NTMobile アダプタの実現と評価

小島　光野[†*]，尾久　史弥[†]，鈴木　秀和[†]，内藤　克浩[‡]，渡邊　晃[†]
([†] 名城大学，[‡] 愛知工業大学)

Realzation of NTMobile Adapter and its evaluation

Koya Kojima[†], Fumiya Ogyu[†], Hidekazu Suzuki[†], Katsuhiro Naito[‡], Akira Watanabe[†]
([†]Meijo University, [‡]Aichi Insutitute of Technology University)

1　はじめに

現状のネットワークは，NAT 越え問題や，通話中にネットワークが切り替えることができないなど様々な問題を抱えている．これらの問題を解決する技術として NTMobile がある [1].

NTMobile はエンド端末にアプリケーションを実装することによって，ネットワークの制約を意識することなくエンドツーエンドの通信を可能にする技術である．しかし，組込み型の家電や安定性を重視するサーバなど，NTMobile を実装することができない装置が存在する．このような装置に NTMobile の機能を与えるための方法として NTMobile アダプタ (NTMA) を提案する．本稿では受信を待ち受ける側の装置 (レスポンダ側) の NTMA を実現したので報告する．

2　NTMobile の概要

NTMobile は Direction Coordinator(DC)，NTMobile の機能を持つ NTM 端末から構成される．DC は各端末に対してトンネル構築の指示を出すとともに，NTM 端末の情報を管理している．

NTM 端末が起動すると，DC に実ＩＰアドレスを登録し，また，DC から仮想 IP アドレスの割り当てを受ける．アプリケーションは仮想 IP アドレスに基づいた通信を行い，実際の通信は実 IP アドレスでカプセル化される．そのため，ネットワークの切り替えに伴う実アドレスの変化をアプリケーションに対して隠蔽することができる．また，DC の指示に従ってトンネル経路を生成することにより NAT 越えを実現する．

3　レスポンダ側 NTMA の動作

Fig.1 に受信を待ち受ける側の装置 (レスポンダ側) に NTMA を設置した場合の通信シーケンスを示す．DC はグローバル空間に存在する．NTM 端末は多くの場合プライベート空間に存在するが，Fig.1 では NAT は省略されている．また，一般端末を GN(General Node）とする．GN にはアプリケーションサーバや組込み型家電機器などが想定される．NTMA は NIC を 2 枚持ち，ネットワークとＧＮの間に直列に設置する．

Fig.1 の前提として，NTM 端末と NTMA は仮想アドレスを取得済みである．また，NTMA には GN の名前と IP アドレスの関係を登録しておく必要がある．GN の名前は NTMobile 端末であることがわかるような名前とする．GN の実 IP アドレスには NTMA が DC から取得した仮想アドレスを設定しておく．

Fig.1 を用いて通信シーケンスの流れを説明する．まず，NTM 端末から DC へ Direction Request(経路指示要求) が送られる．次に DC は Route Direction(経路指示) を NTMA に送信する．NTMA からの応答を受け取ると，同じく経路指示を NTM 端末へ送信する．経路指示を受け取った NTM 端末は経路指示に従い，NTMA へ Tunnel Request(トンネル構築要求) を送信する．NTMA から Tunnel Response が返され，UDP トンネルが構築される．トンネル経路は NTM 端末の設置場所により異なるが，Fig.1 では最も簡単な例を示している．以降の通信はすべてトンネル通信で行われる．

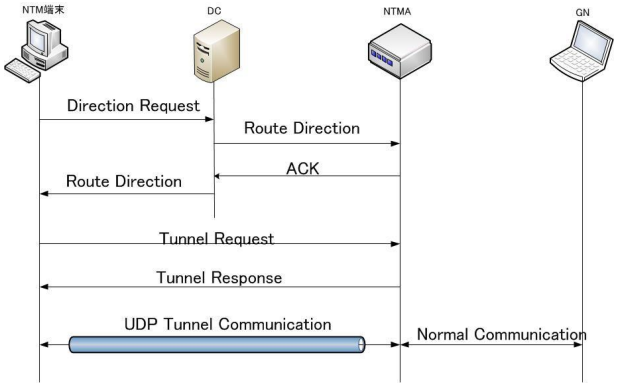


Fig. 1　Sequence of the proposed system

トンネル通信では，NTM 端末のアプリケーションと GN に割り当てられた仮想 IP アドレスによる通信を実際の IP ヘッダを付与してカプセル化する．

パケットの IP アドレスの変化は以下のとおりである．NTM 端末で送受信されるパケットは，送信元，宛先 IP アドレスを自身の仮想 IP アドレスと GN の仮想 IP アドレスとしたパケットを，NTM 端末と NTMA の実 IP アドレスでカプセル化したものである．NTMA ではカプセル化とデカプセル化処理を行い，仮想アドレスのパケットをそのまま中継する．

Fig.1 の動作を検証し，正しく動作することを確認した．また，NTMA の中継時間はわずかであることを確認した．

4　まとめ

NTMobile を実装できない装置に NTMobile 機能を与える NTMobile アダプタをレスポンダ側に設置した場合について提案した．

文　献

[1] 上醉尾．他：IPv4/IPv6 混在環境での移動透過性を実現する NTMobile の実装と評価情報処理学会論文誌 Vol.54,No.10,pp.2288-2299,Oct 2013.

[2] 尾久．他：NTMobile アダプタの実現方式の検討．情報処理学会第 79 回全国大会講演論文集 Mar 2017

NTMobileアダプタの実現と評価

小島　光野†，尾久　史弥†，鈴木　秀和†，内藤　克浩‡，渡邊　晃†
†名城大学　理工学部　情報工学科
‡愛知工業大学　情報科学部

Watanabe Lab.

研究背景

- 現在のネットワーク　→　IPv4ネットワークが主流

- IPアドレスの枯渇　→　NATを介したネットワークの構築
  IPv6アドレスの導入

研究背景

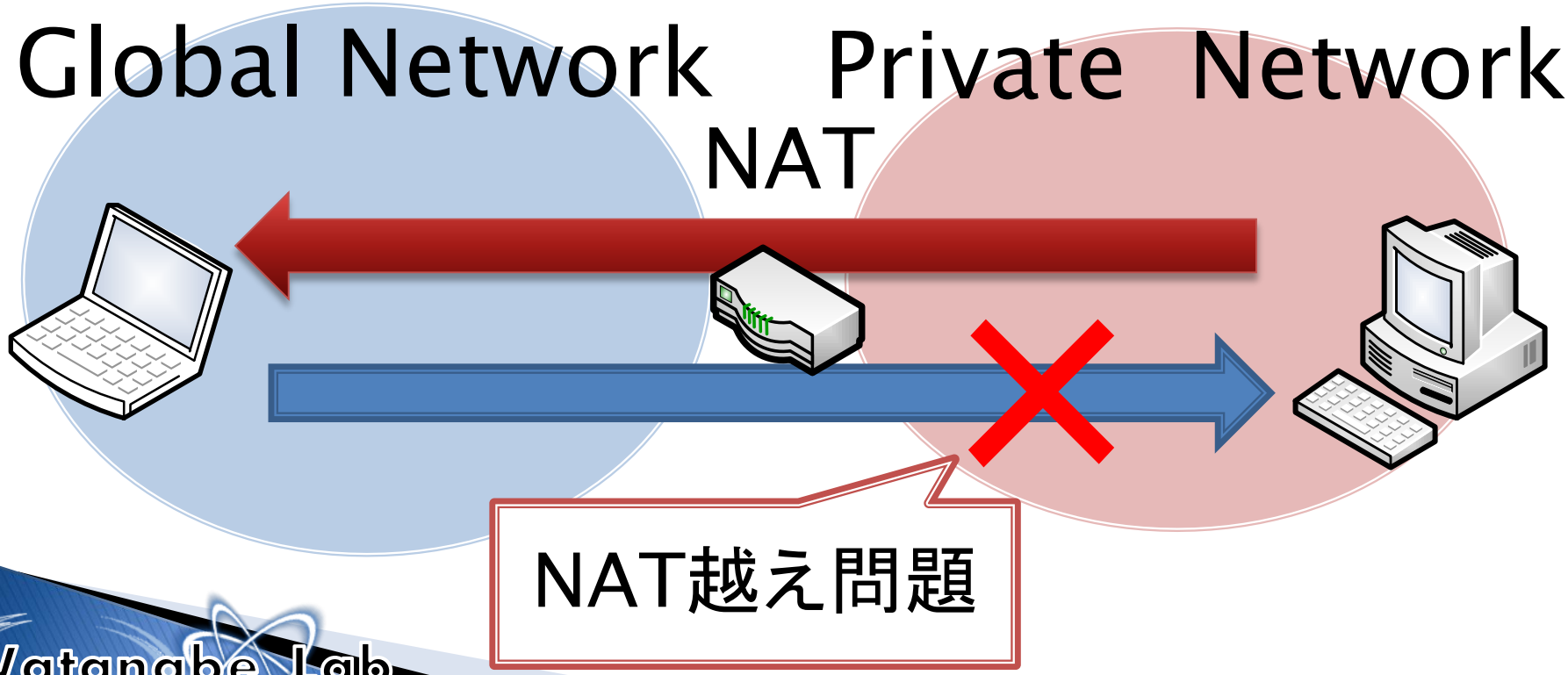
NAT越え問題

移動透過性が必要

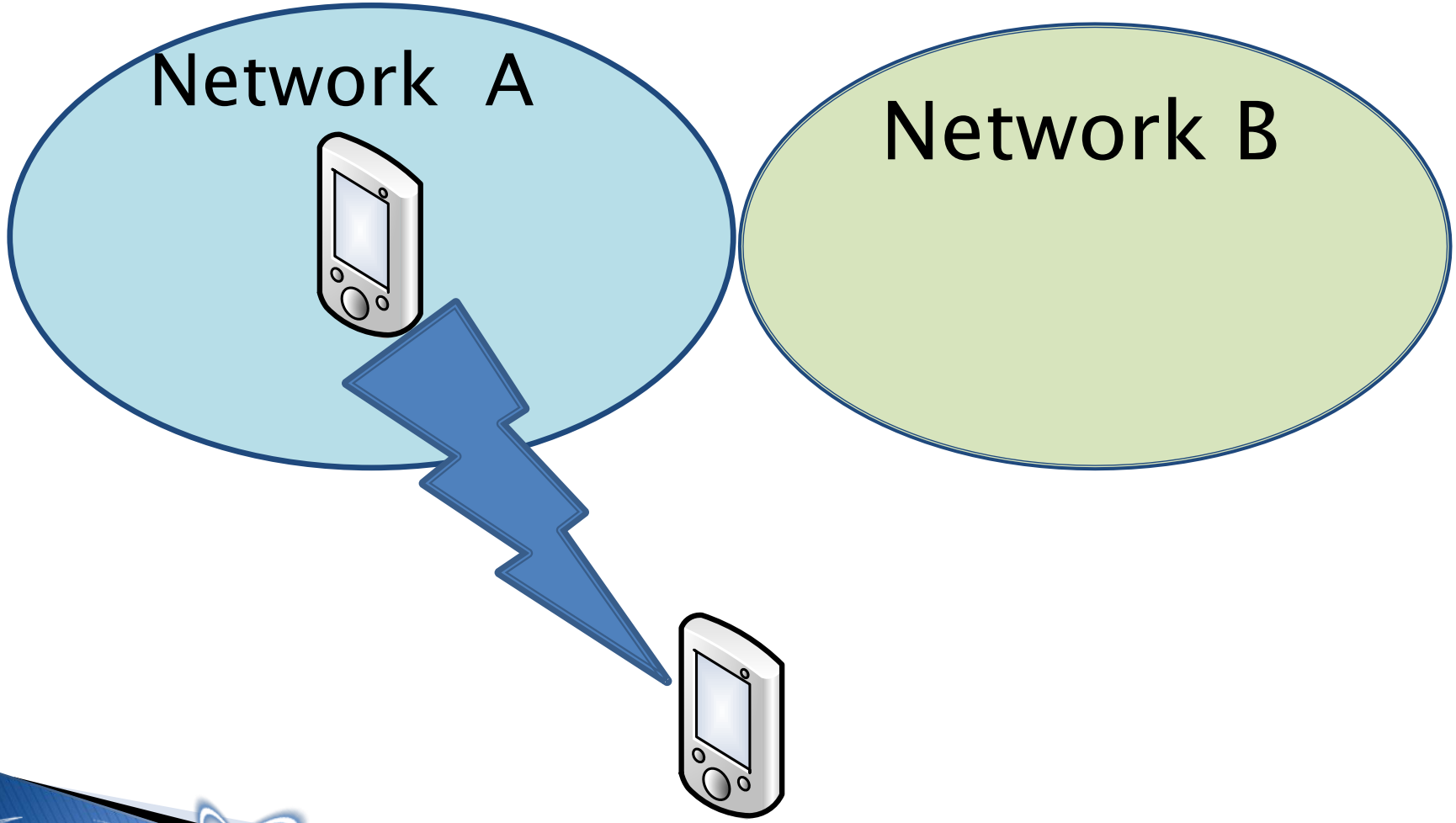
IPv4/IPv6の非互換性

NAT越え問題

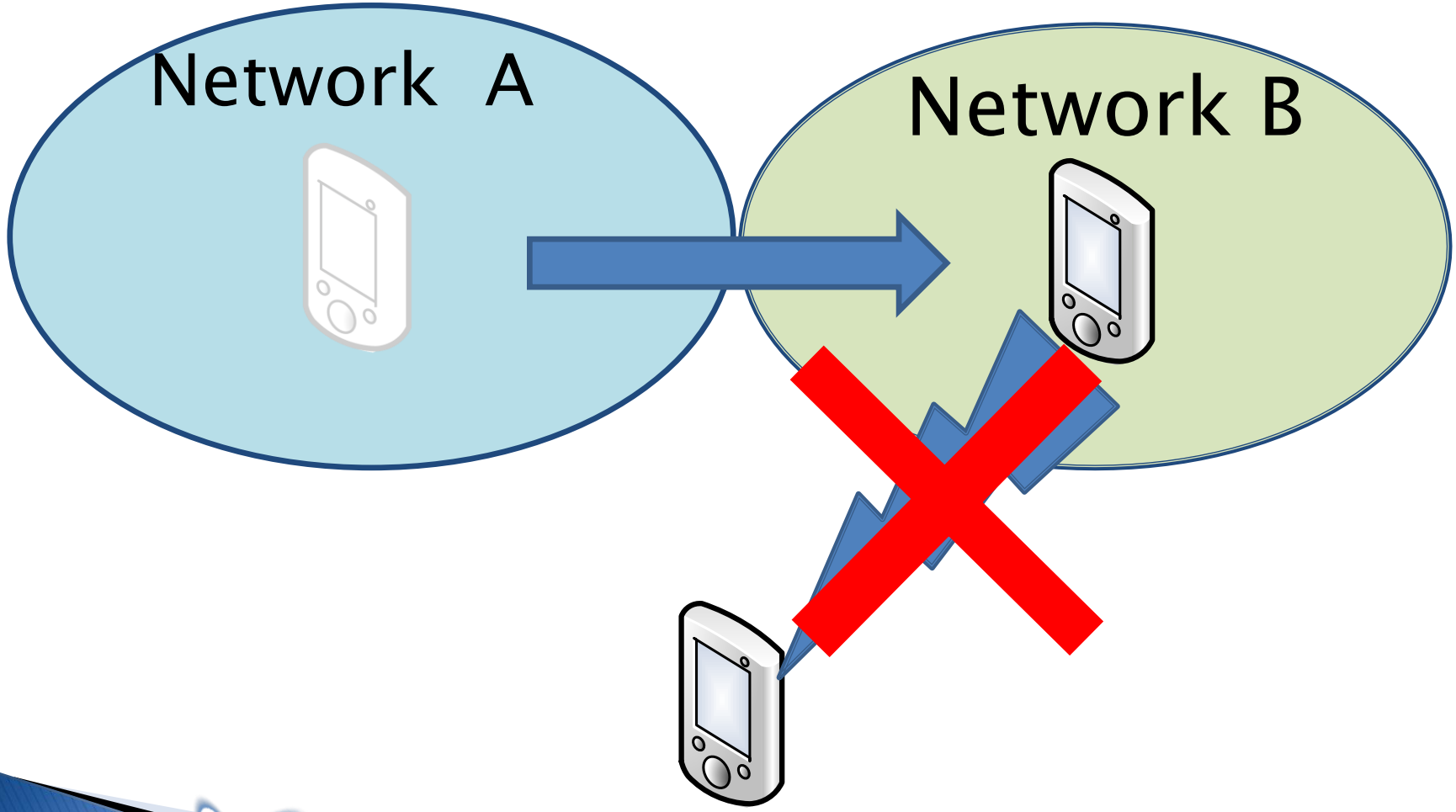
グローバル空間からプライベート空間への通信ができない



移動透過性が必要

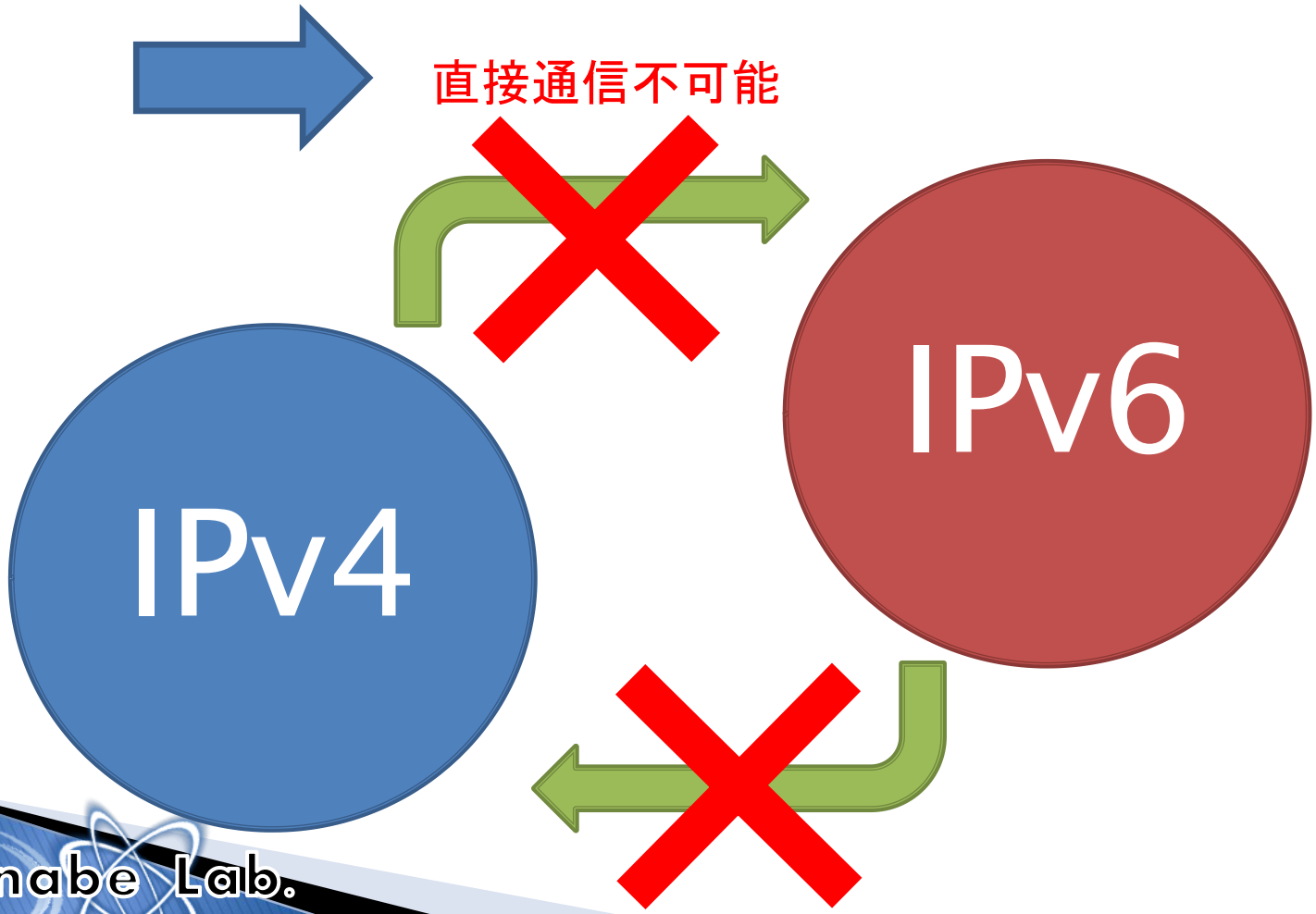


移動透過性が必要



IPv4/IPv6の非互換性

- アドレス空間が全く別のもの



NTMobile (Network Traversal with Mobility)

