

# NTMobileにおける 仮想IPv4アドレス運用手法の提案と実装

加古将規<sup>†</sup> 上醉尾一真<sup>‡</sup> 鈴木秀和<sup>†</sup> 内藤克浩<sup>††</sup> 渡邊晃<sup>†</sup>

<sup>†</sup> 名城大学理工学部 <sup>‡</sup> 名城大学大学院理工学研究科 <sup>††</sup> 三重大学大学院工学研究科

## 1 はじめに

スマートフォンなどの移動通信端末の普及により、移動しながら通信できる技術（移動透過性技術）が必要となっている。また、現在のインターネットでは、インターネット側の端末から NAT 配下の端末に対して通信を開始できない通信接続性の課題が存在しており、これを解決する技術が求められている。

我々は、移動透過性と通信接続性を同時に実現する技術として、NTMobile (Network Traversal with Mobility) を提案している。NTMobile では、NTMobile の機能を実装した端末 (NTM 端末) に対して一意な仮想 IPv4 アドレスを割り当てるが、仮想 IPv4 アドレスとして利用できる範囲がせまいという課題があった。本稿では、NTM 端末内部で仮想 IPv4 アドレスを自律的に生成し、上記課題を解決する手法を提案する。また Linux 上で提案方式の実装を行ったので報告する。

## 2 NTMobile の概要

図 1 に NTMobile の概要を示す。NTMobile は、NTMobile を実装した NTM 端末、通信経路を指示する DC (Direction Coordinator), NTM 端末と一般端末 GN 間の通信などでエンドエンドの通信が行えない場合にパケットを中継する RS (Relay Server) によって構成される。NTM 端末は起動時に端末情報を DC に登録する。その後、DC から仮想 IP アドレスを取得する。また、通信開始時に DC から通信相手の仮想 IP アドレスを取得し、アプリケーションに対して通信相手の IP アドレスとして認識させる。仮想アドレスに基づくアプリケーションパケットは、NTMobile の機能により実 IP アドレスでカプセル化され、通信相手へ送信される。NTM 端末の移動によって実 IP アドレスが変

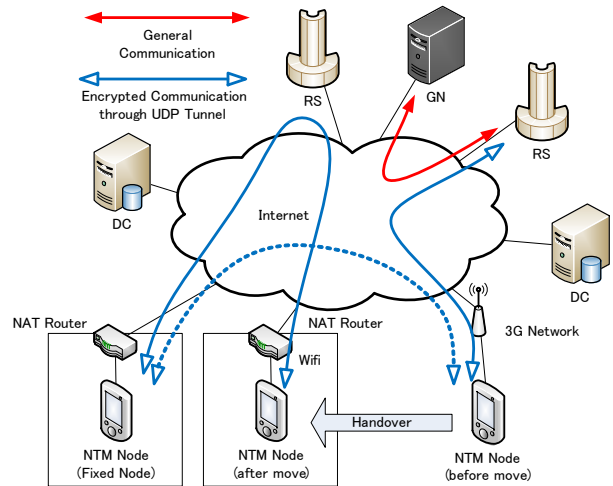


図 1: NTMobile の概要

化しても、仮想 IP アドレスは変化しないため、アプリケーション間の通信を継続することができる。

NTMobile では、仮想 IPv4 アドレスと実 IPv4 アドレスの重複を防ぐため、仮想 IPv4 アドレスを実ネットワークで利用されないアドレス領域から割り当てている。しかし、利用可能なアドレス領域が小さいため、大規模システムに適用できず、NTMobile の拡張性を損なうという課題があった。

## 3 提案方式

NTM 端末が仮想 IPv4 アドレスを自律的に生成し、通信する端末間の仮想 IPv4 アドレスを端末内部で管理する手法を提案する。この手法により、NTMobile 全体で仮想 IPv4 アドレス領域を共有する必要がなくなり、限られた仮想アドレス領域を用いて大規模に NTMobile を運用することが可能となる。

### 3.1 端末登録時の処理

NTM 端末は起動時の DC への端末登録時に、静的な仮想 IPv4 アドレスを自端末の IP アドレスとしてアプリケーションに認識させる。

**Proposal of Management Method of Virtual IPv4 Addresses in NTMobile and its implementation**

Masanori Kako<sup>†</sup>, Kazuma Kamienoo<sup>‡</sup>, Hidekazu Suzuki<sup>†</sup>, Katsuhiko Naito<sup>††</sup> and Akira Watanabe<sup>†</sup>

<sup>†</sup> Faculty of Science and Technology, Meijo University

<sup>‡</sup> Graduate School of Science and Technology, Meijo University

<sup>††</sup> Graduate School of Engineering, Mie University

### 3.2 通信開始時の処理

NTM 端末は通信開始時に DC に名前解決を依頼し、DC から通信経路の指示を受ける。このときに、NTM 端末は端末内部で仮想 IPv4 アドレスを生成し通信相手の IP アドレスとしてアプリケーションに認識させる。通信相手の仮想 IPv4 アドレスは、経路情報 Path ID に関連付けて NTM 端末のトンネルテーブルに登録する。Path ID は DC から指定される情報で、MN と CN が通信識別子として利用する。

