

トランスレータ技術の体系化と実現可能性の検討

日立製作所

角川宗近

sumikawa@sumikawa.jp

概要

- 語句定義
- トランスレータを分類
 - プロトコル変換の技術
 - アドレスの対応付け
- 移行時期と通信方向によるタイプ分け
- 各タイプの実現可能性を検討
- 既存のトランスレータ技術の考察
- まとめ

語句定義

■ "[IPv4/IPv6][プロトコル・]トランスレータ"

■ not "IPv4-IPv6 NAT"

■ not "v6 NAT"

- IPv4とIPv6を透過的に通信させる
- 1台以上のノードが対象
 - ルータ
 - トランスレータ・ボックス
 - ハブ
 - MAU
 - ネットワーク・カード
 - ドライバ

トランスレータの分類

■ 2つの視点から分類

- プロトコル変換の技術
- アドレスの対応付け

プロトコル変換の技術

- ヘッダ変換
- トランスポート中継
- 代理アプリケーション

ヘッダ変換

- ヘッダを入れ替えて転送
- チェックサムをヘッダの分だけ補正
- ftp問題
- ICMPのセマンティクスが違う
- v4->v6変換時に20バイト増えるので分割しないといけない
- UDPゼロチェックサム問題

トランスポート中継

- コネクションを横取りして、張り直す。
 - ファイアウォール、WWW横取りキャッシュ
- ftp問題はあるが、比較的簡単
- 2本のコネクションを扱う時(rsh等)にちょっと大変