NAT越え問題

移動透過性が必要

IPv4/IPv6の非互換性

NTMobile（Network Traversal with Mobility）

NAT越え問題

移

3つの問題点を同時に解決！

IPv4/IPv6の非互換性

NTMobile (Network Traversal with Mobility)

NAT越え問題

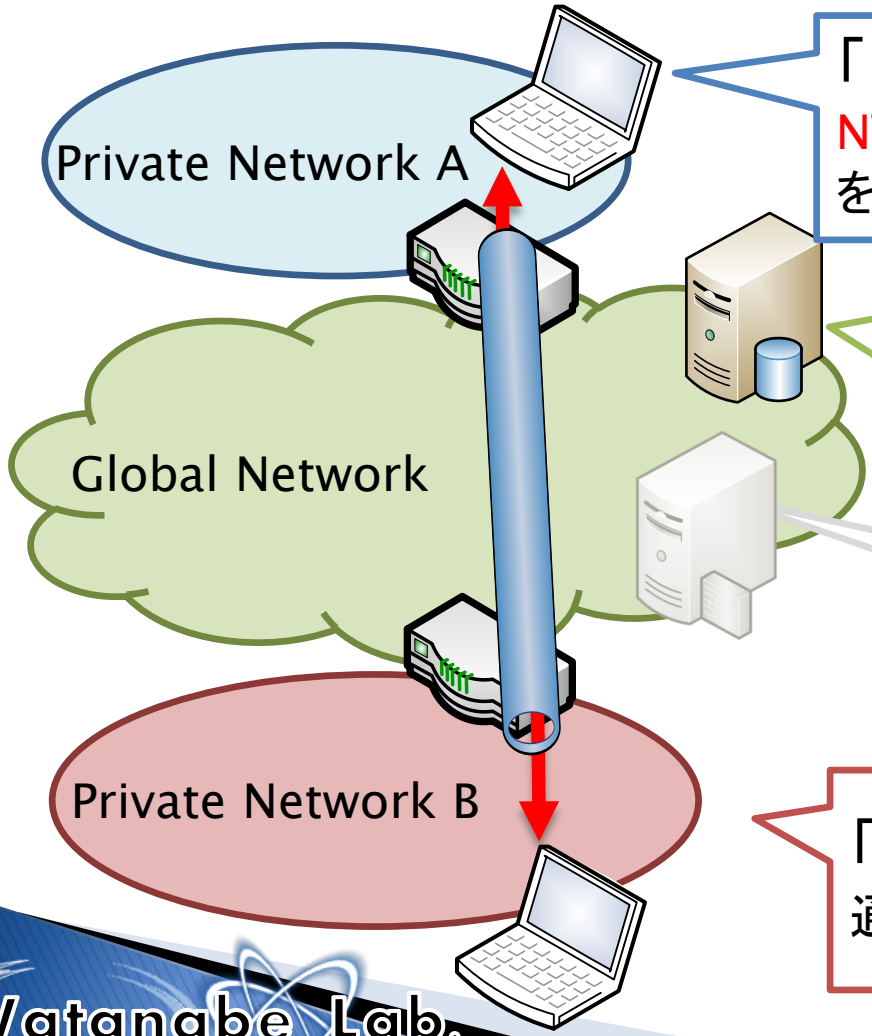
移

NTMobile
(Network Traversal with Mobility)

IPv4/IPv6の非互換性

NTMobile

- 移動透過性の実現とNAT越え問題の解決を同時に実現する技術
- NTM端末，DC(Direction Coordinator),RS(Relay Server)で構成



「NTM端末：MN」
NTMobileframework(NTMfw)
を実装した端末

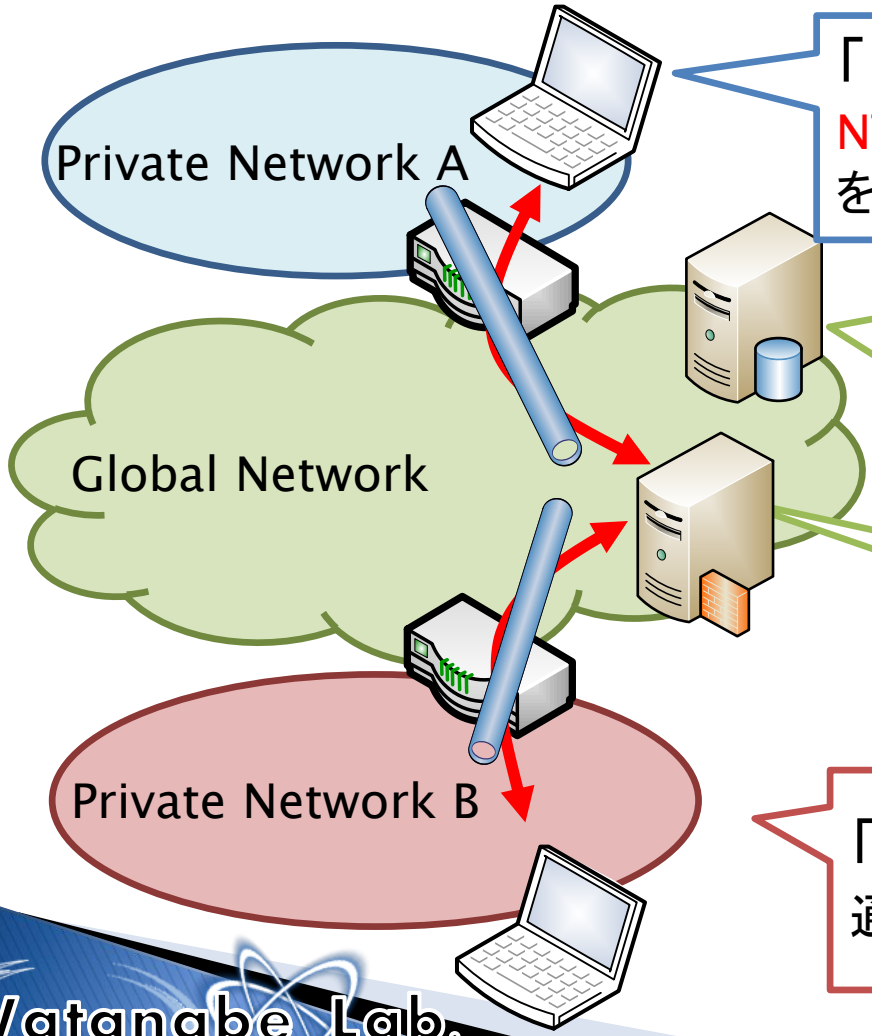
「DC」
・通信の経路指示
・仮想IPアドレスの配布

「RS」
・通信の中継

「NTM端末:CN」
通信相手

NTMobile

- 移動透過性の実現とNAT越え問題の解決を同時に実現する技術
- NTM端末，DC(Direction Coordinator),RS(Relay Server)で構成



「NTM端末：MN」
NTMobileframework(NTMfw)
を実装した端末

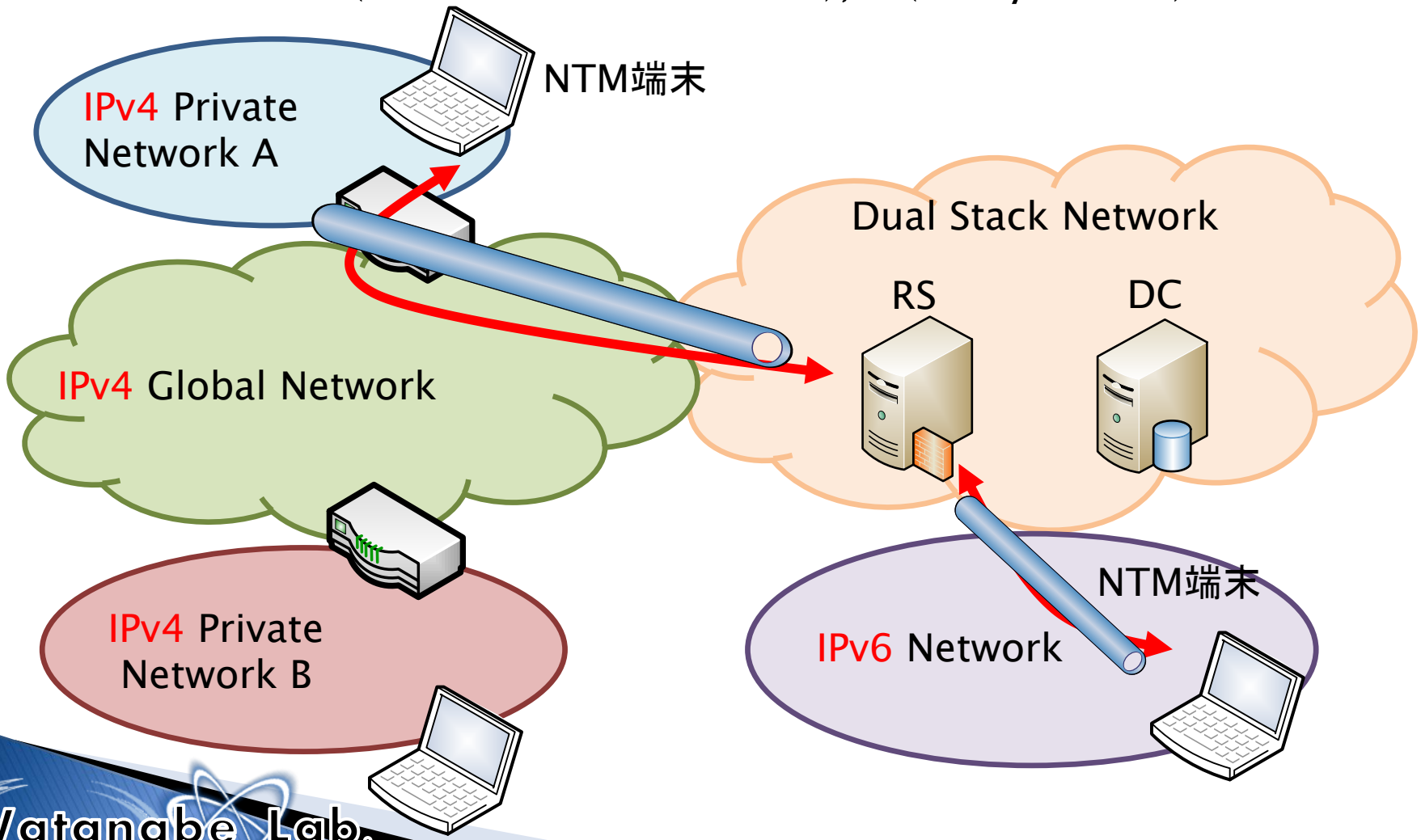
「DC」
・通信の経路指示
・仮想IPアドレスの配布

「RS」
・通信の中継

「NTM端末:CN」
通信相手

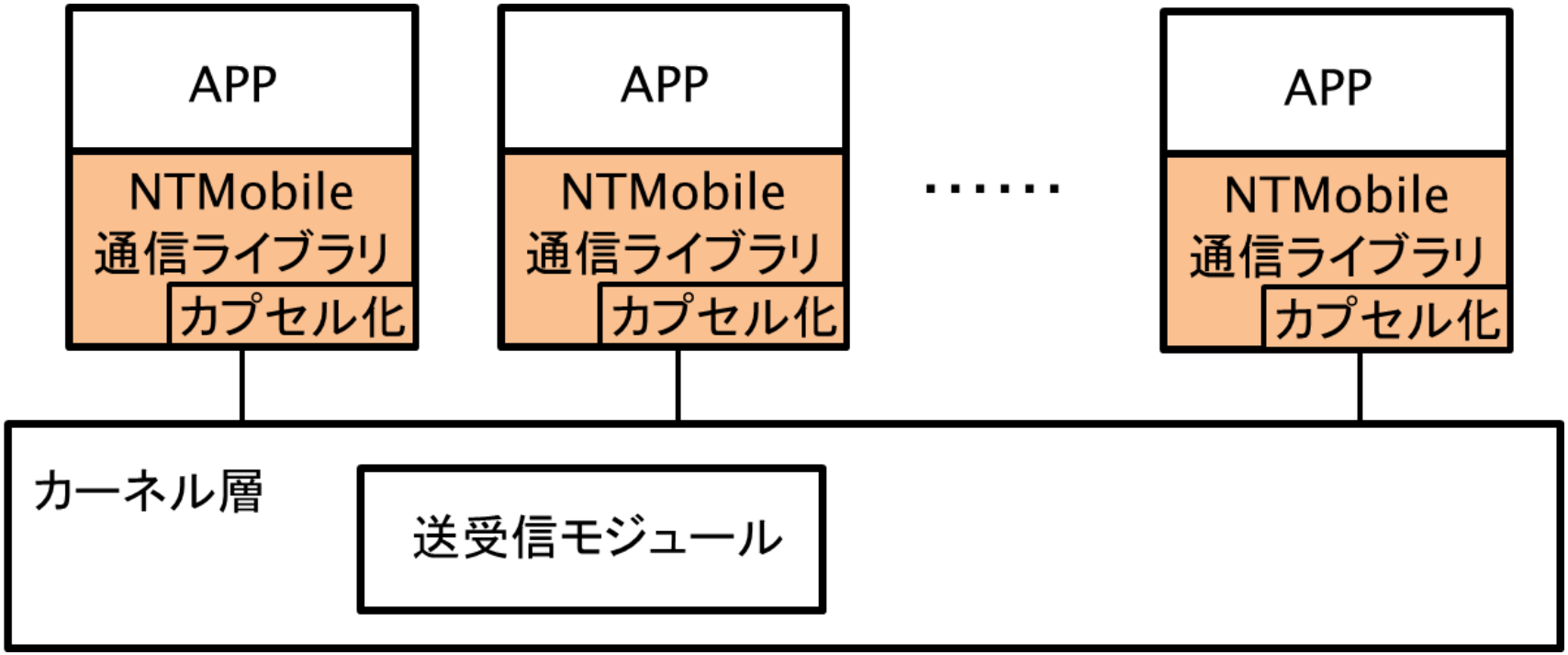
NTMobile

- 移動透過性の実現とNAT越え問題の解決を同時に実現する技術
- NTM端末，DC(Direction Coordinator),RS(Relay Server)で構成



NTMframework(NTMfw)

- NTMobileをアプリケーションライブラリとしてユーザに提供

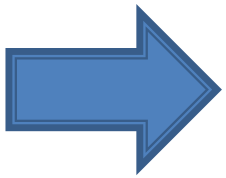


NTMfwの課題

- NTMobile非対応端末では利用できない
  - WindowsなどのOSで利用不可
- NTMfwを組み込めない場合に通信不可
  - 組込み型の家電
    - プログラムを書き換えられない
  - アプリケーションサーバー
    - 安定性を重視することから，新しい機能が追加できない
- アプリケーションがNTMobile通信を意識する必要あり

例） 既存アプリケーション

```
getaddrinfo( );
socket( );
sendto( );
```



例) NTMobileアプリケーション

```
ntmfw_getaddrinfo( );
ntmfw_socket( );
ntmfw_sendto( );
```

NTMfwの課題

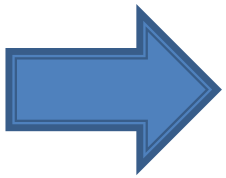
- NTMobile非対応端末では利用できない
  - WindowsなどのOSで利用不可
- NTMfwを組み込めない場合に通信不可
  - 組込み型の家電
    - プログラムを書き換えられない