### 3.3 トンネル通信時の処理

図 2 に、NTM 端末間において提案方式によるトンネル通信を行った場合のシーケンスを示す。MN のアプリケーションは、自身の仮想 IPv4 アドレスを  $VIP_A$ 、CN の仮想 IPv4 アドレスを  $VIP_B$  として認識している。また、CN のアプリケーションは、自身の仮想 IPv4 アドレスを  $VIP_Y$ 、MN の仮想 IPv4 アドレスを  $VIP_X$  として認識している。

MN のアプリケーションが CN へパケットを送信する際、送信元アドレスに  $VIP_A$ 、宛先アドレスに  $VIP_B$  が記載された仮想 IP パケットが生成される。仮想 IP パケットは実 IP アドレスでカプセル化された後、CN へ送信される。このとき、カプセル化されたパケットには Path ID が付加される。CN はカプセル化パケットを受信すると、パケットのデカプセル化を行い仮想 IP パケットを抽出する。その後、CN はパケット内の Path ID を元に自身のトンネルテーブルを検索し、MN の仮想 IPv4 アドレス  $VIP_X$  を取得する。CN はパケット内の送信元アドレスを  $VIP_A$  から  $VIP_X$  へ、宛先アドレスを  $VIP_B$  から  $VIP_Y$  へ変換し、CN のアプリケーションへ渡す。

以上により、NTM 端末内部で仮想 IP アドレスを管理することにより、限られた仮想アドレス領域を用いて大規模に NTMobile を運用することが可能となる。

## 4 実装

NTMobile の基本動作は Linux において既に動作が検証されている。NTM 端末はユーザ空間の NTMobile デーモンと、カーネル空間の NTMobile カーネルモジュールにより動作する。NTMobile デーモンは DC への NTM 端末情報の登録と仮想 IP アドレスの取得、および DC の指示に従ったトンネル構築を行う。カーネルモジュールはパケットのカプセル化/デカプセル化および暗号化処理を行う。各モジュールに以下のような改造を行った。

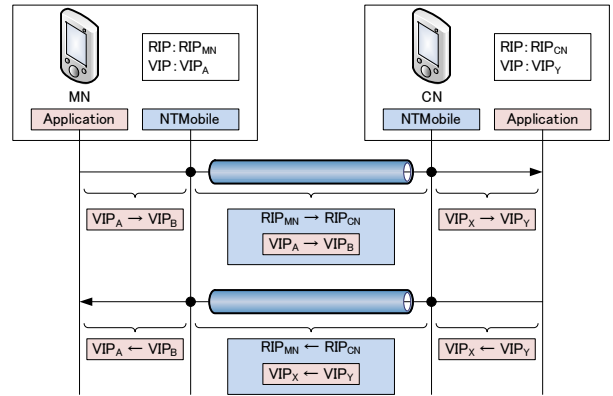


図 2: トンネル通信時のアドレス遷移

### 4.1 NTMobile デーモン

NTM 端末の端末登録時に自端末の仮想インタフェースに静的な仮想 IPv4 アドレスを設定する。また、通信開始時に通信相手の仮想 IPv4 アドレスを端末内部に設定し、トンネルテーブルに登録する。

### 4.2 NTMobile カーネルモジュール

NTMobile カーネルモジュールが受信パケットをフックし、デカプセル化を行ったパケットから Path ID を取得する。Path ID をキーとして、トンネルテーブルから通信相手の仮想 IPv4 アドレスとして設定した IP アドレスを検索する。その後、パケット内の仮想 IPv4 アドレスの送信元および宛先を端末内部で管理する仮想 IPv4 アドレスに変換する。

## 5 まとめ

本稿では、NTM 端末内部で仮想 IPv4 アドレスを自律的に生成し、通信する端末間の仮想 IPv4 アドレスを端末内部で管理する手法を提案した。この手法により、NTMobile 全体で仮想 IPv4 アドレス領域を共有する必要がなくなるため、限られた仮想アドレス領域で大規模に NTMobile を運用することが可能となる。また Linux 上で提案方式の実装を行い動作を検証した。

## 6 謝辞

本研究は SCOPE/PREDICT の委託研究に基づく結果である。

## 参考文献

- [1] 内藤克浩. 他: NTMobile における移動透過性の実現と実装, 情報処理学会論文誌, Vol.54, No.1, pp.380-393 (2013) .