代理アプリケーション

- トランザクション・サービスを転送
- プロトコル的に代理サービスをサポートしている必要がある
- ftp問題なし

アドレスの対応づけ

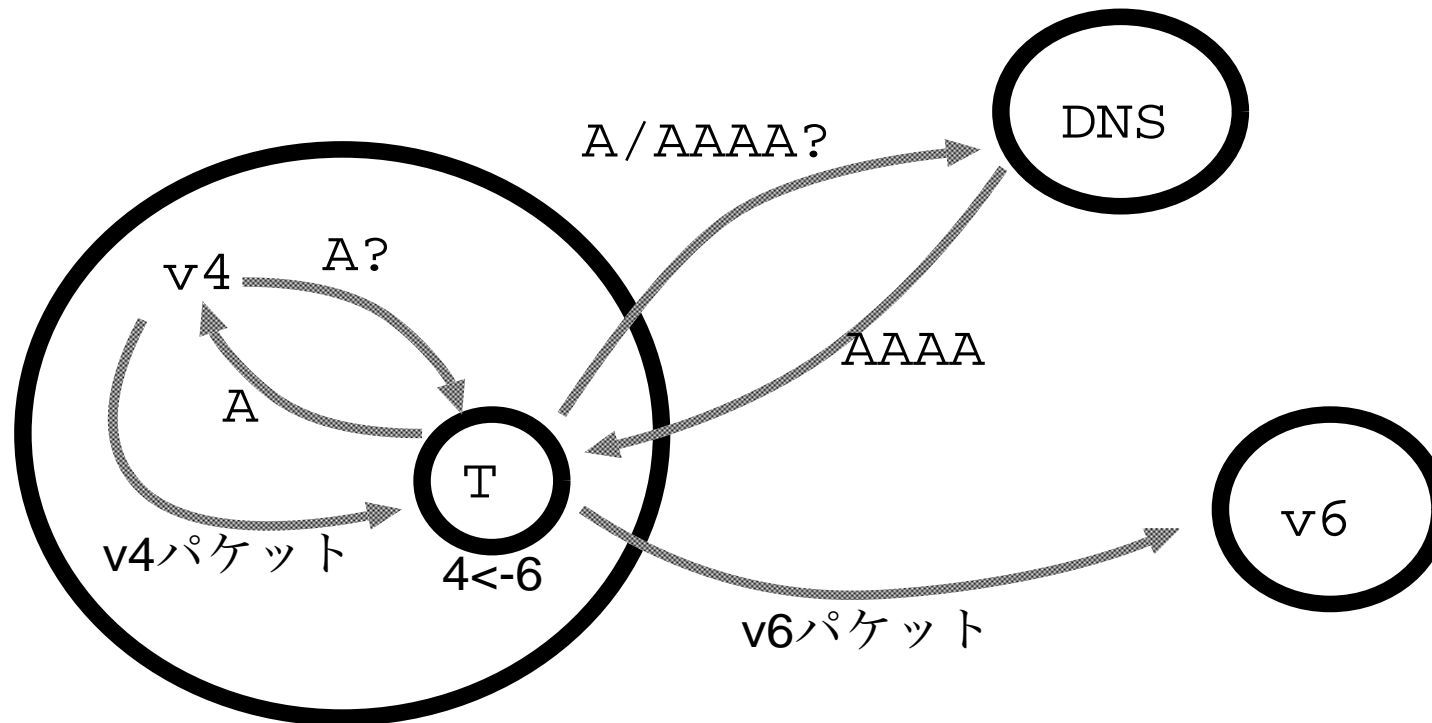
- 静的な対応づけ
- DNSによる対応づけ
- 名前で通信

静的な対応付け

- v6空間にv4空間を対応づけ
- 広い空間に狭い空間をマップするので静的でよい
 - 合宿ネットは 3ffe:501:10ff::/48 の一部を利用
 - 100.0.0.1 -> 3ffe:501:10ff:2000::100.0.0.1

DNSによる対応づけ

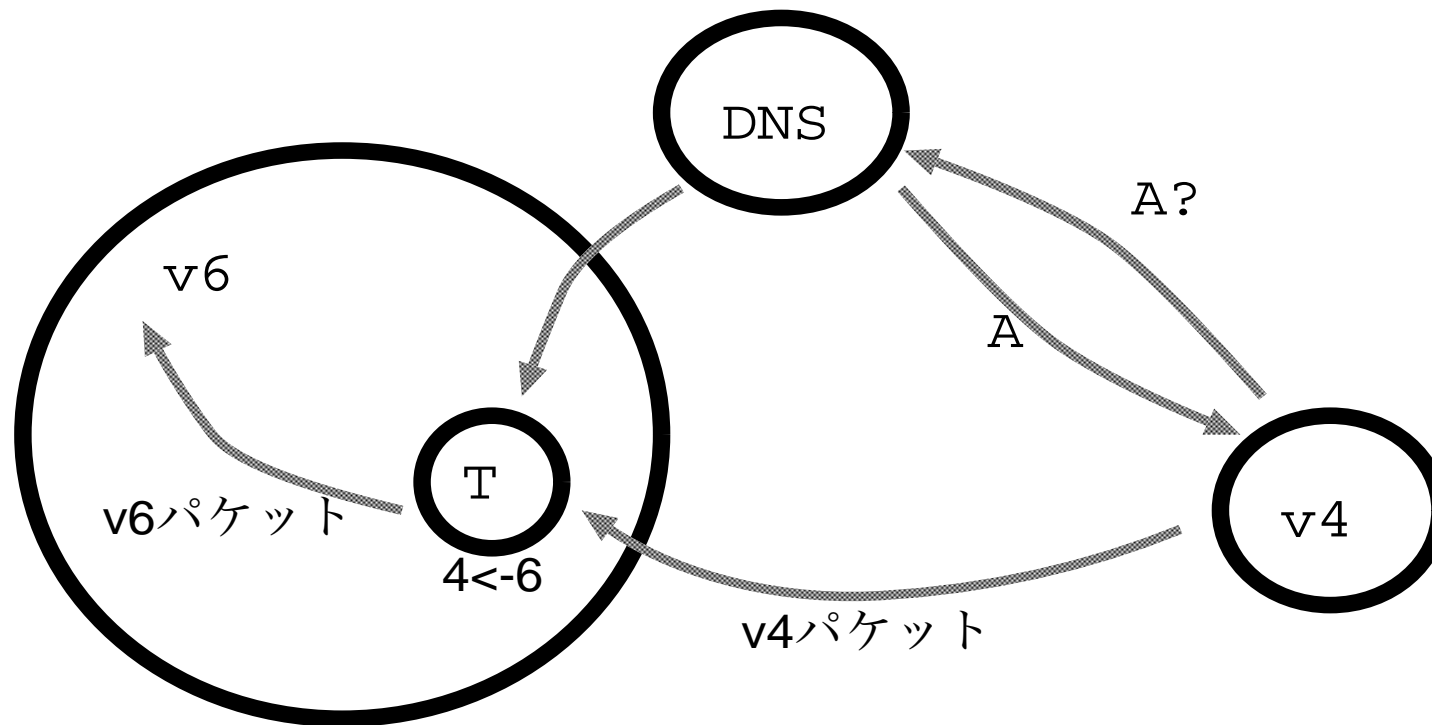
- v4空間にv6空間を対応づけ
- 動的に対応付ける必要がある



- DNSキャッシュ問題がちょっとだけあり

DNSによる対応づけ(2)