NTMアダプタ(NTMA)の提案

- アプリケーションがNTMobile通信を意識する必要あり

例） 既存アプリケーション

```
getaddrinfo( );
socket( );
sendto( );
```

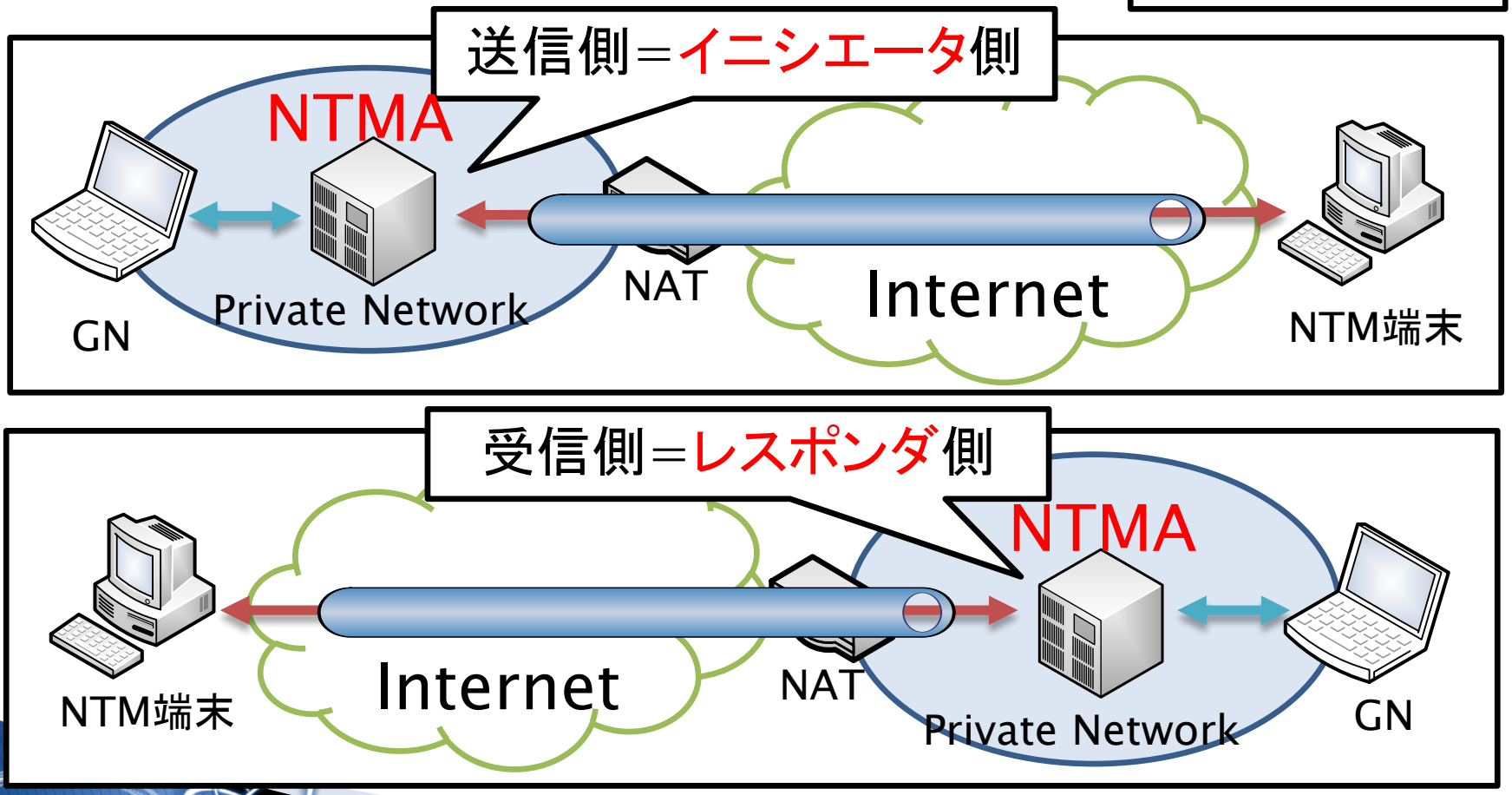


例) NTMobileアプリケーション

```
ntmfw_getaddrinfo( );
ntmfw_socket( );
ntmfw_sendto( );
```

NTMAとは

- 一般端末(GN:General Node)にNTMAを隣接設置
  - GNの通信をNTMobile通信に変換
  - GNのプログラムに手を加えず通信することができる

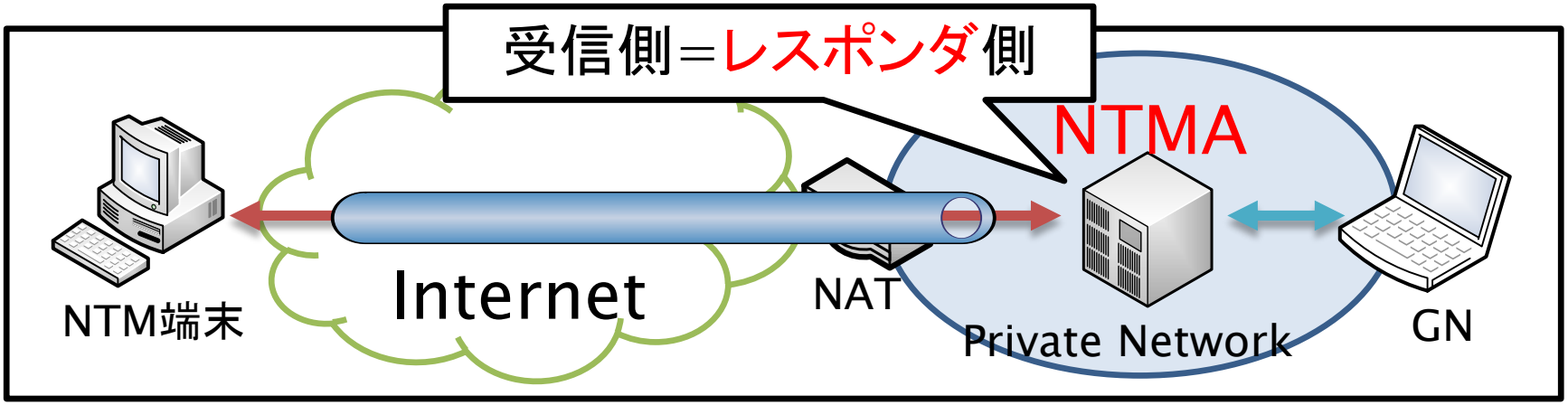


NTMAとは

- 一般端末(GN:General Node)にNTMAを隣接設置
  - GNの通信をNTMobile通信に変換
  - GNのプログラムに手を加えず通信することができる

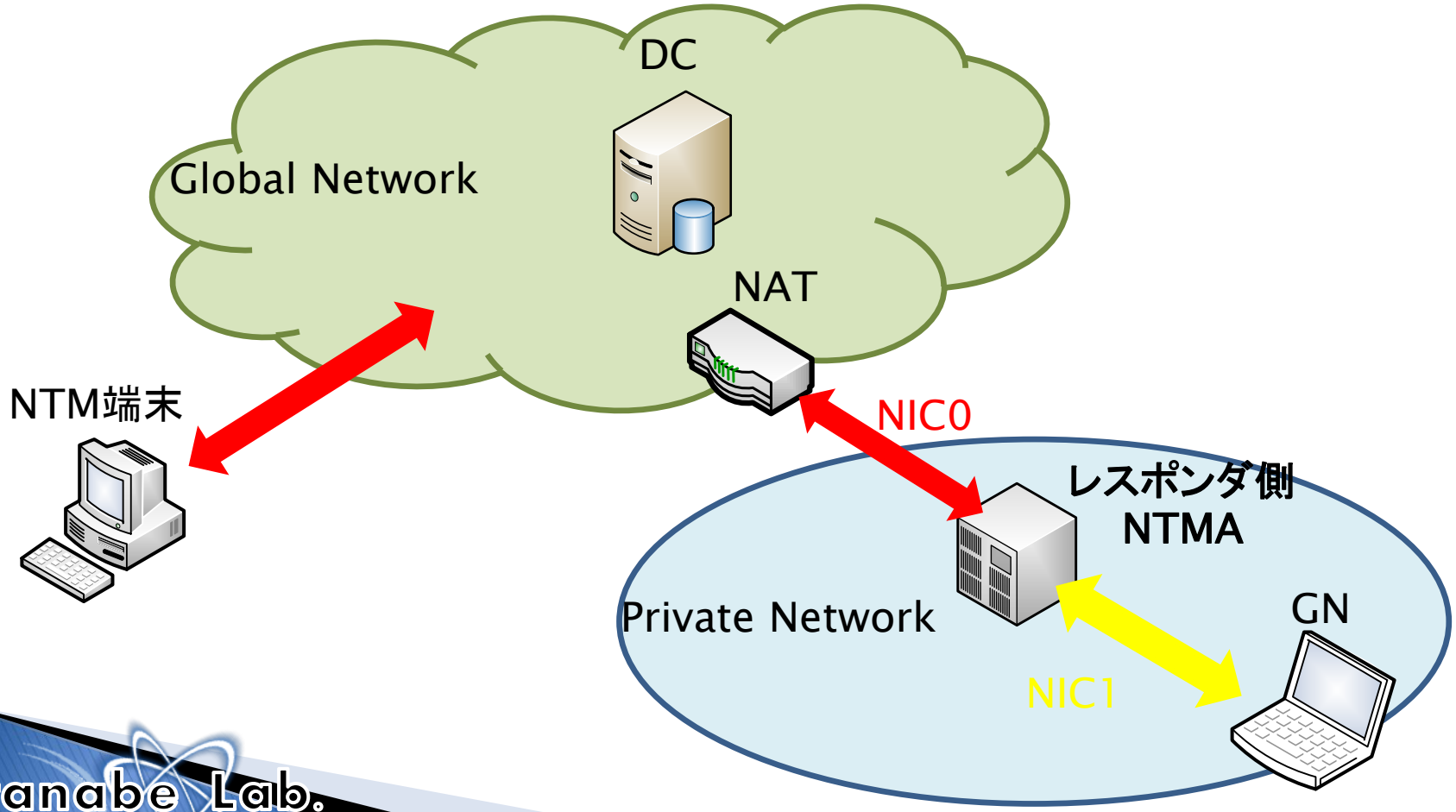
NTM通信：

一般通信：

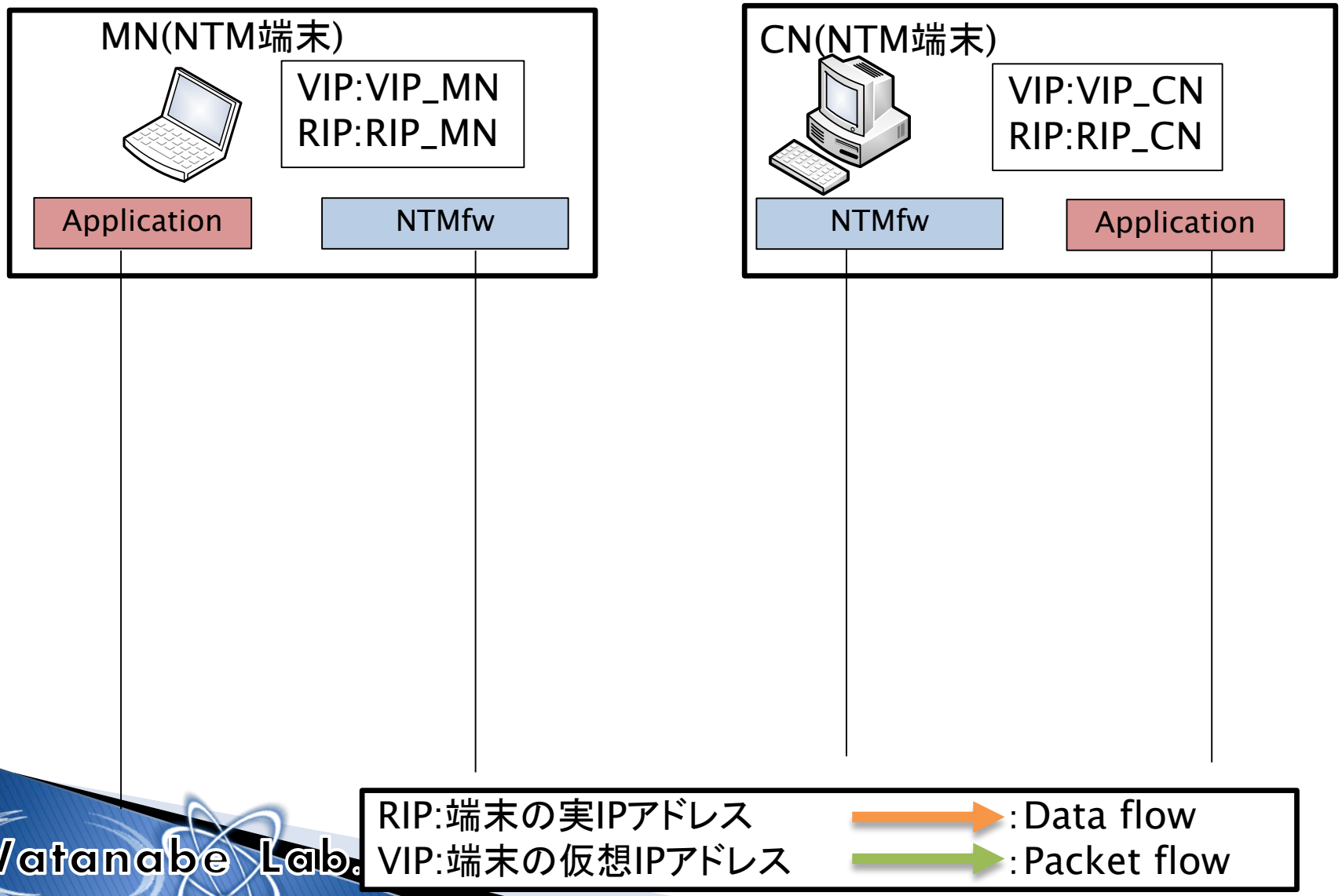


NTMAの構成

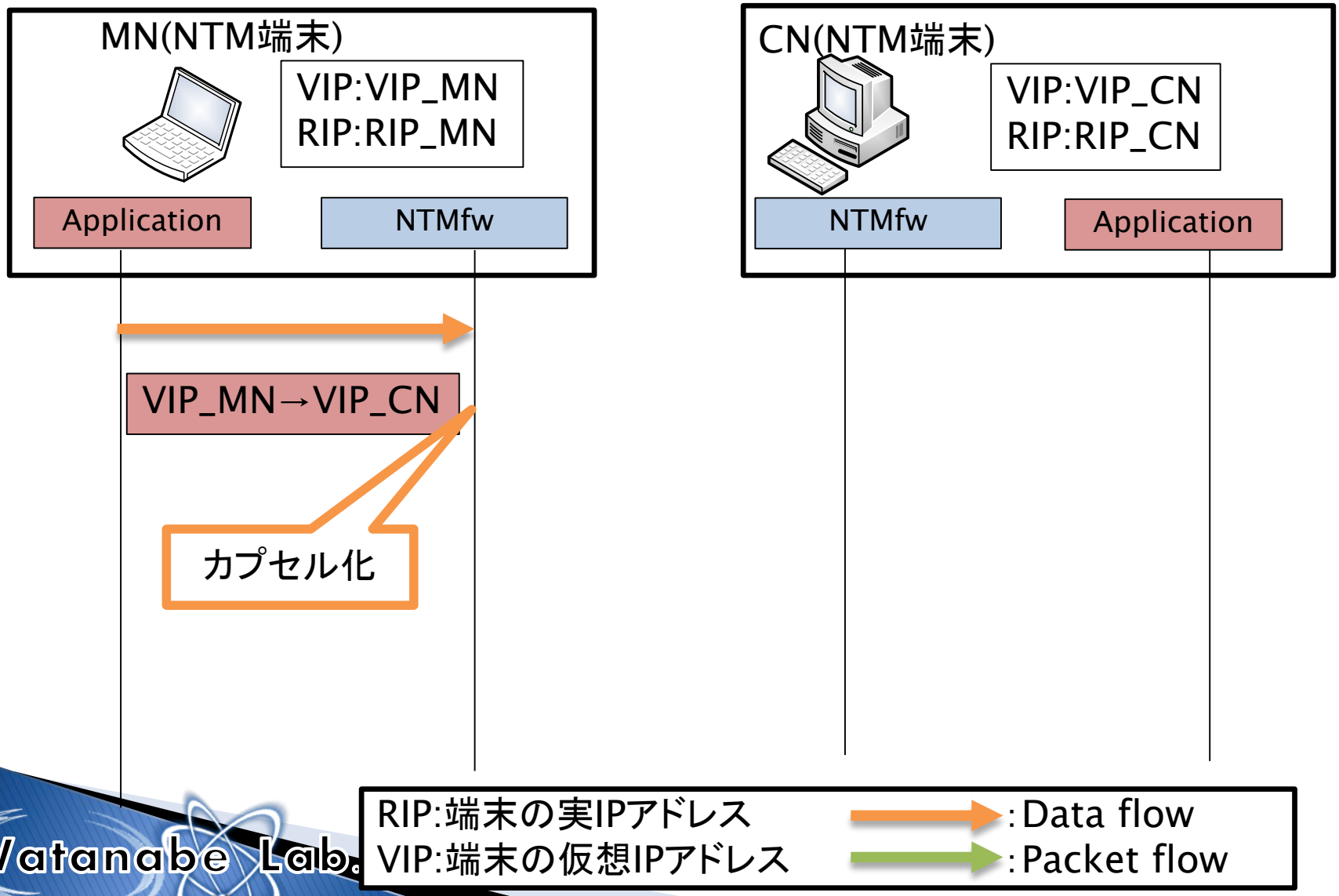
- NIC(Network Interface Card)を2枚用意
  - NIC0をインターネット側にブリッジ接続
  - NIC1をGNにブリッジ接続