# NTMobileにおける 仮想IPv4アドレス運用手法の提案と実装

---

加古将規<sup>†</sup> 上酔尾一真<sup>††</sup> 鈴木秀和<sup>†</sup> 内藤克浩<sup>‡</sup> 渡邊晃<sup>†</sup>

<sup>†</sup>名城大学 理工学部

<sup>††</sup>名城大学大学院 理工学部研究科

<sup>‡</sup>三重大学大学院 工学研究科

## ■ インターネット通信の需要増加

- 公衆無線網やスマートフォンなどの携帯端末の普及

## ■ 移動透過性技術の必要性

- 現在のIPネットワークでは, IPアドレスによって通信を識別
- 移動時などのネットワーク切り替えによって, 端末のIPアドレスが変化

移動しながらの通信ができない

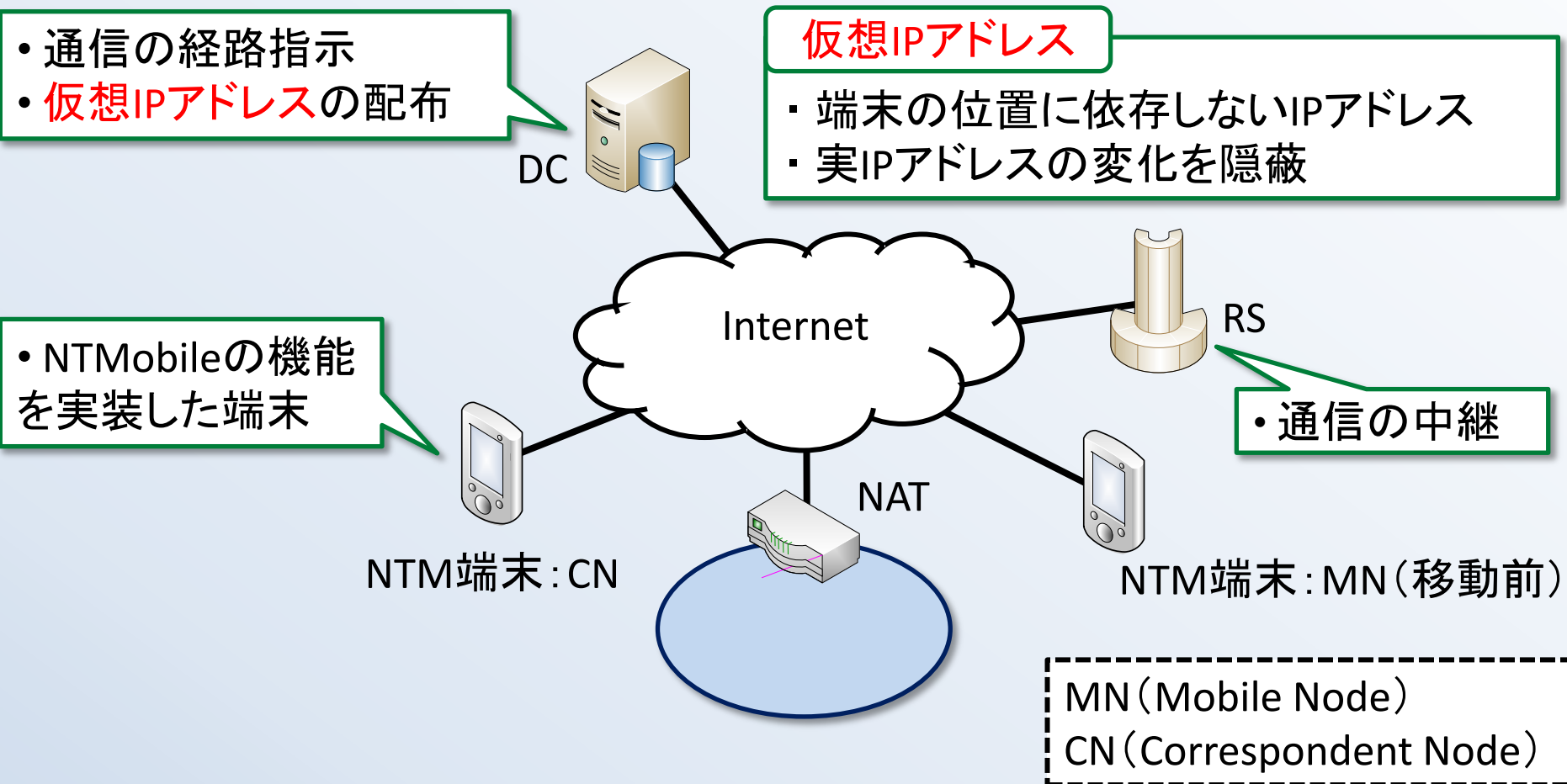
## ■ NAT越え技術の必要性

- IPv4アドレスの枯渇に伴い, NATを導入し, プライベートネットワークを構築することが一般的
- NATの外側から内側の端末にアクセスできない

端末同士の自由な通信ができない

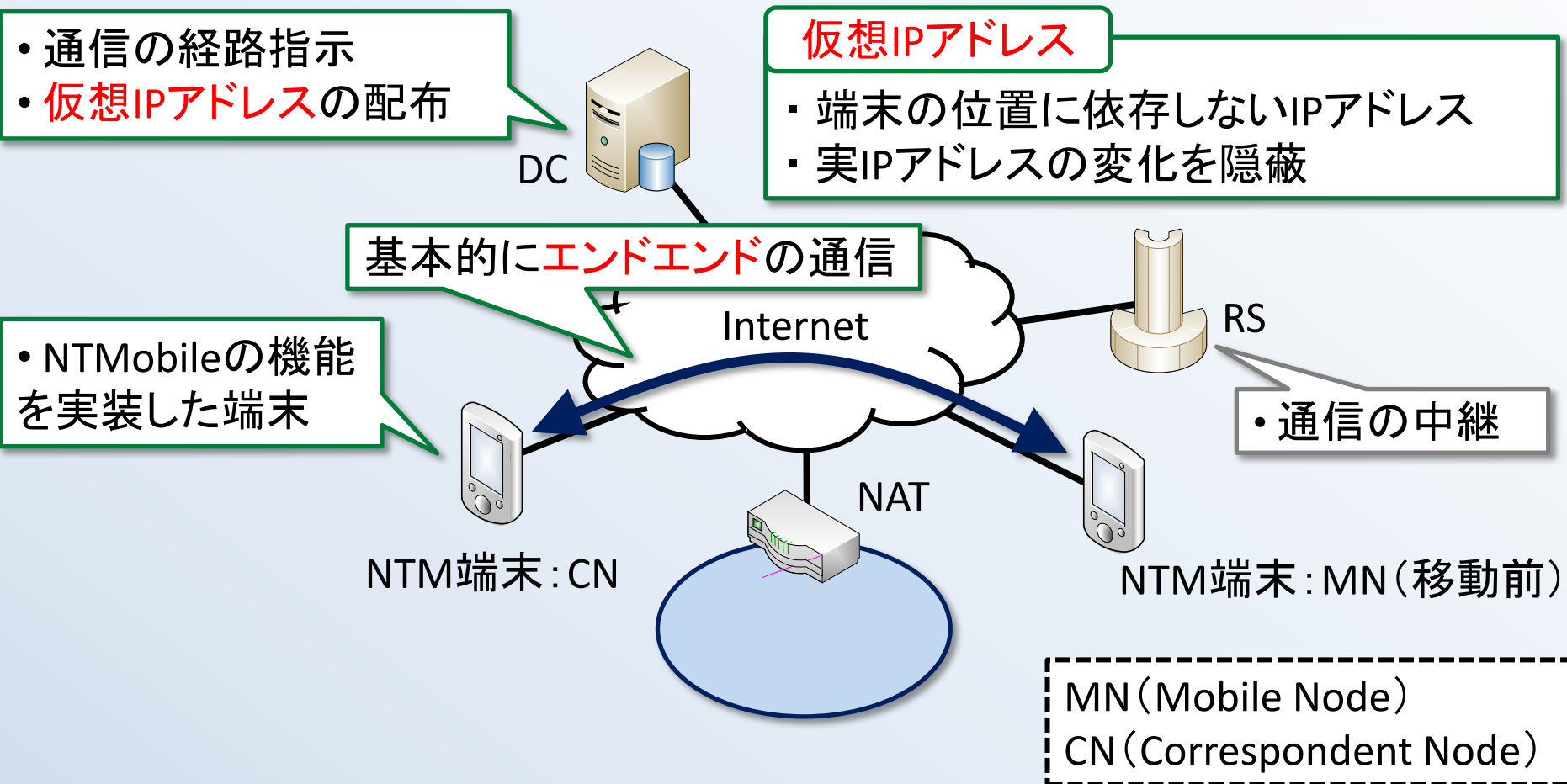
## ■ NTMobile(Network Traversal with Mobility)

