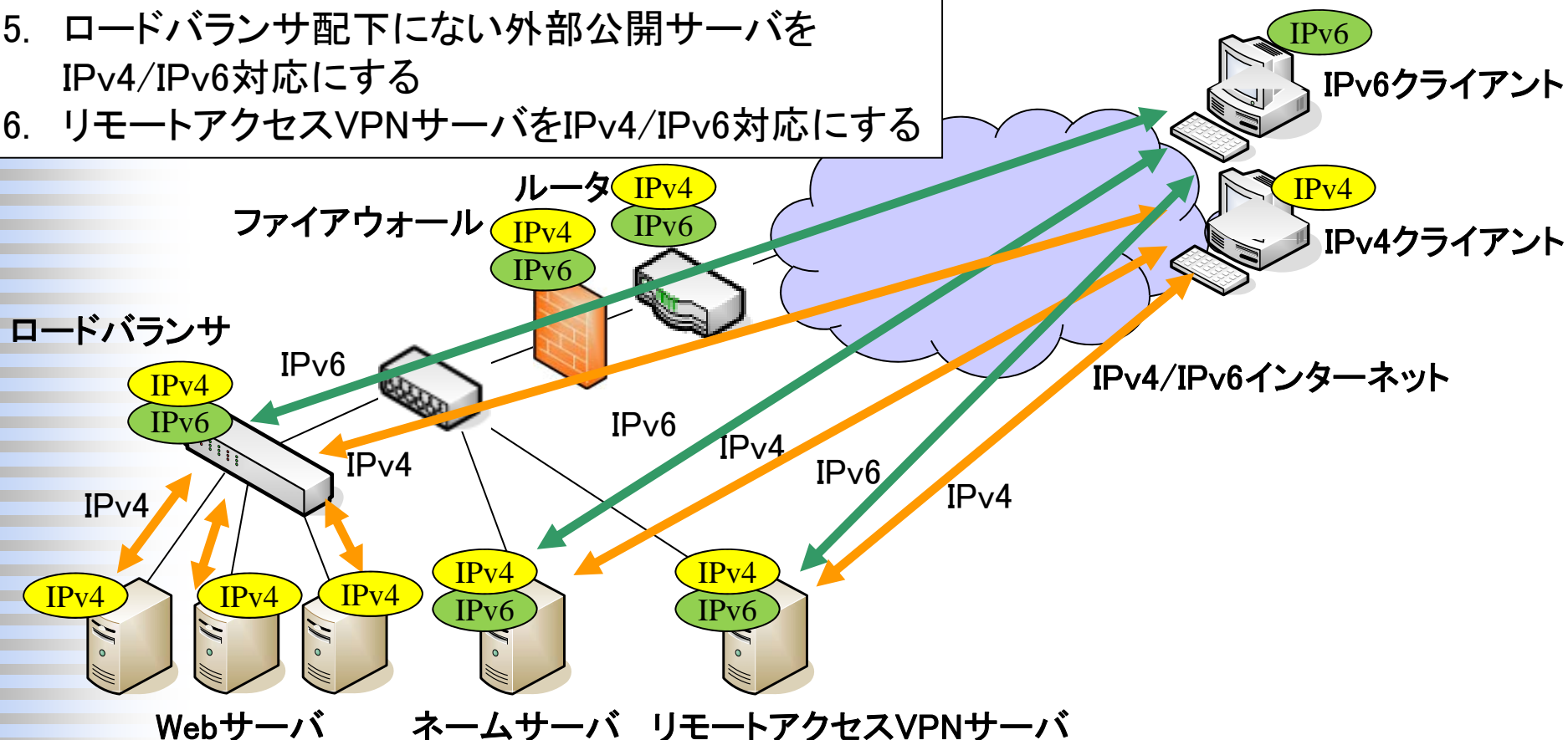


企業ネットワークのIPv6移行



STEP1 外部に公開しているサーバをIPv6でアクセス可能にする

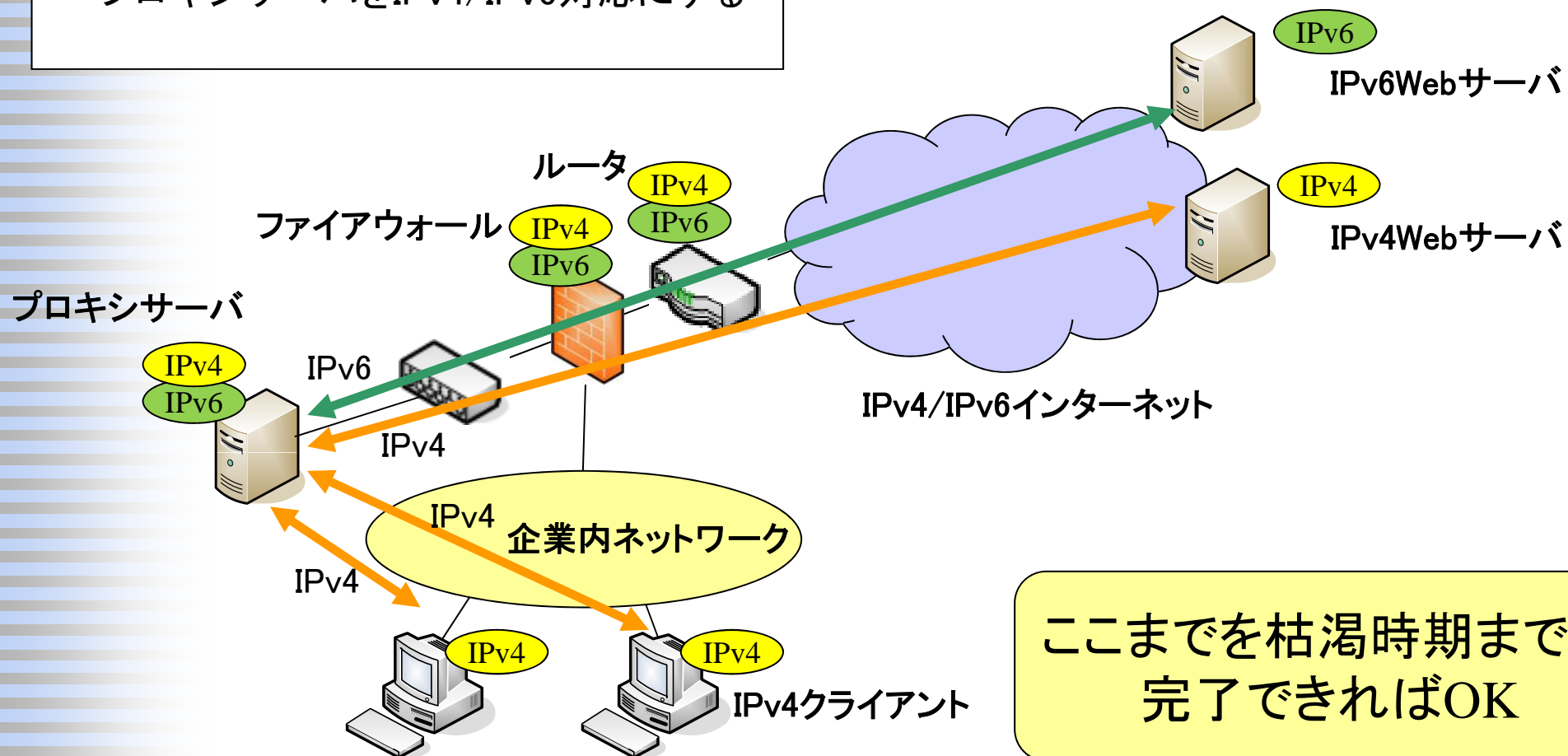
1. ISPのIPv6接続サービスを申し込む
2. インターネット接続用ルータをIPv4/IPv6対応にする
3. ファイアウォールをIPv4/IPv6対応にする
4. ロードバランサをIPv4/IPv6対応にする
5. ロードバランサ配下でない外部公開サーバをIPv4/IPv6対応にする
6. リモートアクセスVPNサーバをIPv4/IPv6対応にする