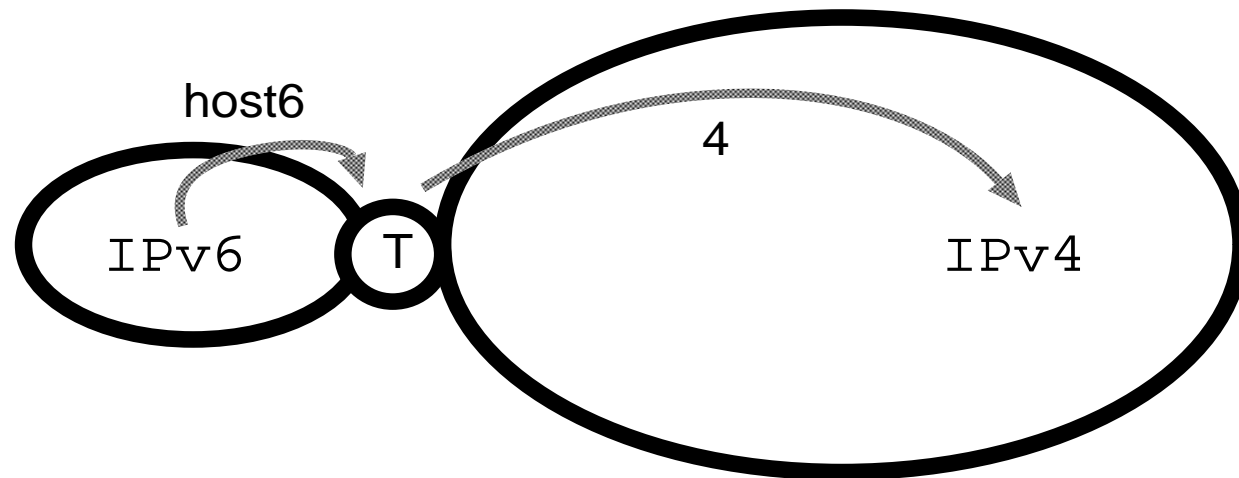
■ 外からの場合



■ 外からの通信はDNSキャッシュが問題

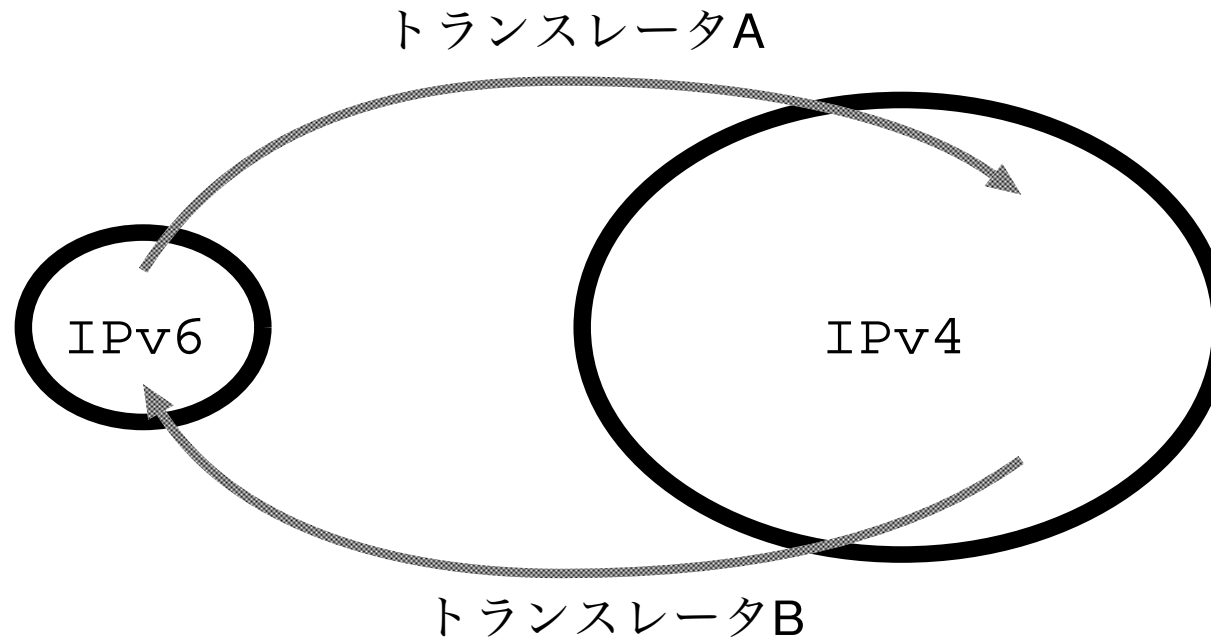
名前で通信

- 名前空間は連続している