- 移動透過性とNAT越えを同時に実現する技術
- NTM端末, DC(Direction Coordinator), RS(Relay Server)で構成



## ■ NTMobile(Network Traversal with Mobility)

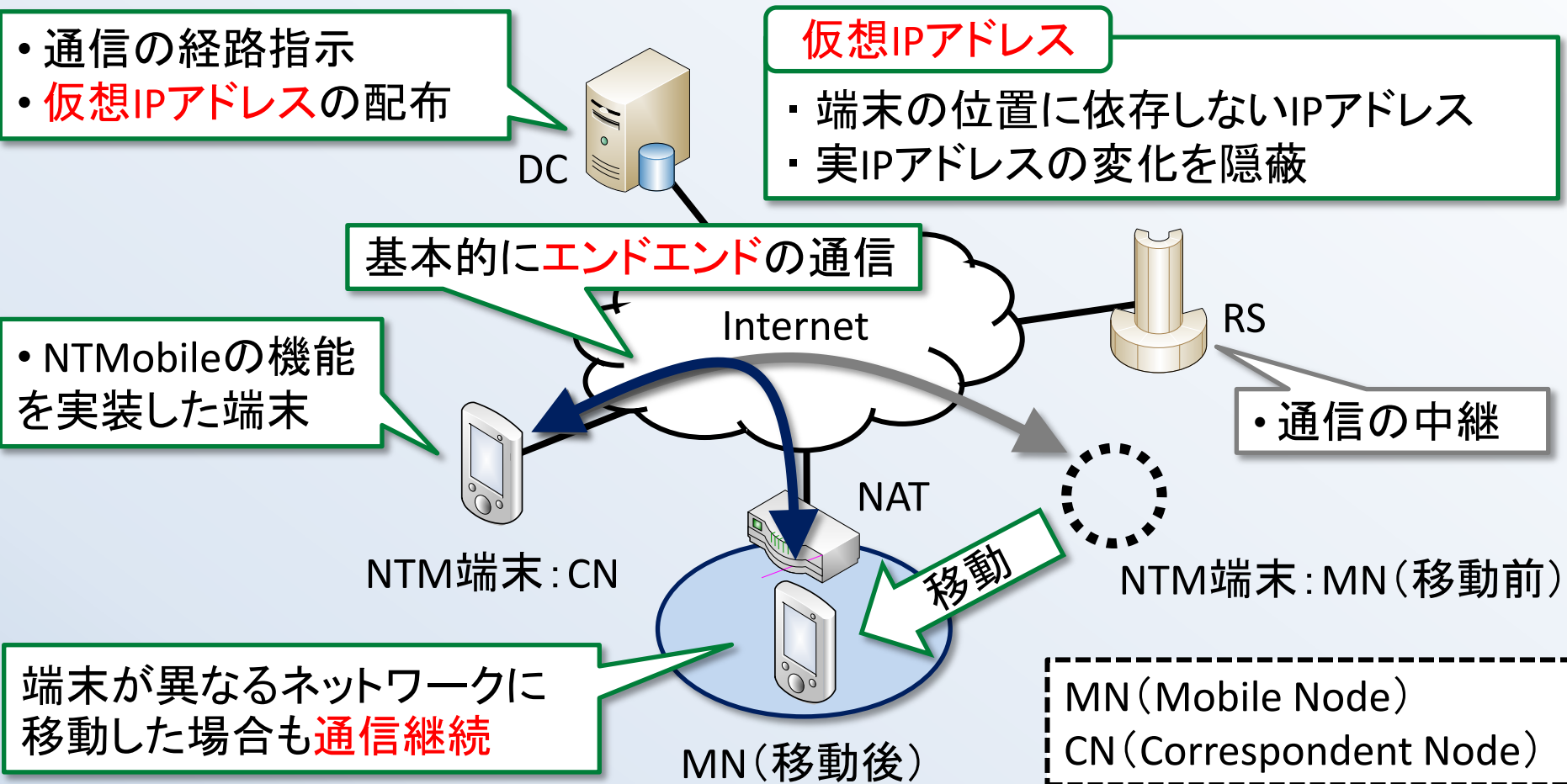
- 移動透過性とNAT越えを同時に実現する技術
- NTM端末, DC(Direction Coordinator), RS(Relay Server)で構成