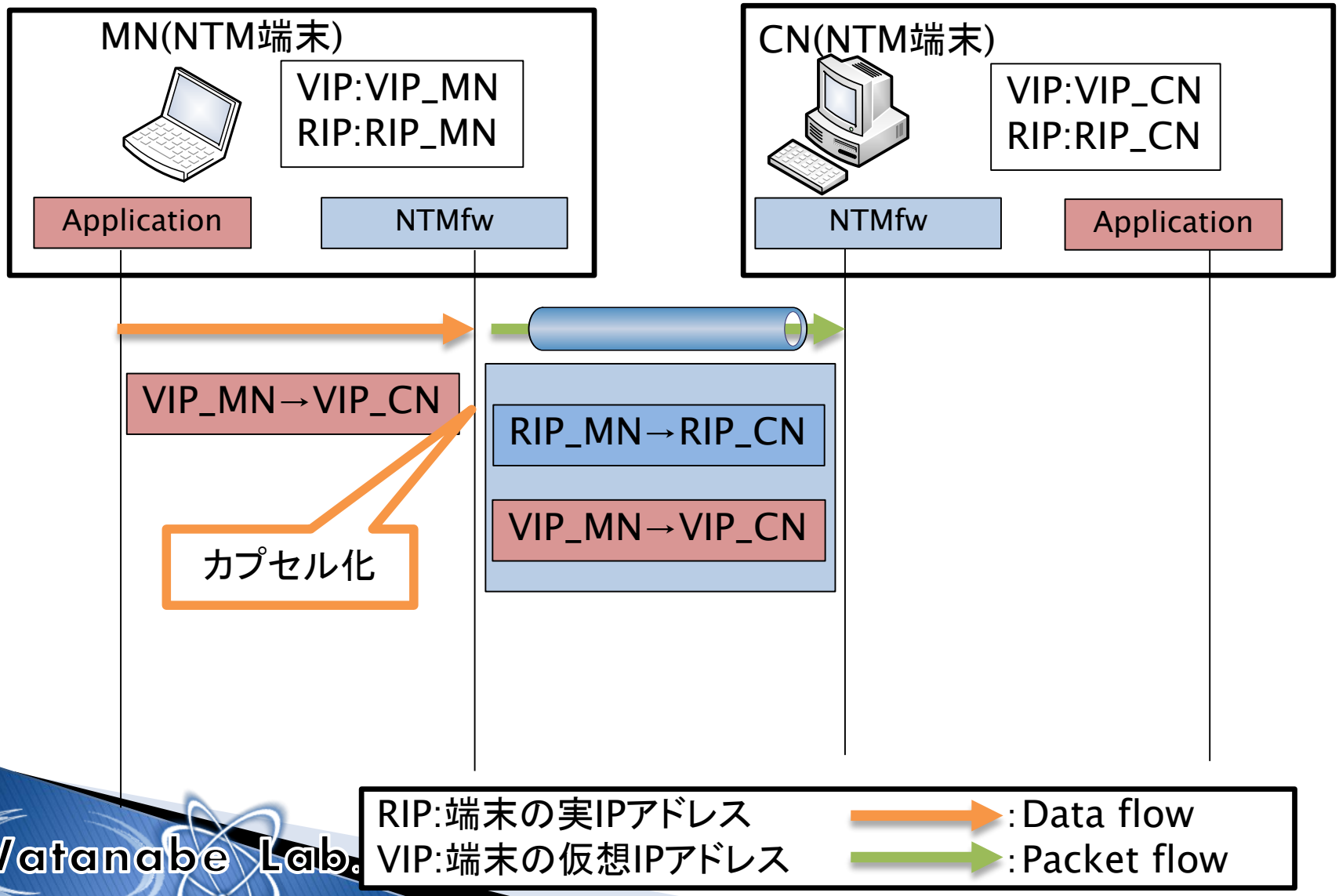
NTMobile通信


MN(NTM端末)
VIP:VIP_MN
RIP:RIP_MN
Application
NTMfw
CN(NTM端末)
VIP:VIP_CN
RIP:RIP_CN
NTMfw
Application
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

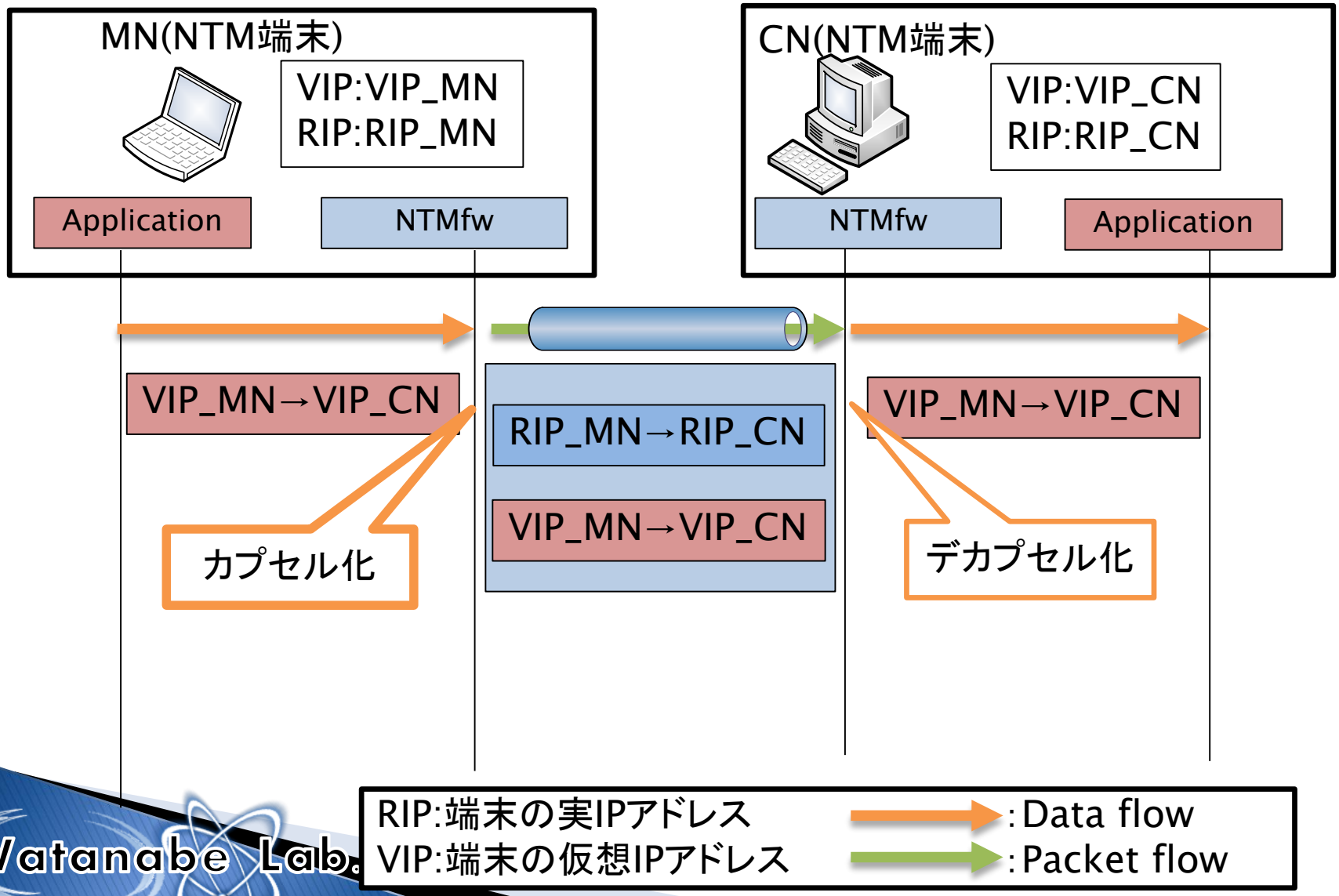
NTMobile通信


MN(NTM端末)
VIP:VIP_MN
RIP:RIP_MN
Application
NTMfw
CN(NTM端末)
VIP:VIP_CN
RIP:RIP_CN
NTMfw
Application
VIP_MN→VIP_CN
カプセル化
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

NTMobile通信


MN(NTM端末)
VIP:VIP_MN
RIP:RIP_MN
Application
NTMfw
CN(NTM端末)
VIP:VIP_CN
RIP:RIP_CN
NTMfw
Application
VIP_MN→VIP_CN
RIP_MN→RIP_CN
VIP_MN→VIP_CN
カプセル化
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

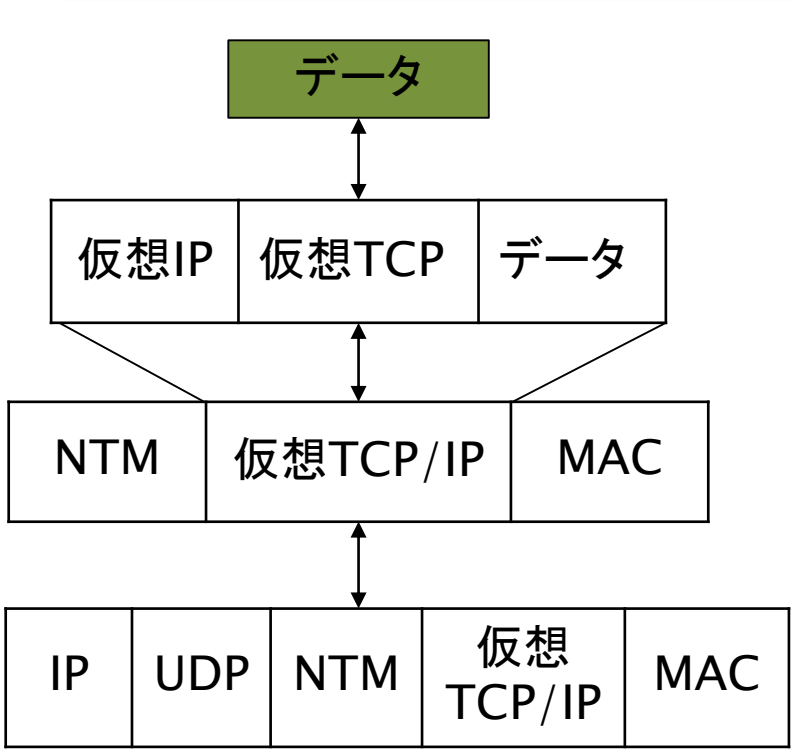
NTMobile通信

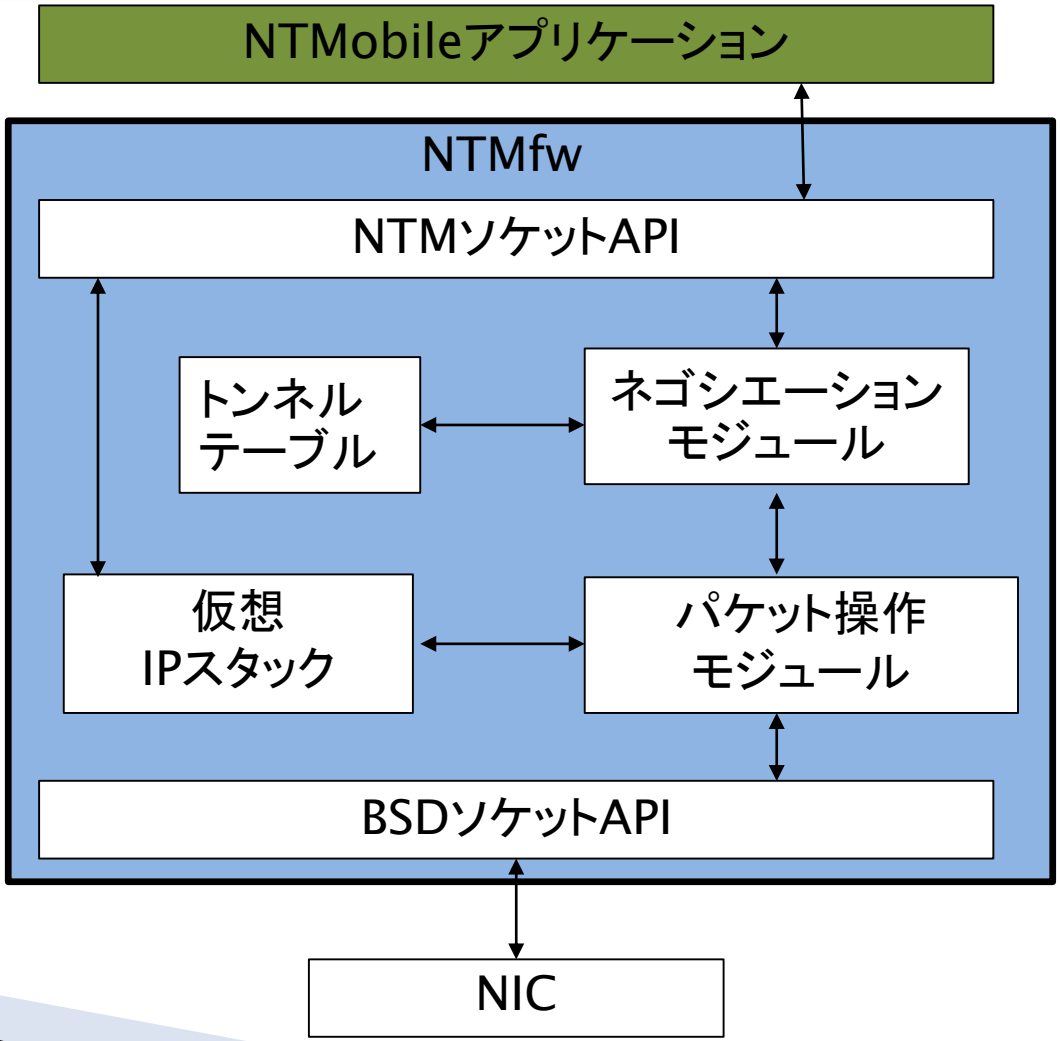

MN(NTM端末)
VIP:VIP_MN
RIP:RIP_MN
Application
NTMfw
CN(NTM端末)
VIP:VIP_CN
RIP:RIP_CN
NTMfw
Application
VIP_MN→VIP_CN
RIP_MN→RIP_CN
VIP_MN→VIP_CN
VIP_MN→VIP_CN
カプセル化
デカプセル化
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