- 隣接ポイントまで名前での通信
 - サービス毎に独自のプロトコル



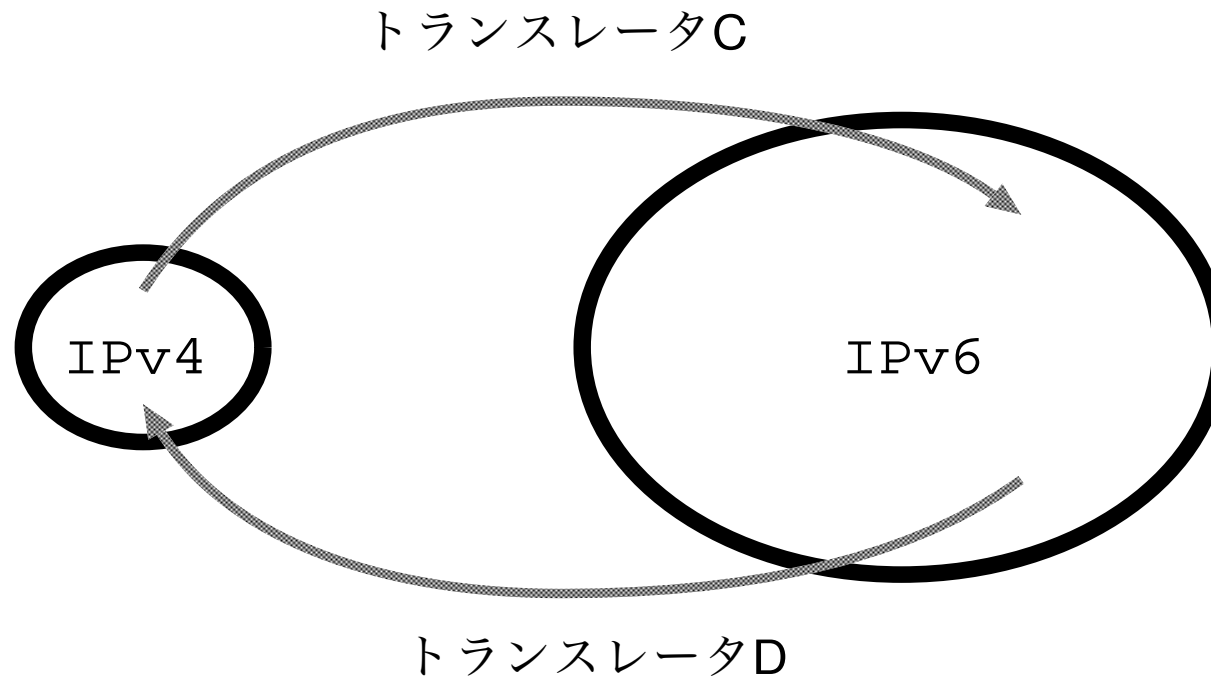
移行時期と通信方向によるタイプ分け

移行初期



移行時期と通信方向によるタイプ分け(2)