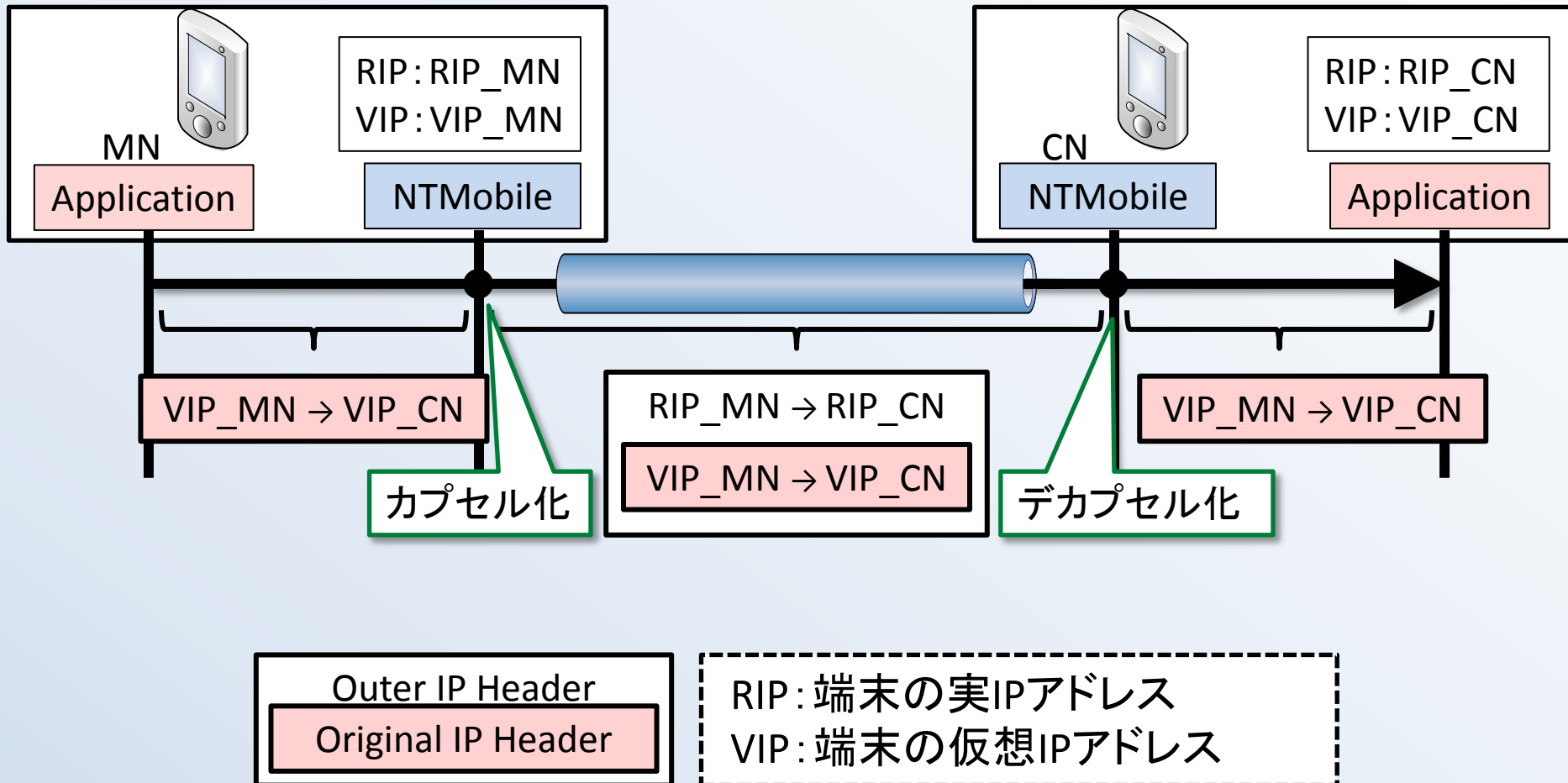


## ■ NTMobile(Network Traversal with Mobility)

- 移動透過性とNAT越えを同時に実現する技術
- NTM端末, DC(Direction Coordinator), RS(Relay Server)で構成

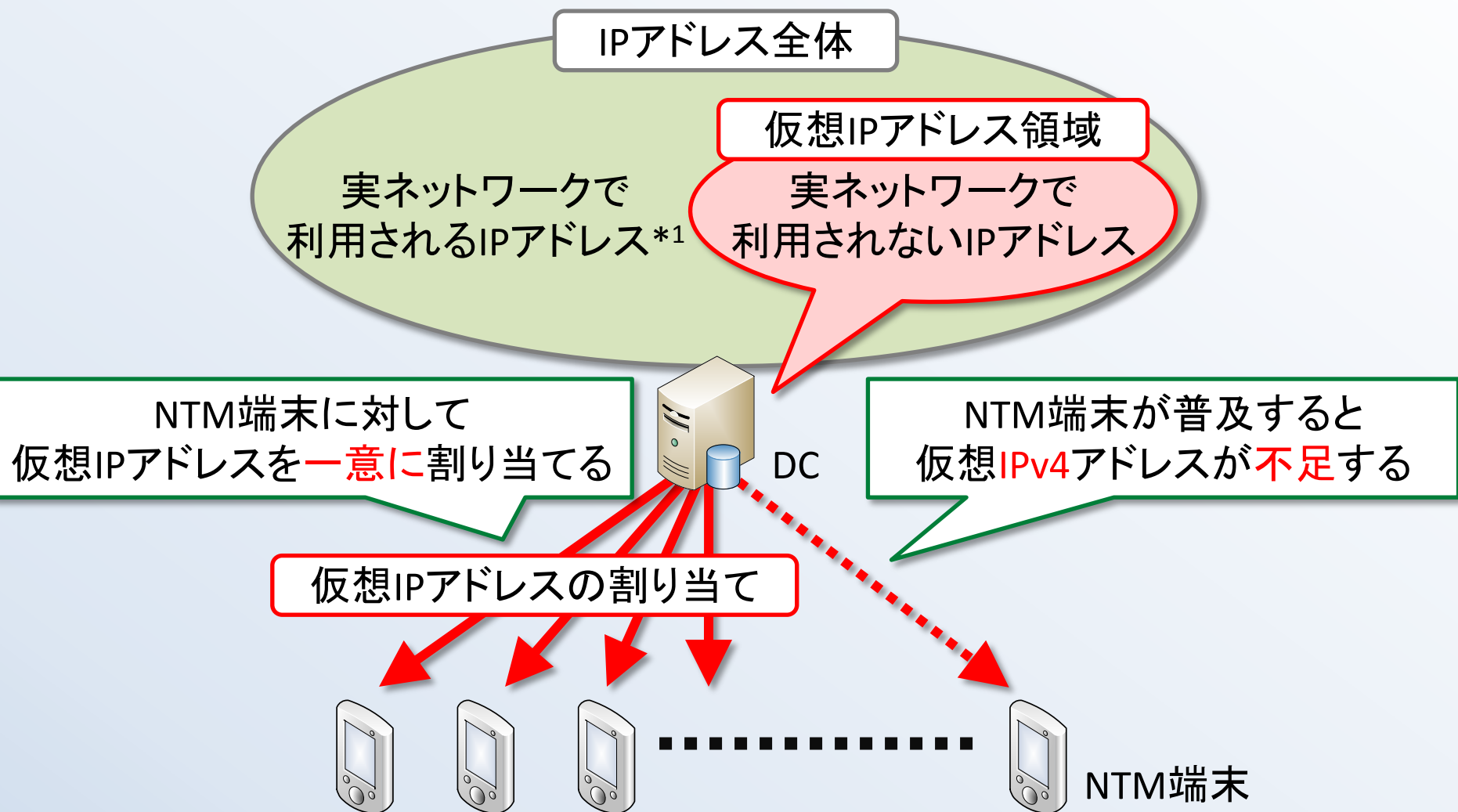