NTMfw

NTMobileアプリケーション

・NTMソケットAPIで記述されたアプリケーション

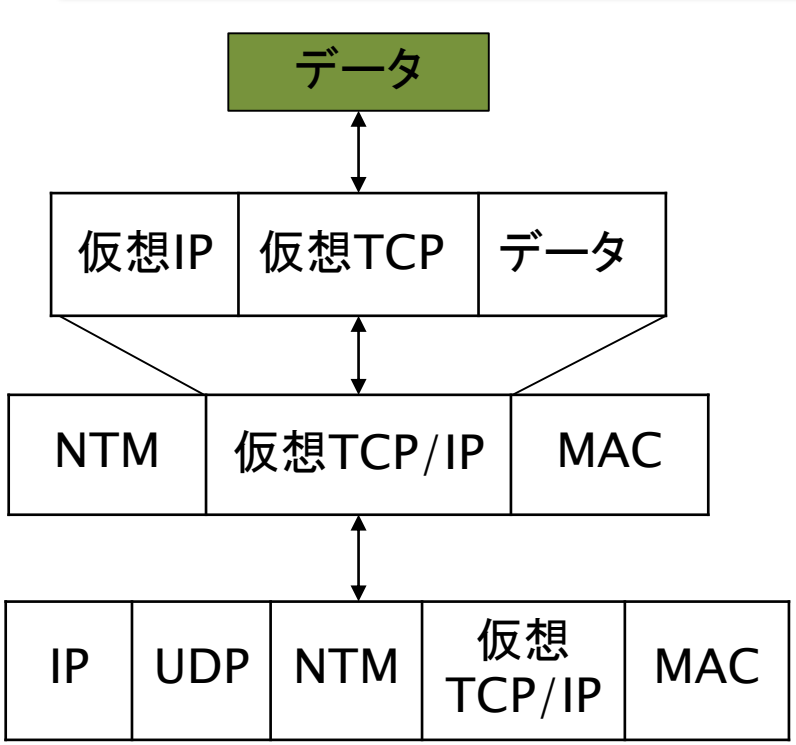


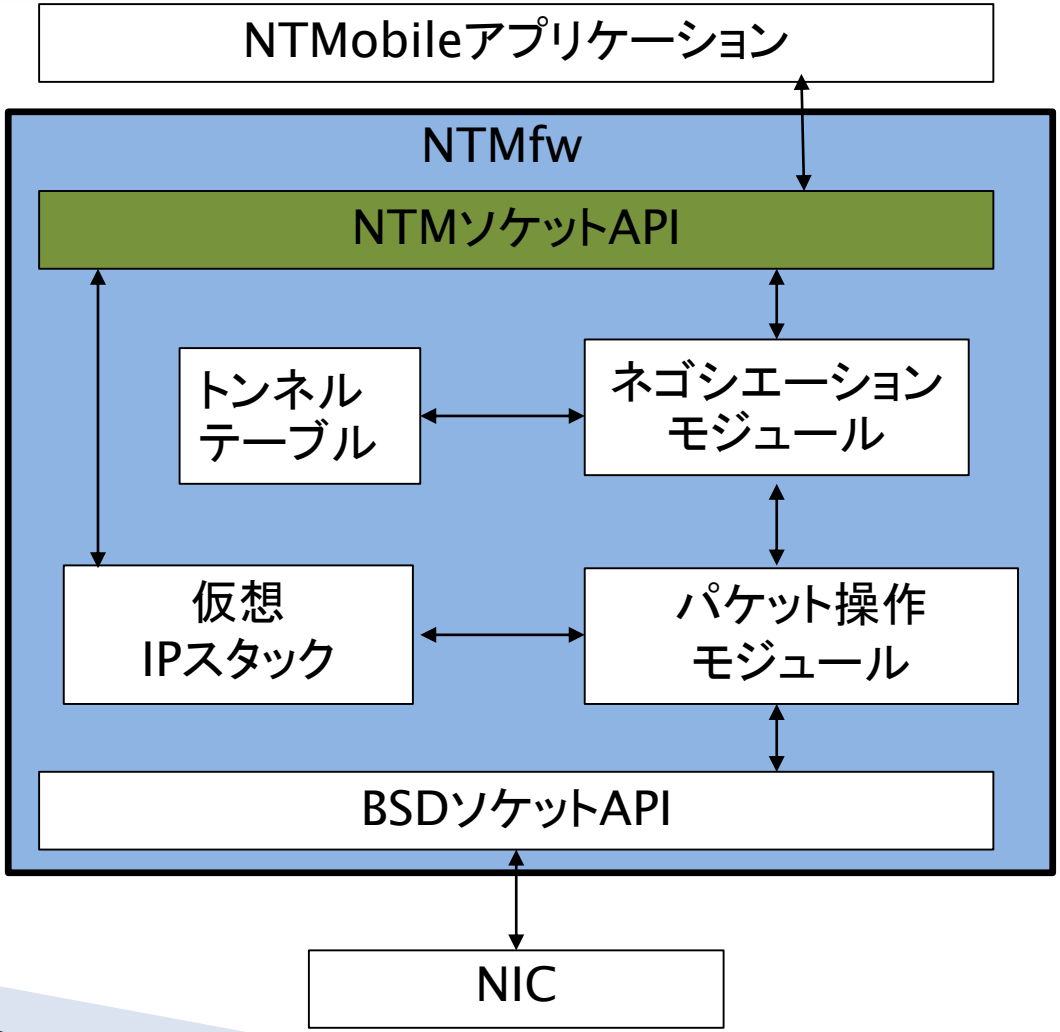


NTMfw

NTMソケットAPI

・BSDソケットAPI互換のソケットAPI
・アプリケーションデータを仮想IPスタックへ渡す

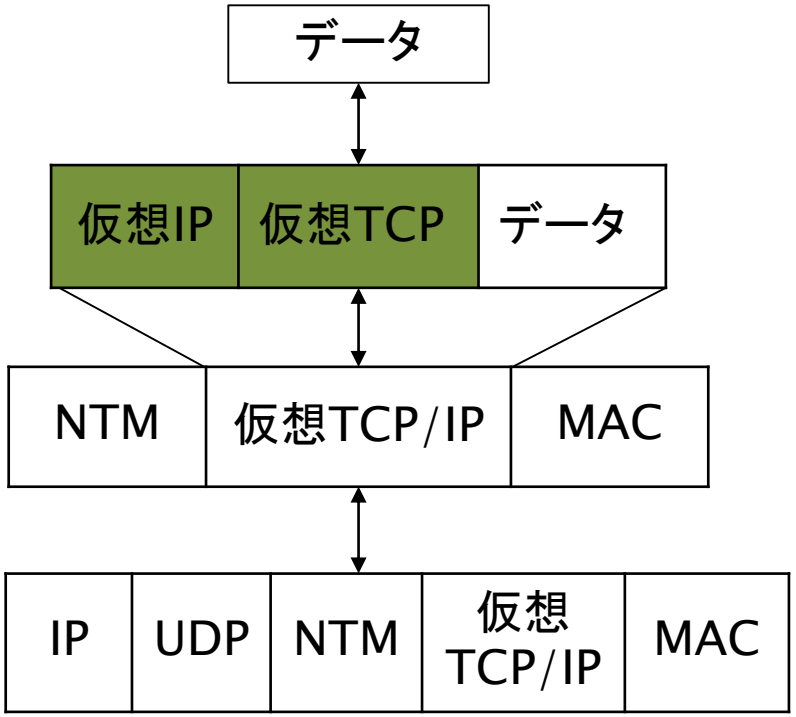


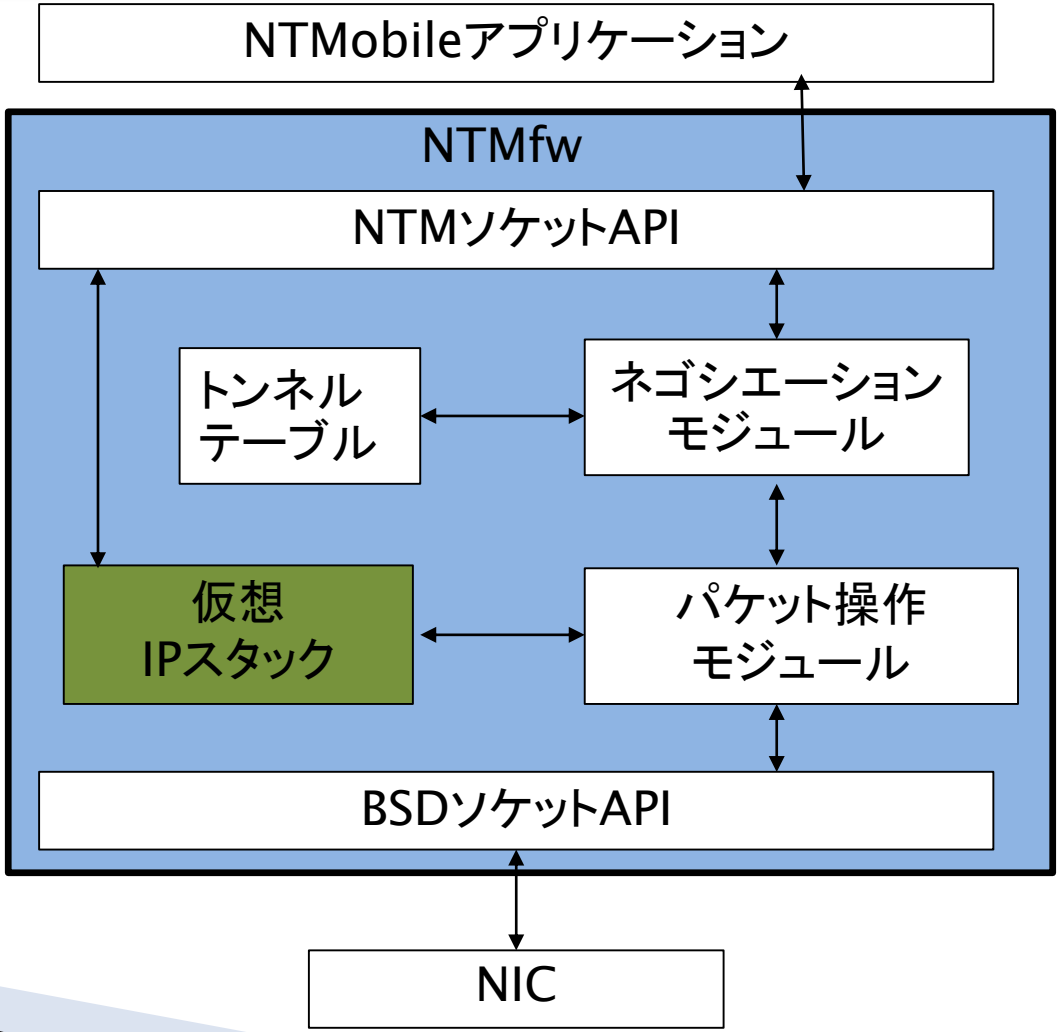


NTMfw

仮想IPスタック

・アプリケーションデータから仮想TCP/IPパケットを生成

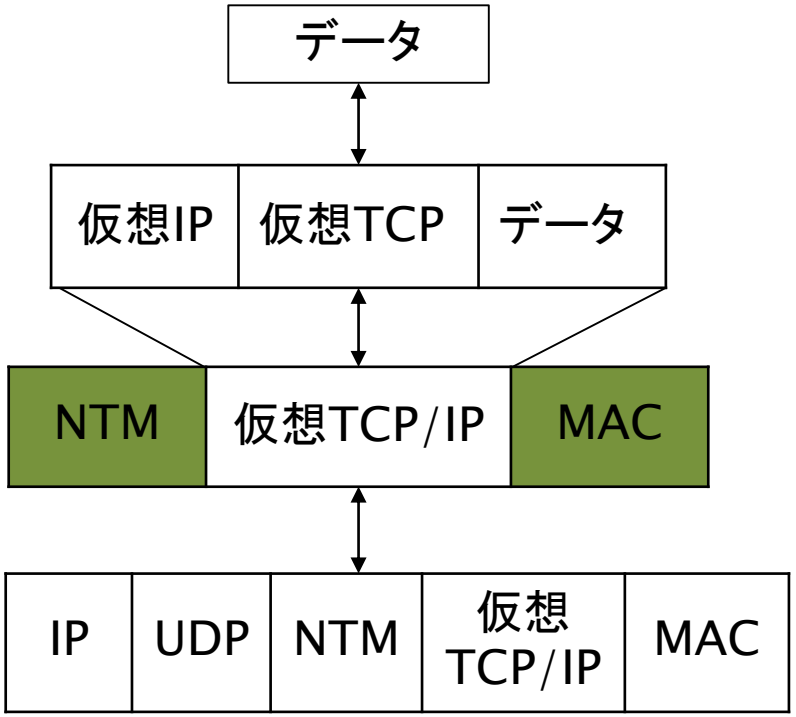


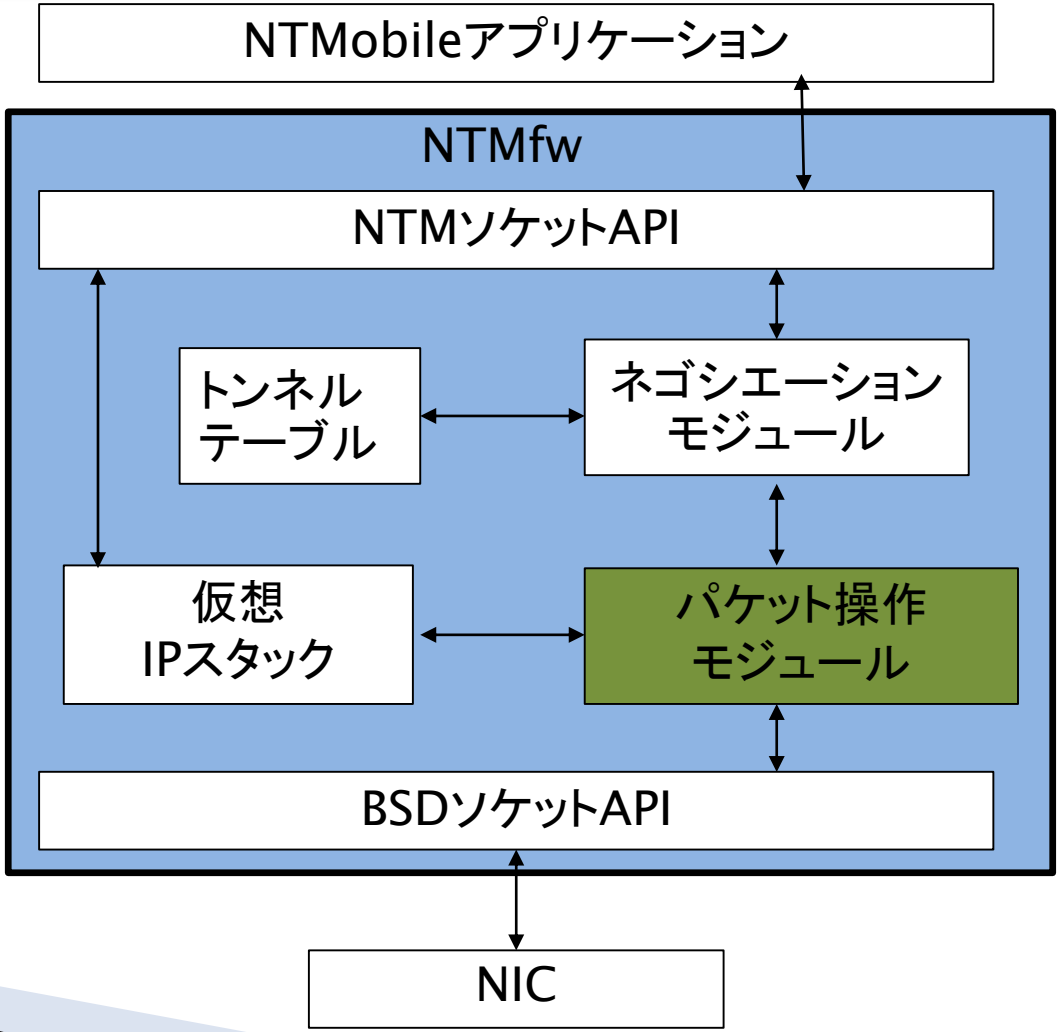


NTMfw

パケット操作モジュール

・NTMヘッダの付与
・MAC付与/検証等

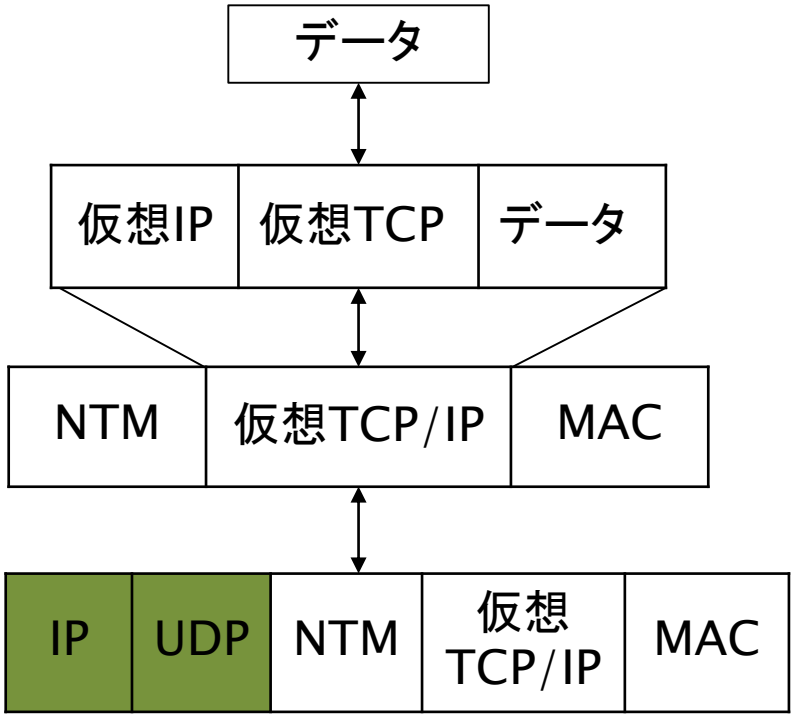


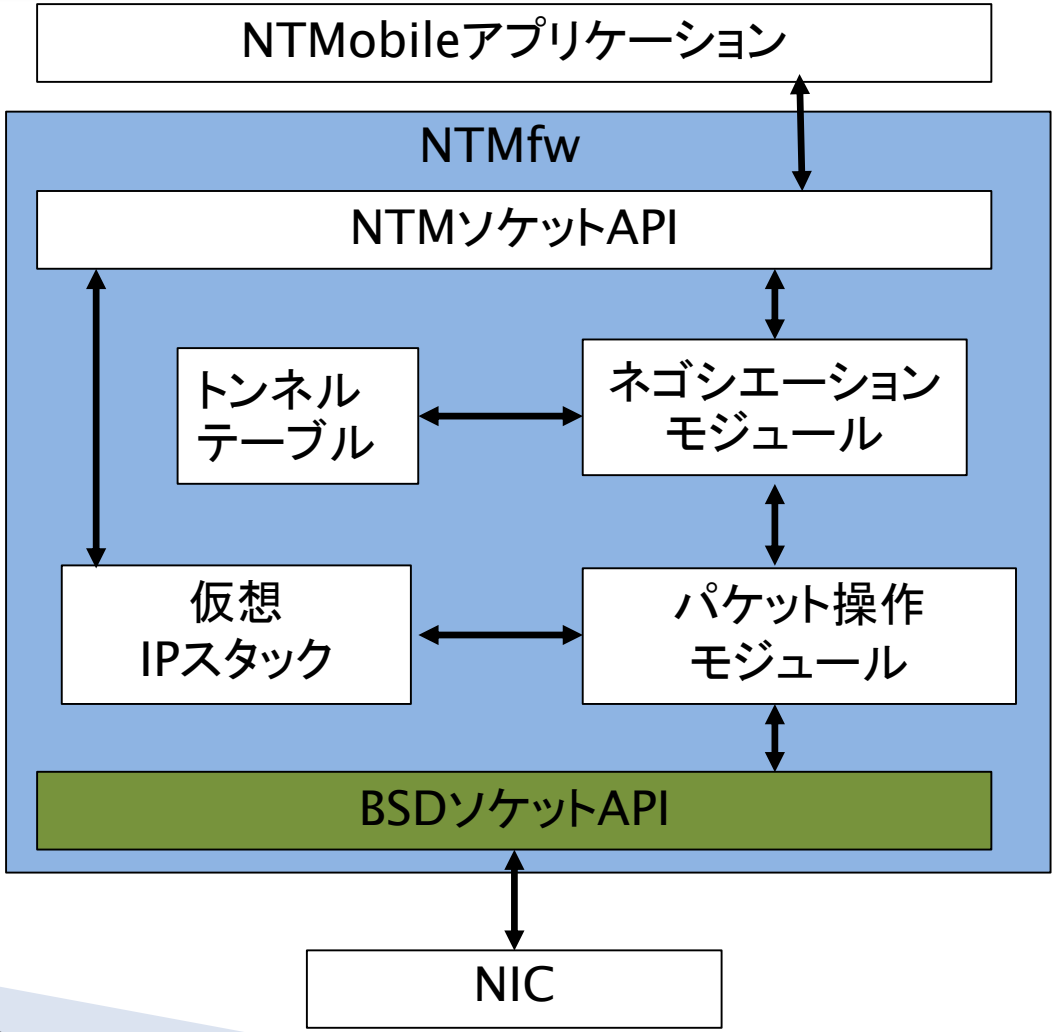


NTMfw

NTMソケットAPI

・C言語のソケットAPI
・データパケットの送受信（カプセル化/デカプセル化）



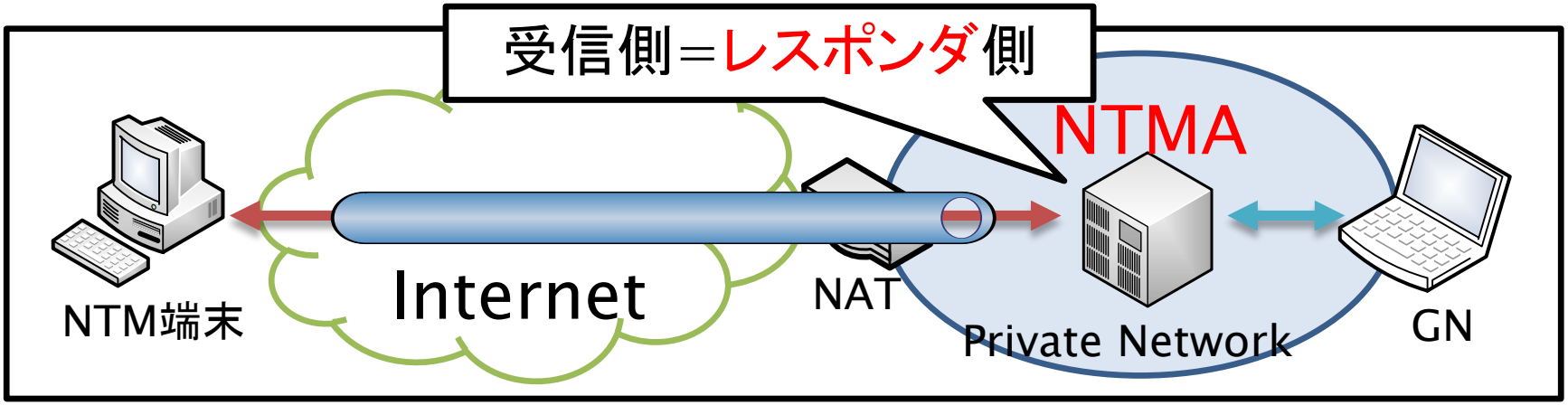


NTMAとは

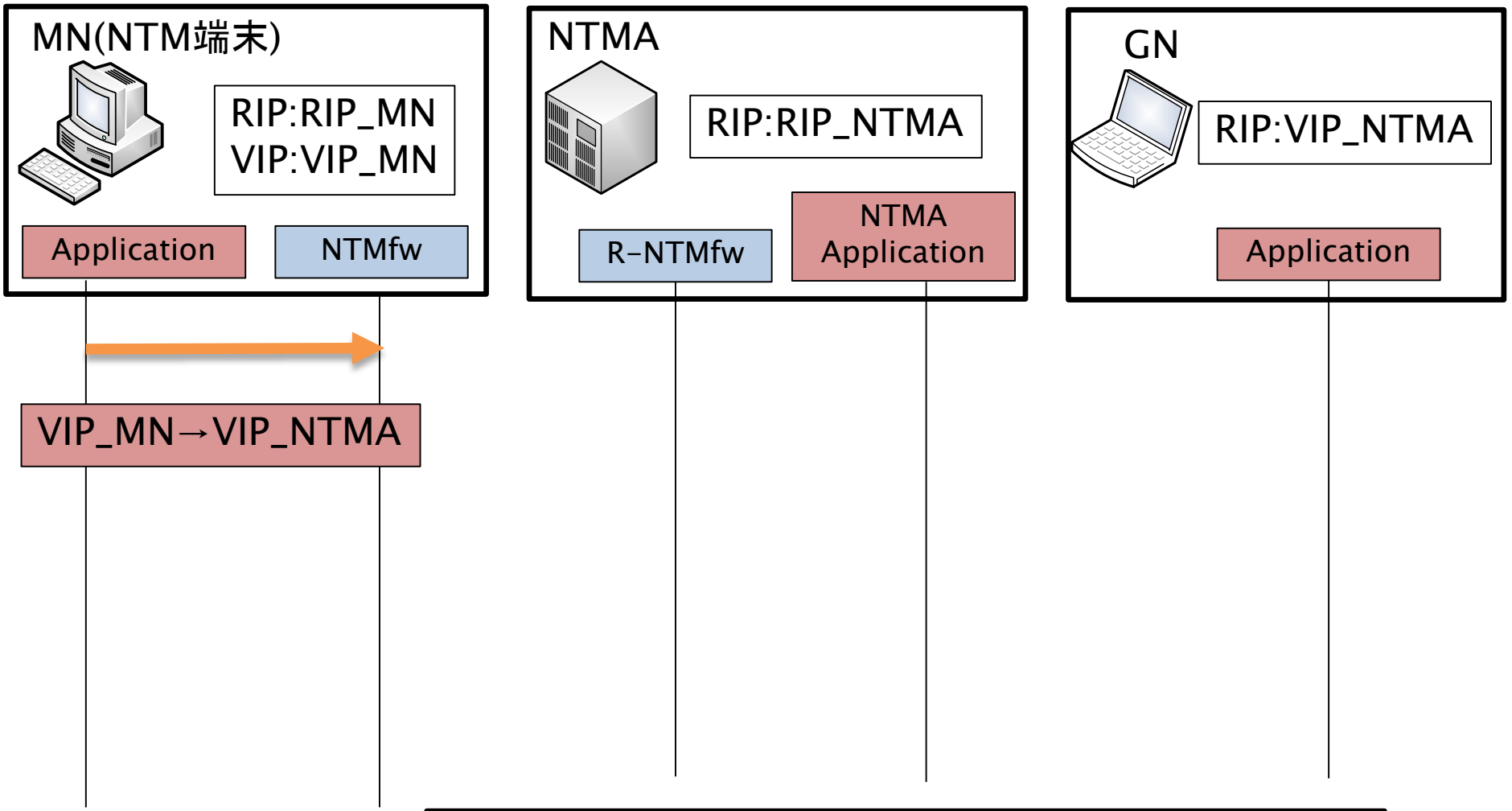
- 一般端末(GN:General Node)にNTMAを隣接設置
  - GNの通信をNTMobile通信に変換
  - GNのプログラムに手を加えず通信することができる

NTM通信：