移行後期



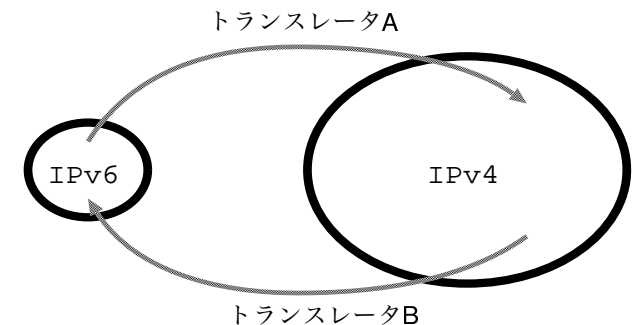
各タイプの実現可能性

トランスレータA

- v6からv4への静的対応づけ
- DNSキャッシュ問題： なし
- 実現可能性： 簡単

トランスレータB

- v4からv6への動的対応づけ
- DNSキャッシュ問題： あり
- 実現可能性： 不可能に近い



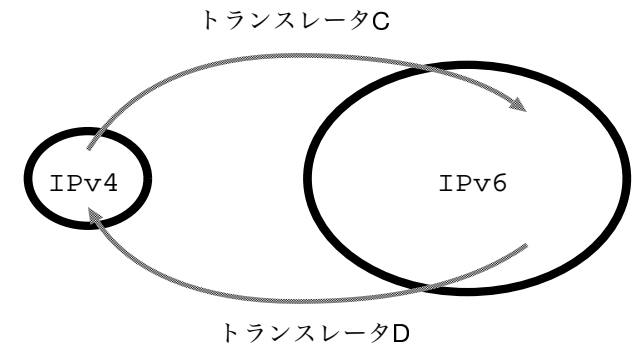
タイプ毎の問題点(2)

トランスレータC

- v4からv6への動的対応づけ
- DNSキャッシュ問題： サイト内に閉じている
- 実現可能性： 可能

トランスレータD

- v6からv4への静的対応づけ
- DNSキャッシュ問題： なし
- 実現可能性： 簡単



WIDEのトランスレータ技術

■ FAITH

- トランスレータA、D
- トランスポート中継 + 静的対応づけ
- 実現容易

■ SOCKS64

- トランスレータA、C
- トランスポート中継 + トランスレータまで名前での通信
- 実現容易

WIDEのトランスレータ技術(2)

■ BIS

- セルフ・トランスレータによるデュアルスタックの実現
- モデル的にはトランスレータではない
- ヘッダ変換 + 動的対応づけ

WIDE以外のトランスレータ技術

■ NAT-PT (Network Address Translation - Protocol Translation)

- トランスレータA、D
- ヘッダ変換 + 静的対応づけ

■ NAT-PT+DNS-ALG

- トランスレータB、C
- ヘッダ変換 + 動的対応づけ

■ SIIT(Stateless IP ICMP Translator)

- トランスレータA、(B、C、D)
- ただし制約条件が強い
- ヘッダ変換

■ apache、sendmail、bind ...

- トランスレータA、C
- 代理アプリケーション

まとめ

- 変換技術、アドレスの対応づけ方法の観点から分類
- トランスレータを移行時期、移行方向から4つに分類
 - それぞれの特徴、実現可能性を検討
- 既存の技術がどれに当てはまるか考察
- IDがngtrans WG Last Call終わりました
 - "Categorizing Translators between IPv4 and IPv6"
 - (draft-ietf-ngtrans-translator-01.txt)

おまけ

今回の合宿ネット

■ 合宿内：NATPT(静的マッピング)

■ SFC : FAITH

の対向トランスレータ



■ v6ユーザからするとトランスレータA

■ v4ユーザからすると、あまり意味ない

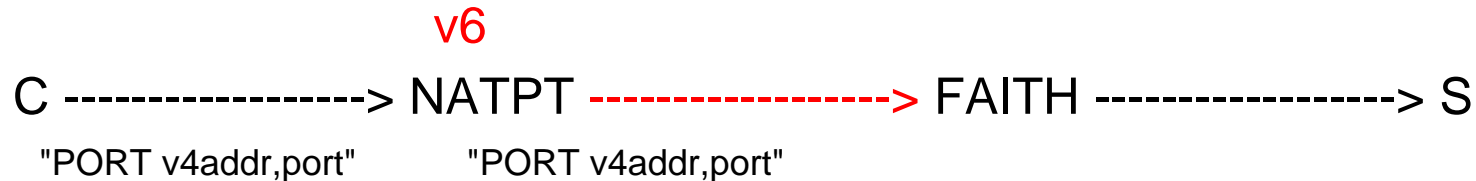
- v4 over v6 tunnel
- v4 with NAT

FTP問題(さらにおまけ)

前提条件

- faithはPORT、PASVの変換をする
- (今の)ふじさわNAT-PTは、PORT、PASVは変換しない。

PORTの場合



■ v6コネクションから、v4addrの指定が来てるので変換不能。

- 1. 今のFAITHの実装は落としてる。
- 2. よしんば無変換で通したとしても、
 - (1) clientがprivate addressなら通信できない
 - (2) clientがglobal addressなら通信できる

PASVの場合



■ v6コネクションから、PASVの指定が来てるので変換不能。

- 1. 今のFAITHの実装は落としてる。
- 2. よしんば通したとしても帰りが、