## ■ IPパケットを実IPアドレスでカプセル化した通信



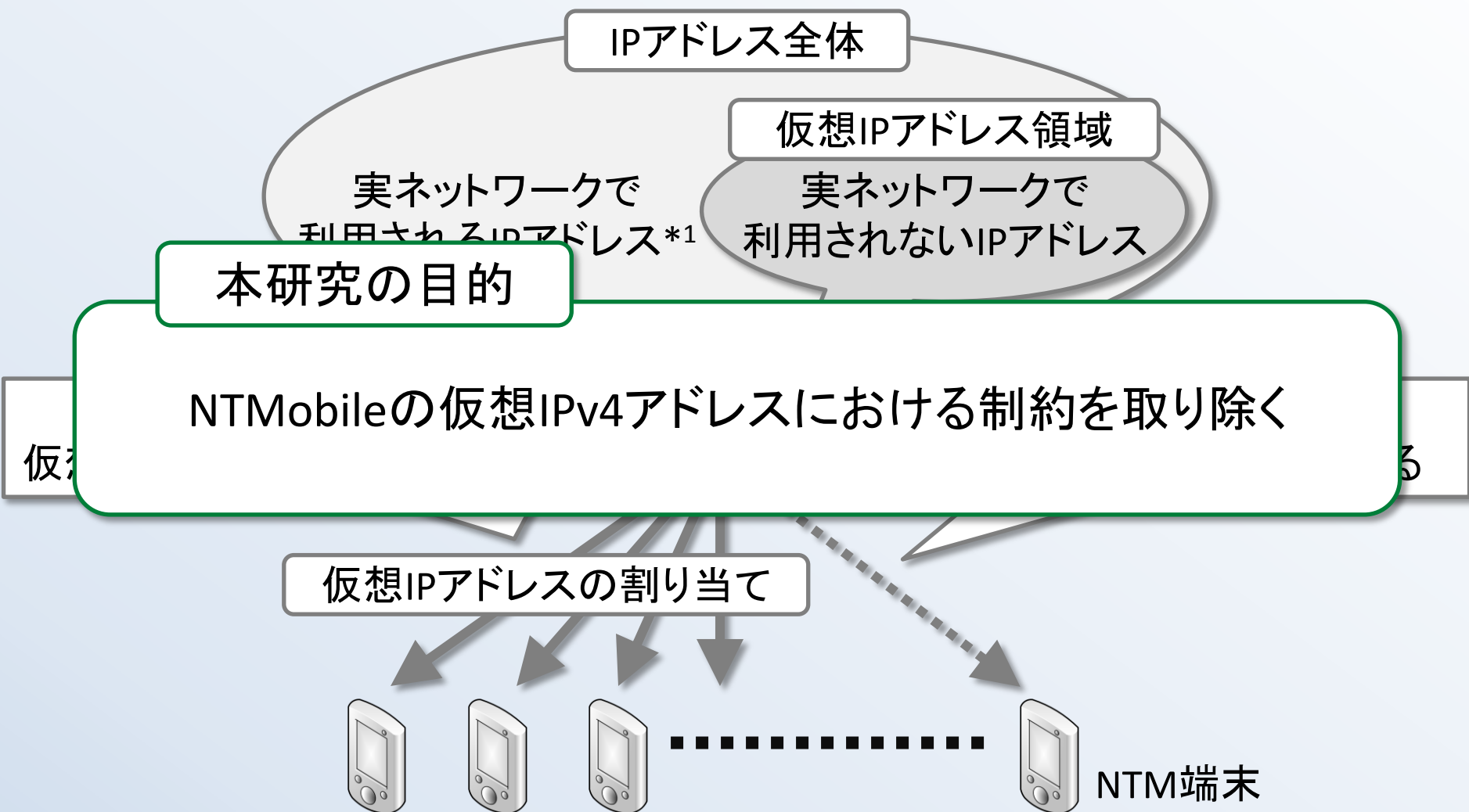


## ■ 大規模運用を考えた場合に、仮想IPv4アドレスが足りない



\*1: グローバルIPアドレス, プライベートIPアドレスなどのIPアドレス

- 大規模運用を考えた場合に、仮想IPv4アドレスが足りない



\*1: グローバルIPアドレス, プライベートIPアドレスなどのIPアドレス