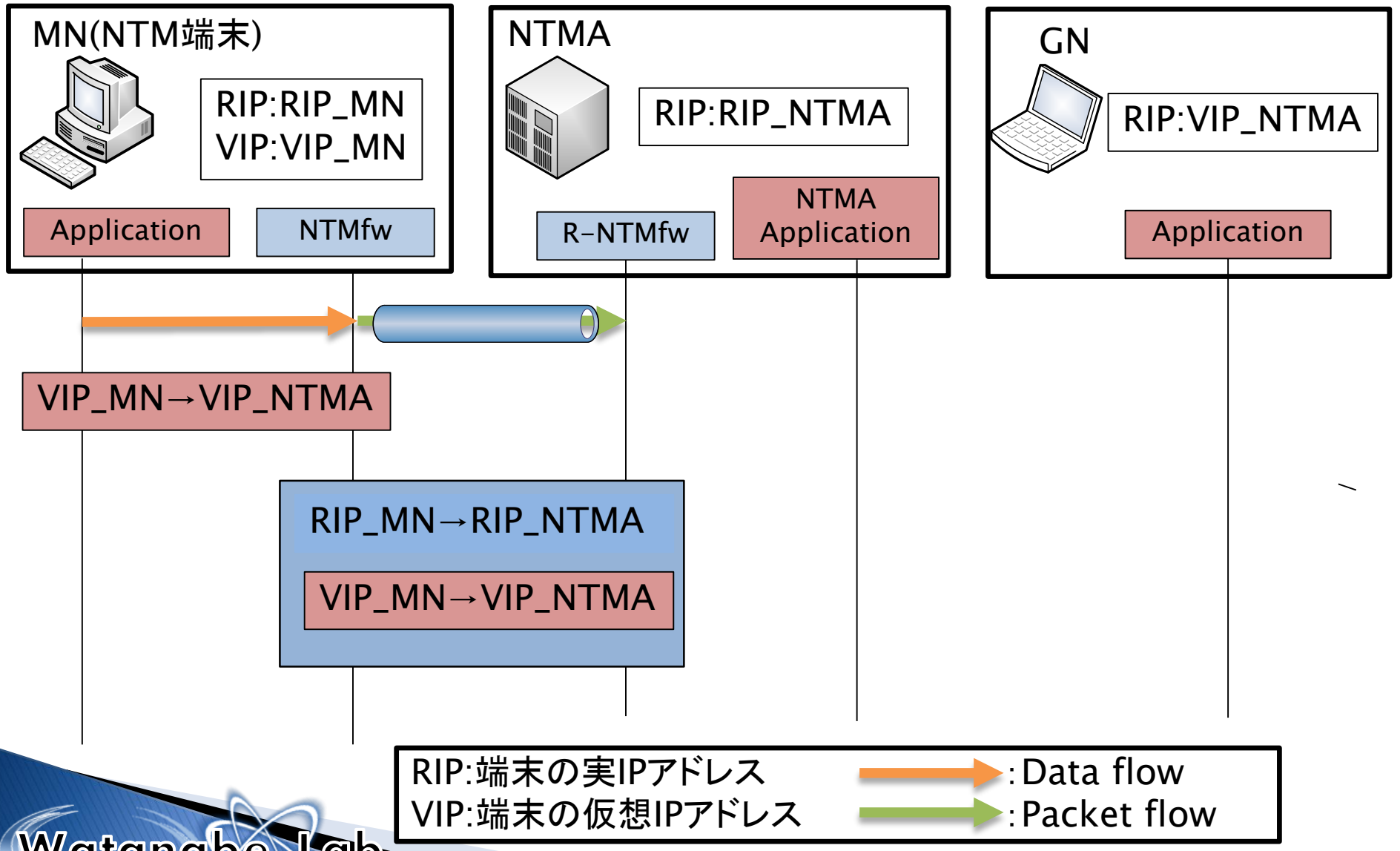
一般通信：



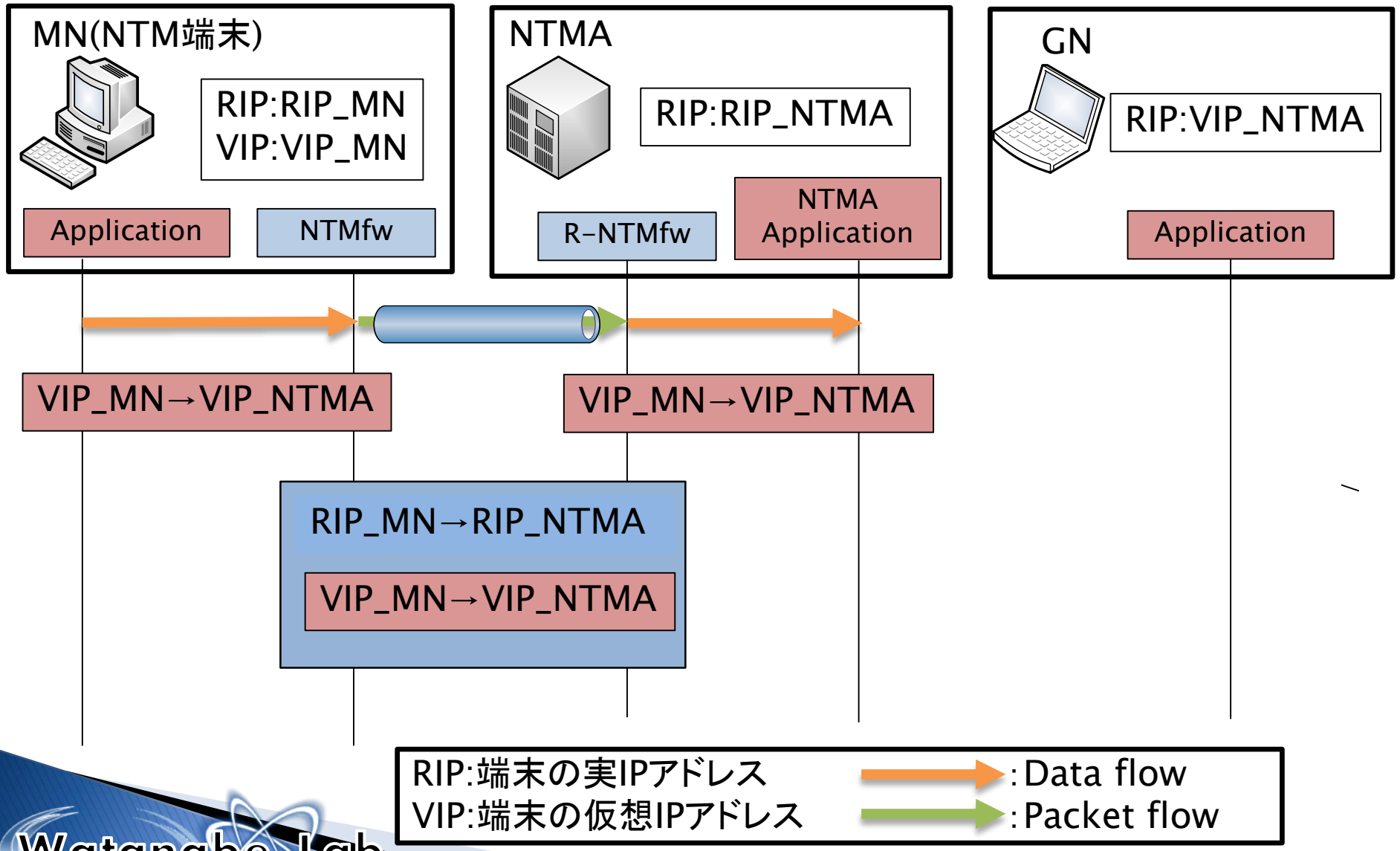
NTMAの通信


MN(NTM端末)
RIP:RIP_MN
VIP:VIP_MN
Application
NTMfw
NTMA
RIP:RIP_NTMA
R-NTMfw
NTMA Application
GN
RIP:VIP_NTMA
Application
VIP_MN→VIP_NTMA
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

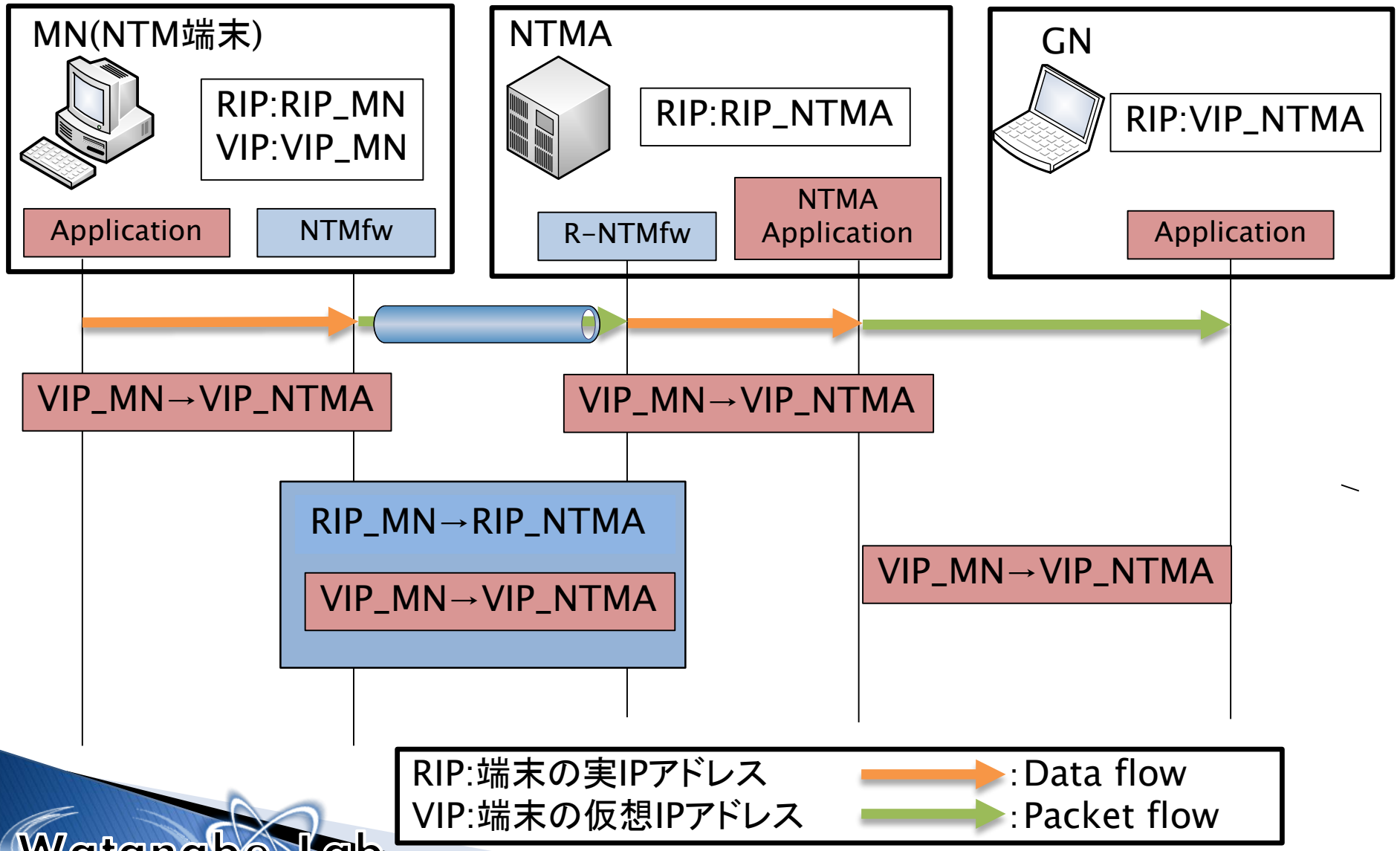
NTMAの通信


MN(NTM端末)
RIP:RIP_MN
VIP:VIP_MN
Application
NTMfw
NTMA
RIP:RIP_NTMA
R-NTMfw
NTMA Application
GN
RIP:VIP_NTMA
Application
VIP_MN→VIP_NTMA
RIP_MN→RIP_NTMA
VIP_MN→VIP_NTMA
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

NTMAの通信


MN(NTM端末)
RIP:RIP_MN
VIP:VIP_MN
Application
NTMfw
NTMA
RIP:RIP_NTMA
R-NTMfw
NTMA Application
GN
RIP:VIP_NTMA
Application
VIP_MN→VIP_NTMA
VIP_MN→VIP_NTMA
RIP_MN→RIP_NTMA
VIP_MN→VIP_NTMA
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