STEP2 内部から外部WebサーバにIPv6でアクセス可能にする

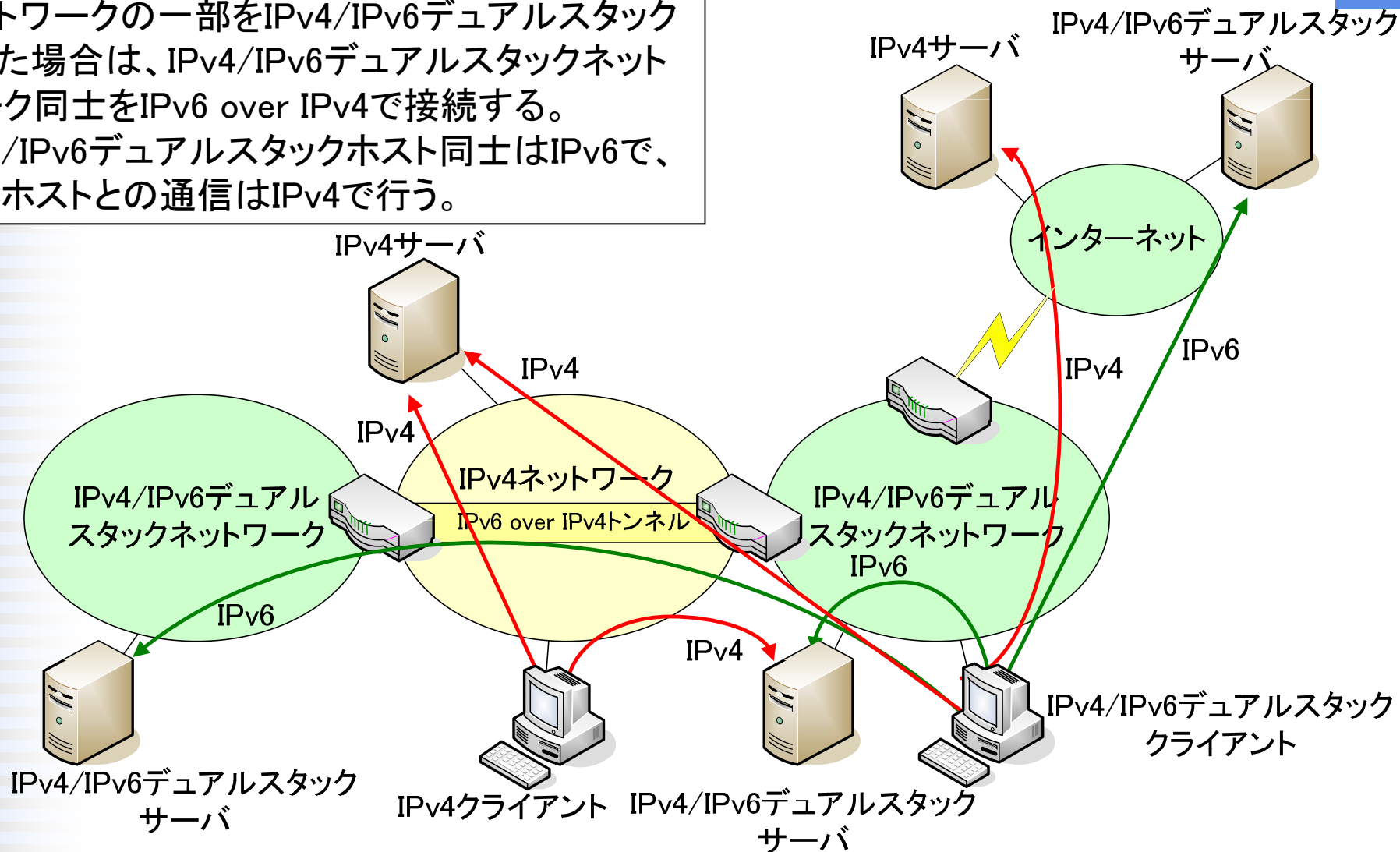
プロキシサーバをIPv4/IPv6対応にする





STEP3 社内ネットワークはできるところからデュアルスタック化

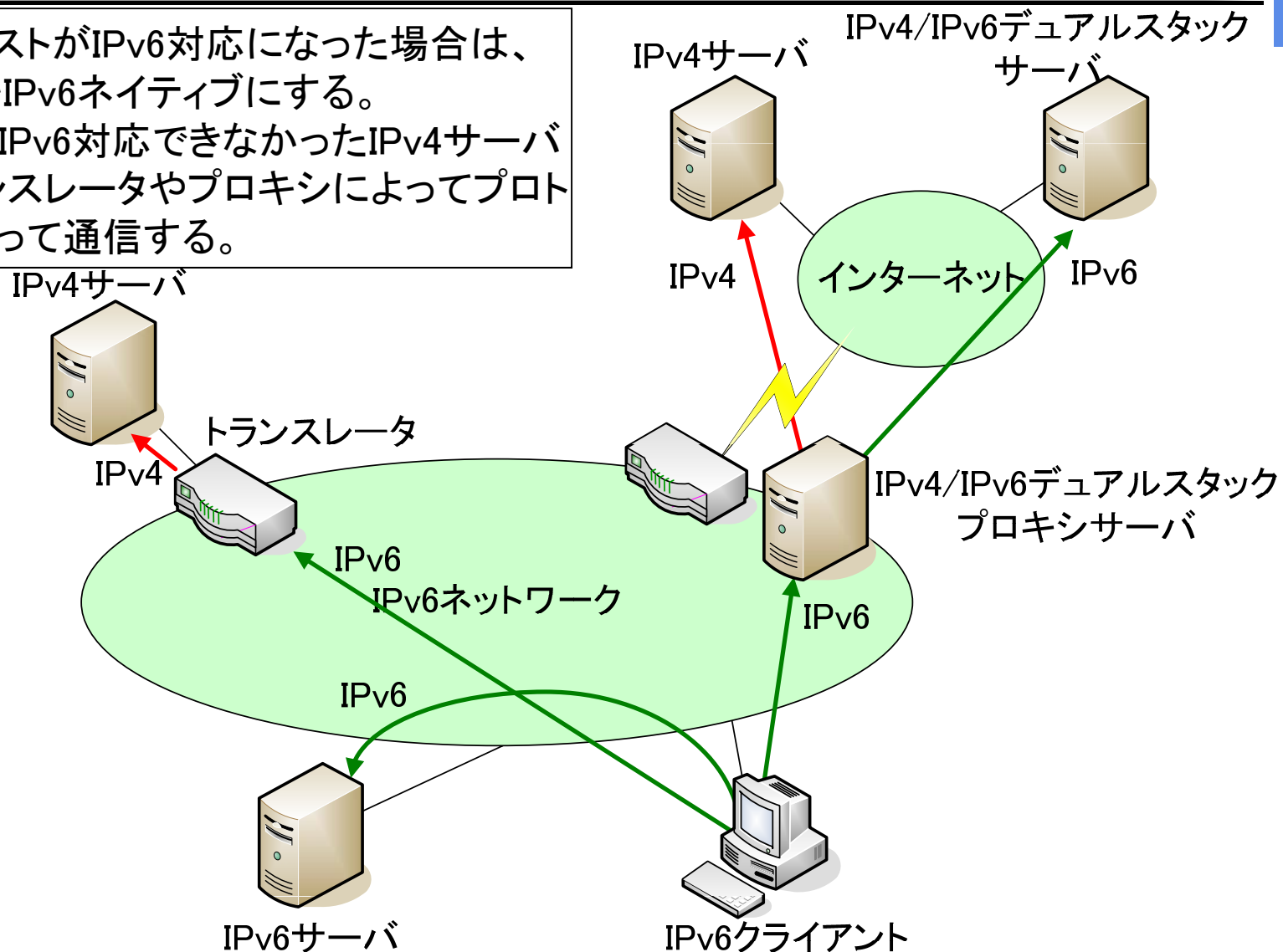
ネットワークの一部をIPv4/IPv6デュアルスタックにした場合は、IPv4/IPv6デュアルスタックネットワーク同士をIPv6 over IPv4で接続する。IPv4/IPv6デュアルスタックホスト同士はIPv6で、IPv4ホストとの通信はIPv4で行う。





STEP4 IPv6ネイティブネットワークへの移行

ほとんどのホストがIPv6対応になった場合は、ネットワークをIPv6ネイティブにする。IPv6ホストと、IPv6対応できなかったIPv4サーバの間は、トランスレータやプロキシによってプロトコル変換を行って通信する。





最後に ～IPv6対応のポイント～

1. IPv6対応が必要なシステムを洗い出す
 - インターネット経由で不特定多数の方がアクセスするシステム
 - 将来の拡張に十分なIPv4グローバルアドレスを確保していないシステム
2. 現在利用されている製品/サービス/自社開発ソフトウェアを調査する
 - 使用している製品がIPv6に対応しているかどうか
 - 使用しているサービスがIPv6に対応しているかどうか
 - 使用している自社開発ソフトウェアがIPv6に対応しているかどうか
3. IPv6対応計画を策定する
 - 対応の必要性と難易度、更改タイミングなどを考慮して各システムの対応スケジュールを検討
 - 対応のための予算の確保

残された時間はわずかです！早めの対応を！



変える力を、ともに生み出す。

NTT DATAグループ

NTT
Data