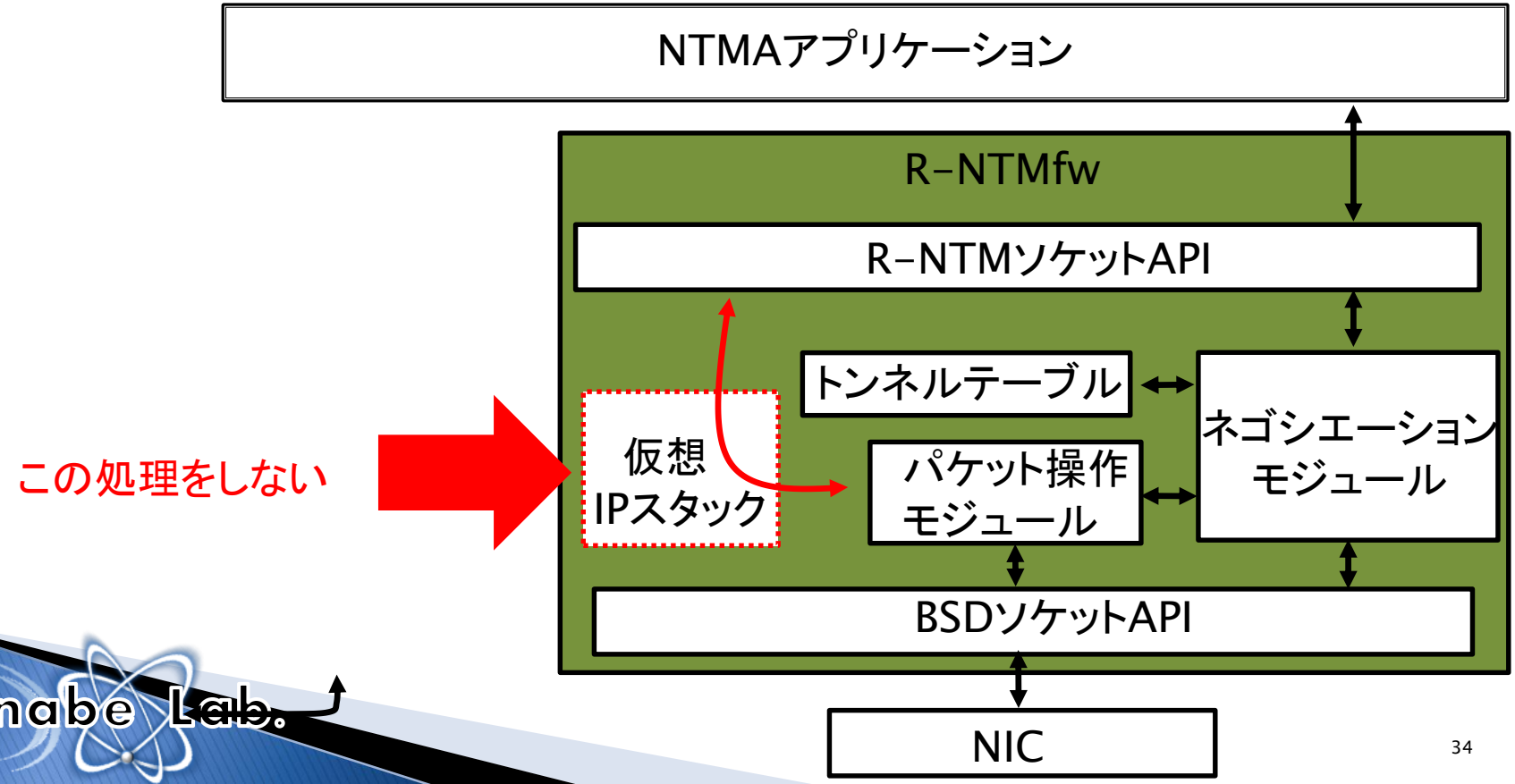
NTMAの通信


MN(NTM端末)
RIP:RIP_MN
VIP:VIP_MN
Application
NTMfw
NTMA
RIP:RIP_NTMA
R-NTMfw
NTMA Application
GN
RIP:VIP_NTMA
Application
VIP_MN→VIP_NTMA
VIP_MN→VIP_NTMA
RIP_MN→RIP_NTMA
VIP_MN→VIP_NTMA
VIP_MN→VIP_NTMA
RIP:端末の実IPアドレス
VIP:端末の仮想IPアドレス
:Data flow
:Packet flow

NTMAのモジュール構成

R-NTMfw

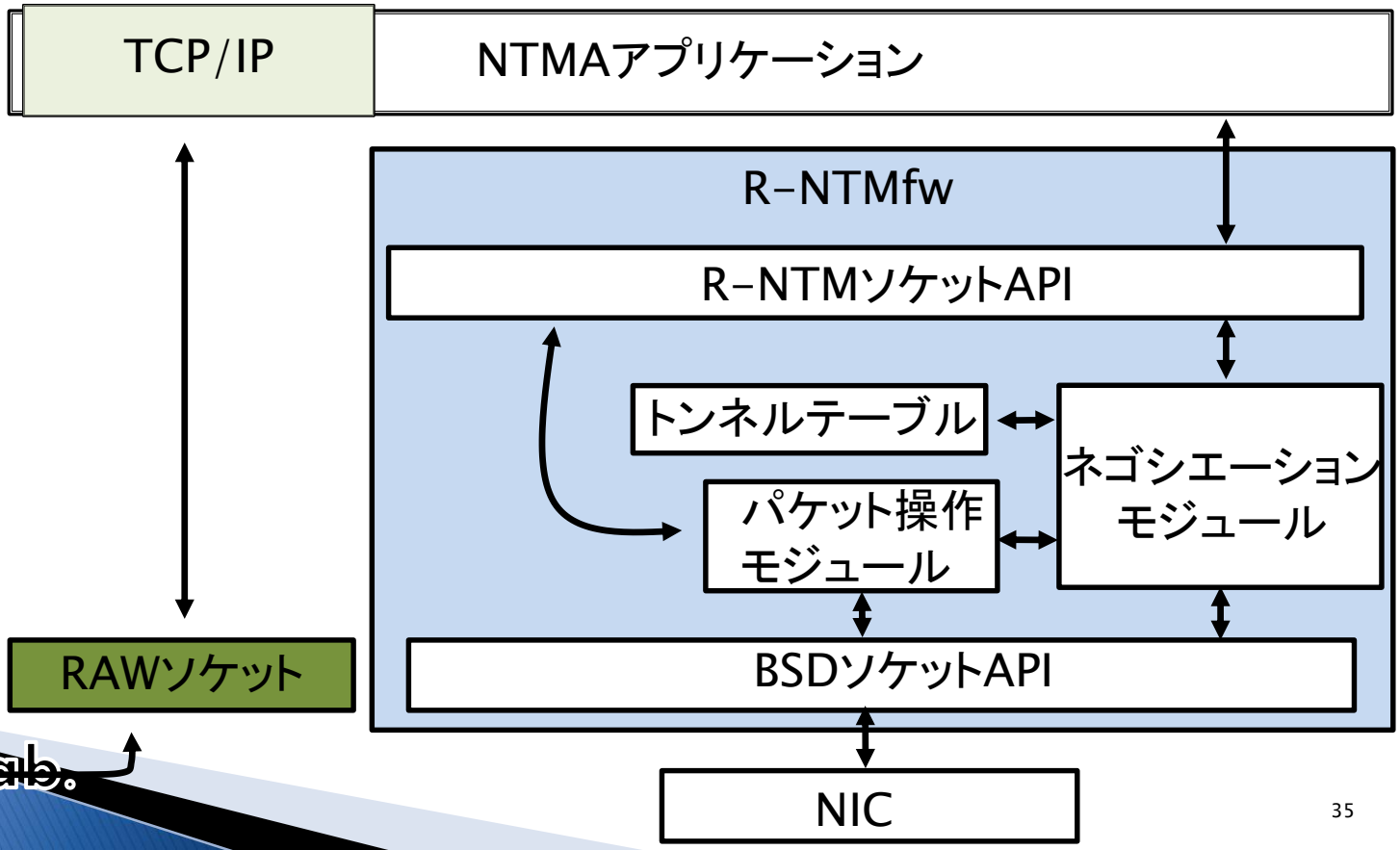
- NTMfwから仮想IPスタックの処理をスキップ



NTMAのモジュール構成

RAWソケット

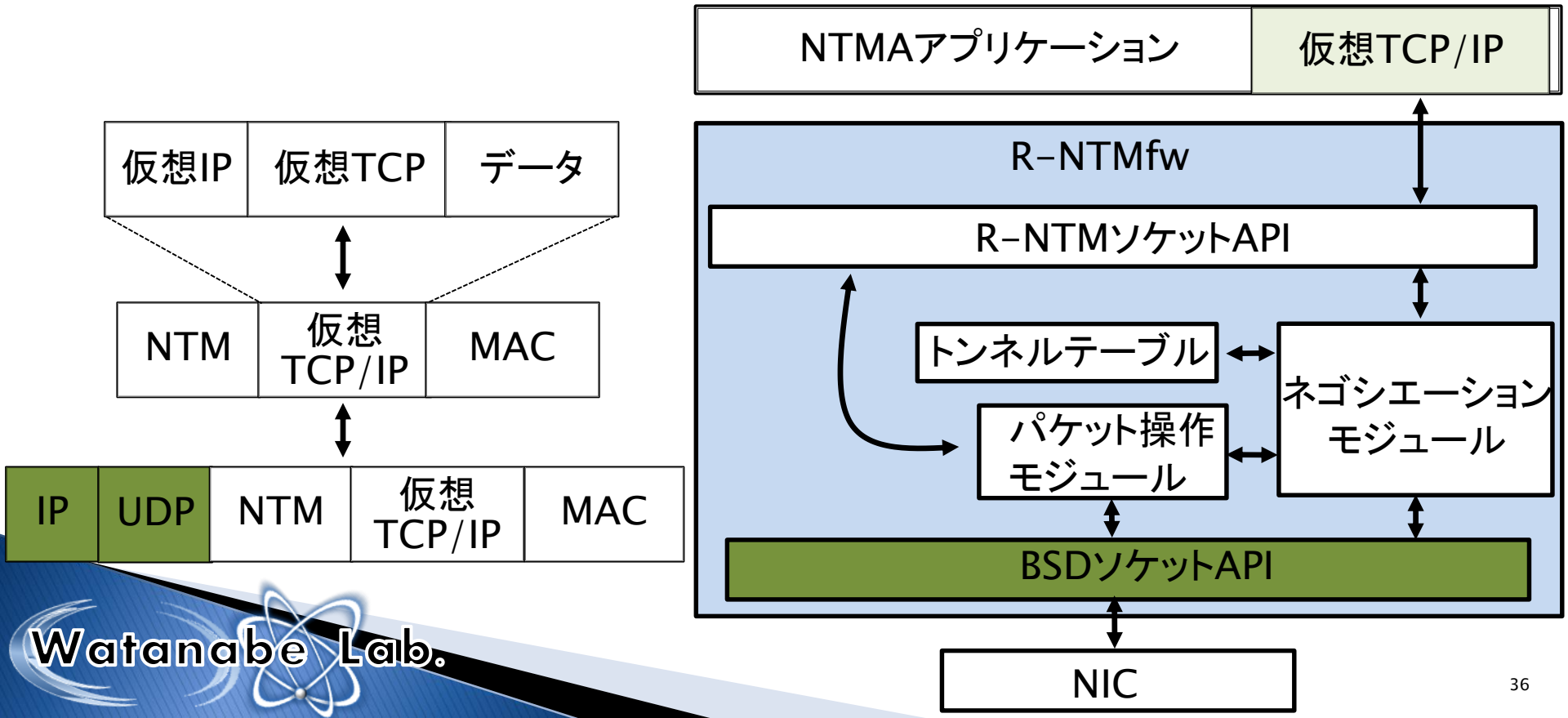
- 生のパケットをダイレクトに送受信
- 自インターフェース宛/以外のパケットも送受信できるように設定



NTMAのパケット遷移

BSDソケットAPI

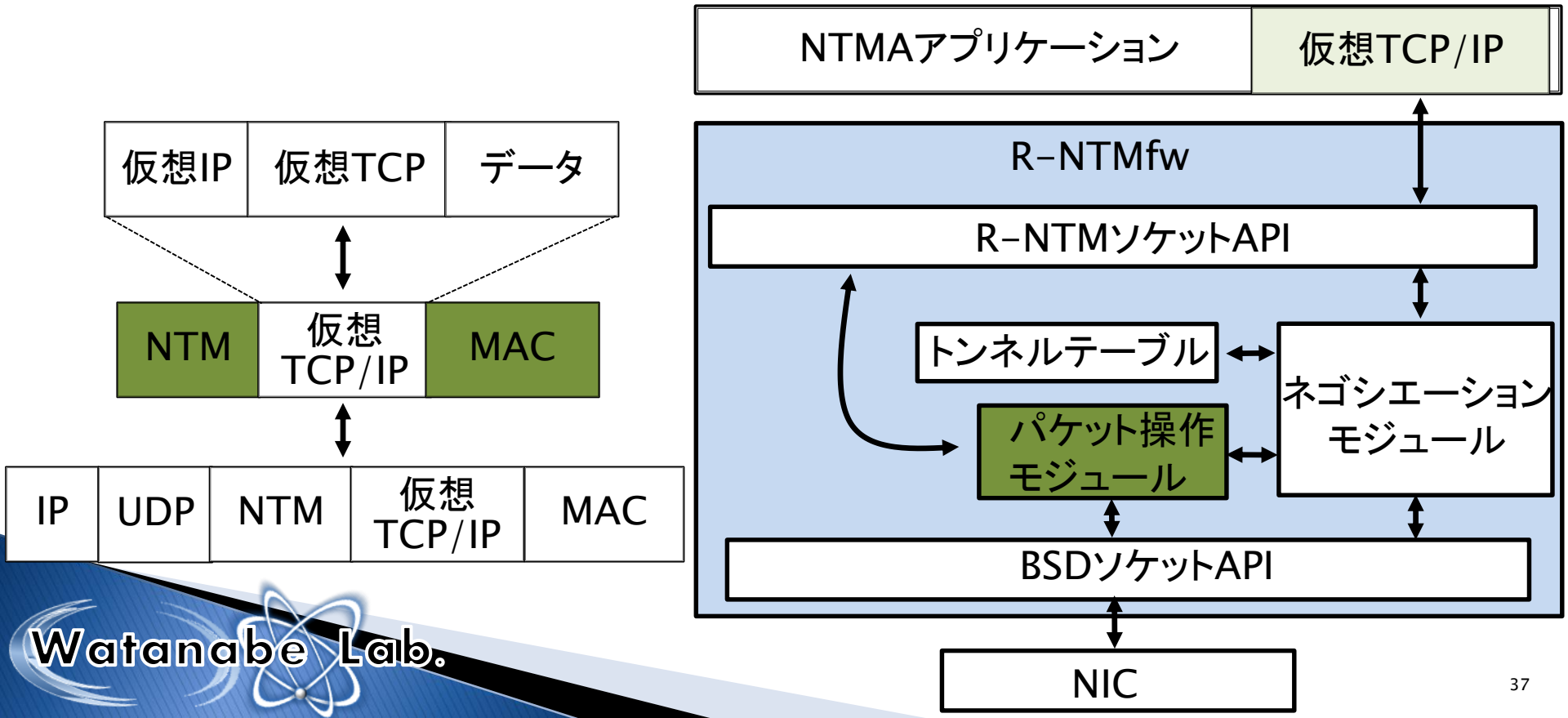
- NTMfwと同様の処理



NTMAのパケット遷移

パケット操作モジュール

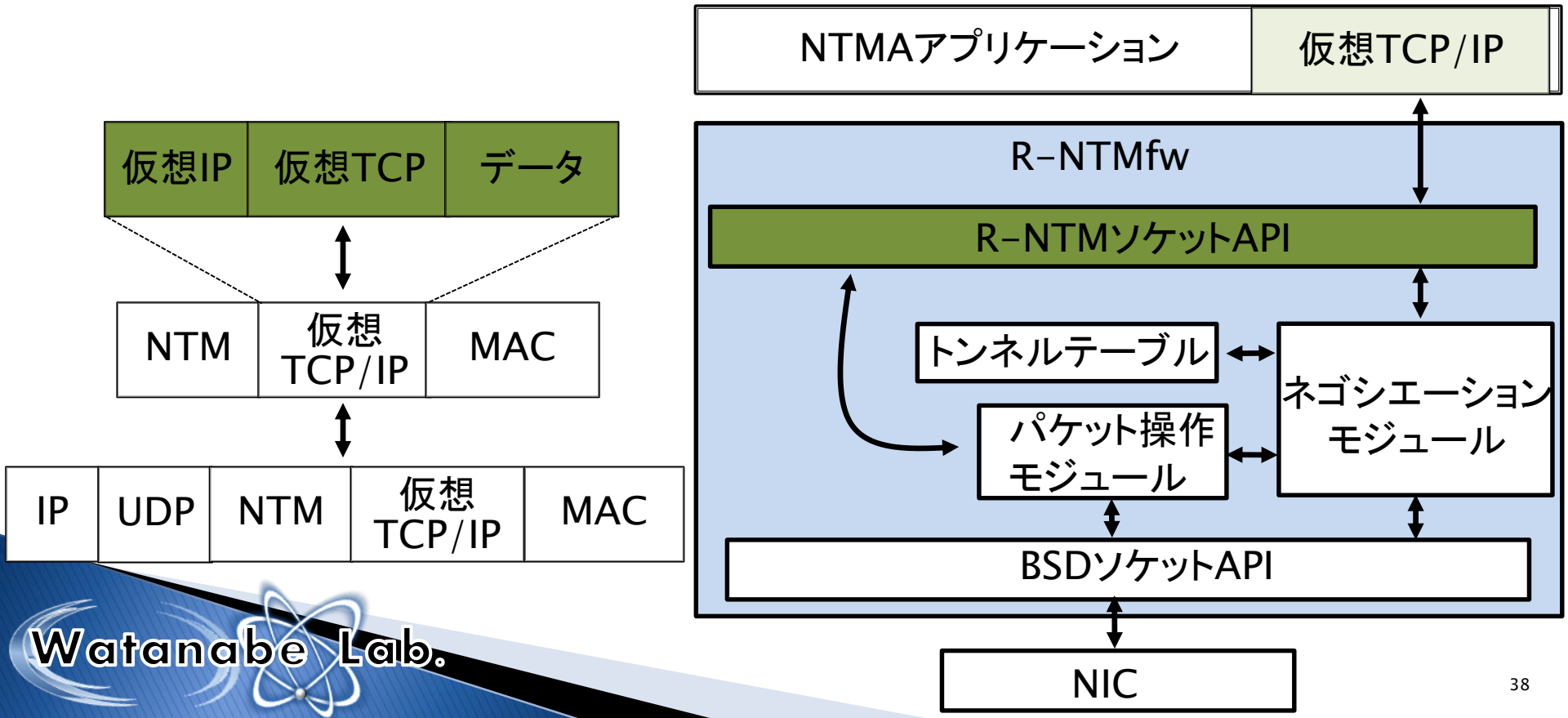
- NTMヘッダの処理，MAC検証
- 仮想IPスタックの処理を飛ばしてR-NTMfwへ処理を渡す



NTMAのパケット遷移

R-NTMfwソケットAPI

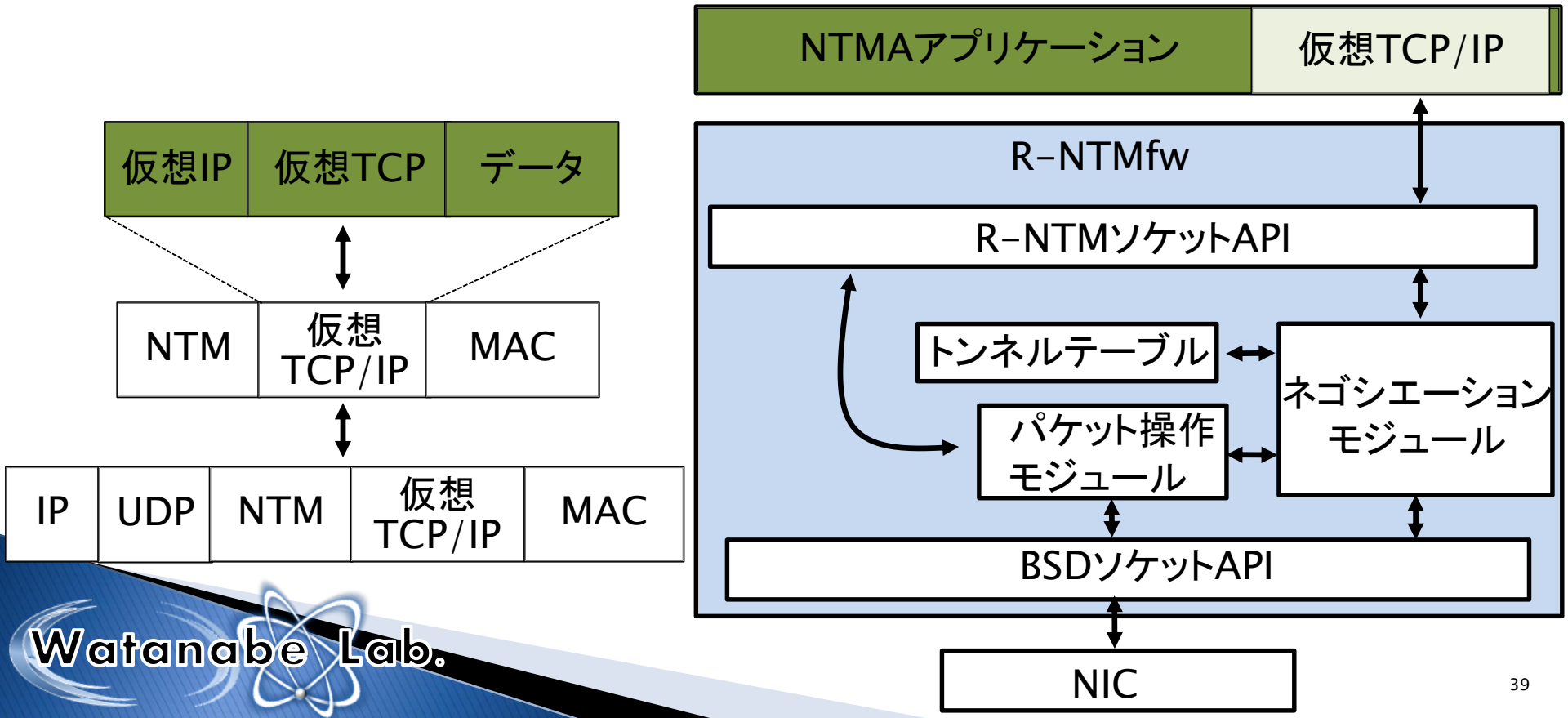
- TCP/IPが付与されたままのパケットをNTMAアプリケーションへ流す



NTMAのパケット遷移

NTMAアプリケーション

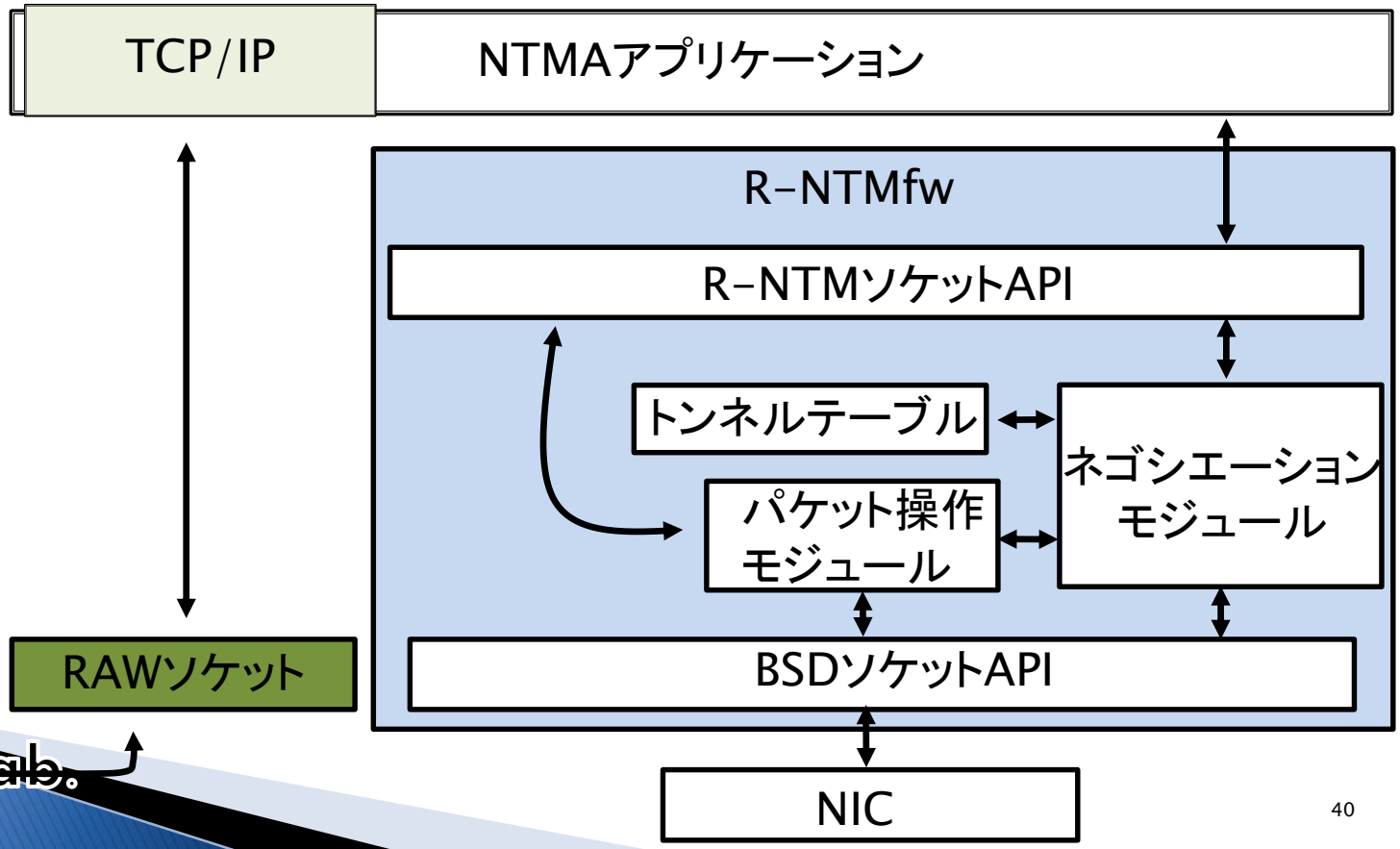
- GNとの通信は，RAWソケットで送受信



NTMAのモジュール構成

RAWソケット

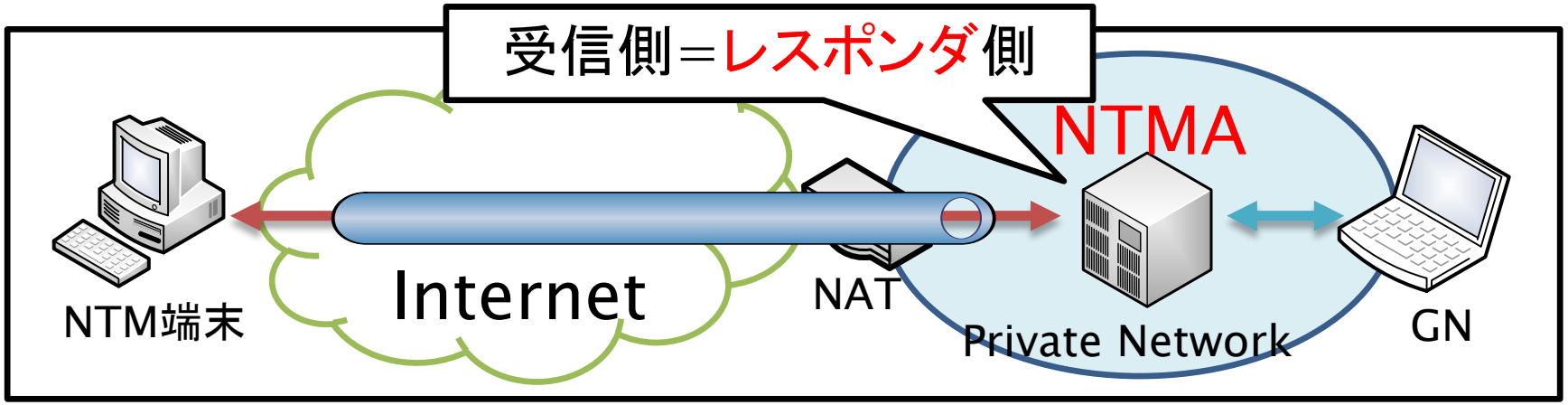
- 生のパケットをダイレクトに送受信
- 自インターフェース宛/以外のパケットも送受信できるように設定



動作検証

- 提案方式をLinux上に実装
  - レスポンダ側にNTMAを配置した通信を確認した

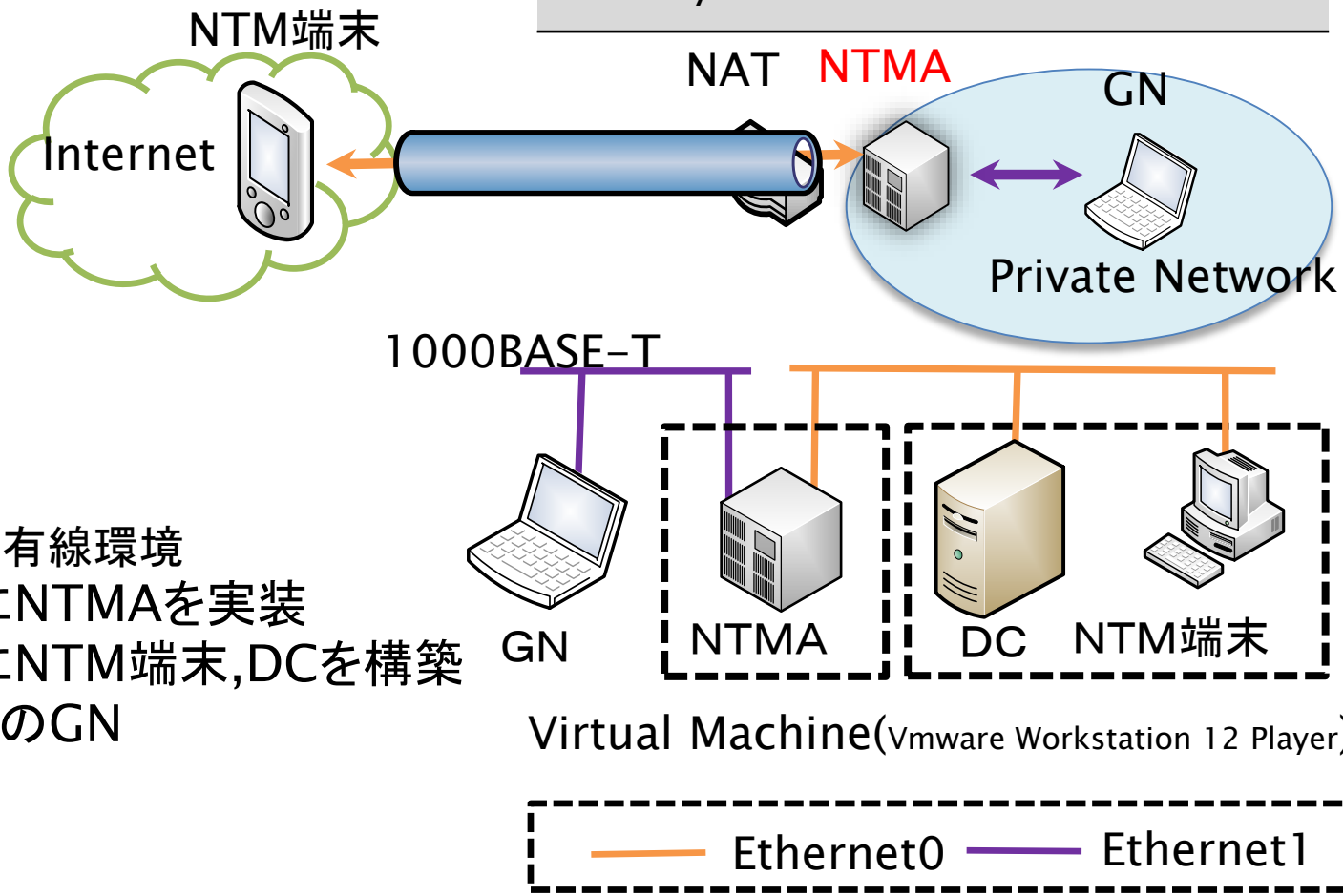
- 環境
  - 仮想マシン上にNTMAを実装
  - 仮想マシン上にNTM端末，DCを構築
  - Windows実機のGN



評価構成

- NTMAを評価

	NTMA(Virtual Machine)
OS	Ubuntu 14.04
CPU	Intel Corei7-930(2.80GHz)
Core	1コア
Memory	2GB



- 測定環境
  - 1000BASE-Tの有線環境
  - 仮想マシン上にNTMAを実装
  - 仮想マシン上にNTM端末,DCを構築
  - Windows実機のGN

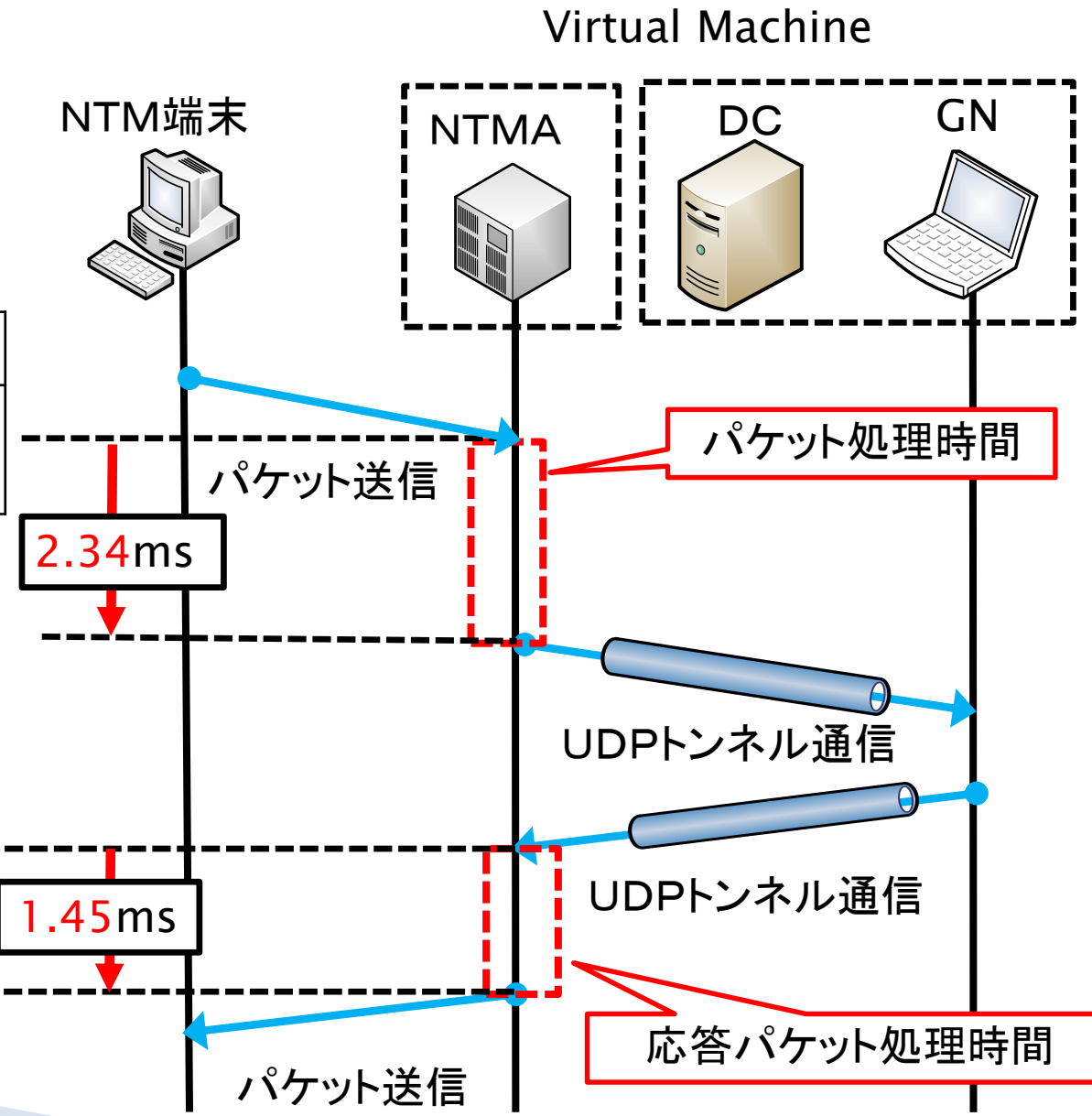
性能評価

10回の平均値

パケット処理時間	2.34ms
応答パケット処理時間	1.45ms

※Wiresharkにより測定

NTMAがパケット処理に要する時間は**僅か**



まとめ

- NTMAの実現方式の検討
  - 一般通信をNTMobile通信に変換
  - 一般端末のプログラムに手を加えない
- 今後の方針
  - GNに割り当てるIPアドレスの検討