- 端末内部で一意的な仮想IPv4アドレスを**自律的**に生成・管理
- パケット受信時, 端末が管理する仮想IPv4アドレスに変換
  - 通信は**Path ID**\*<sup>2</sup>を用いて識別

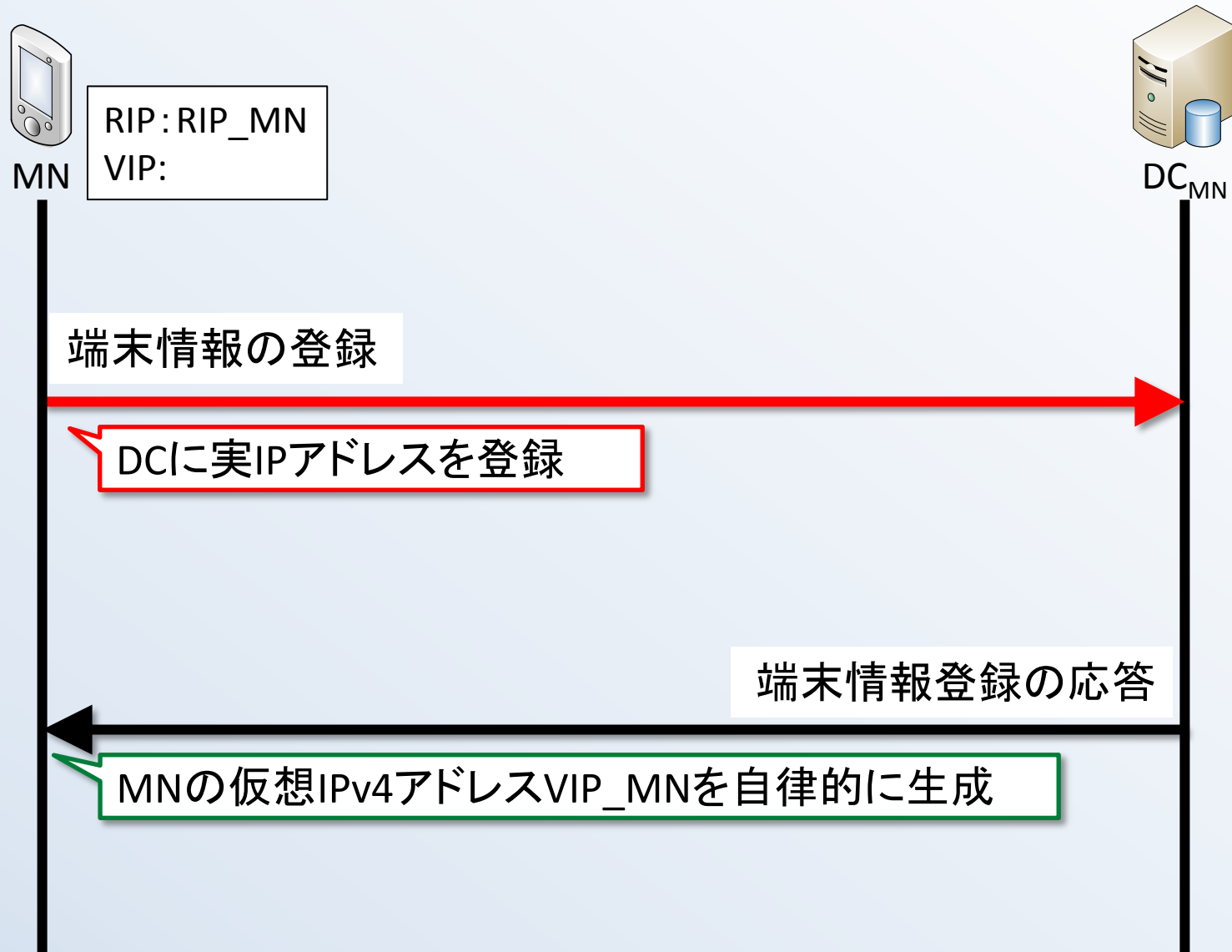
一意に通信を識別するPath IDを用いて  
端末が仮想IPv4アドレスの管理する

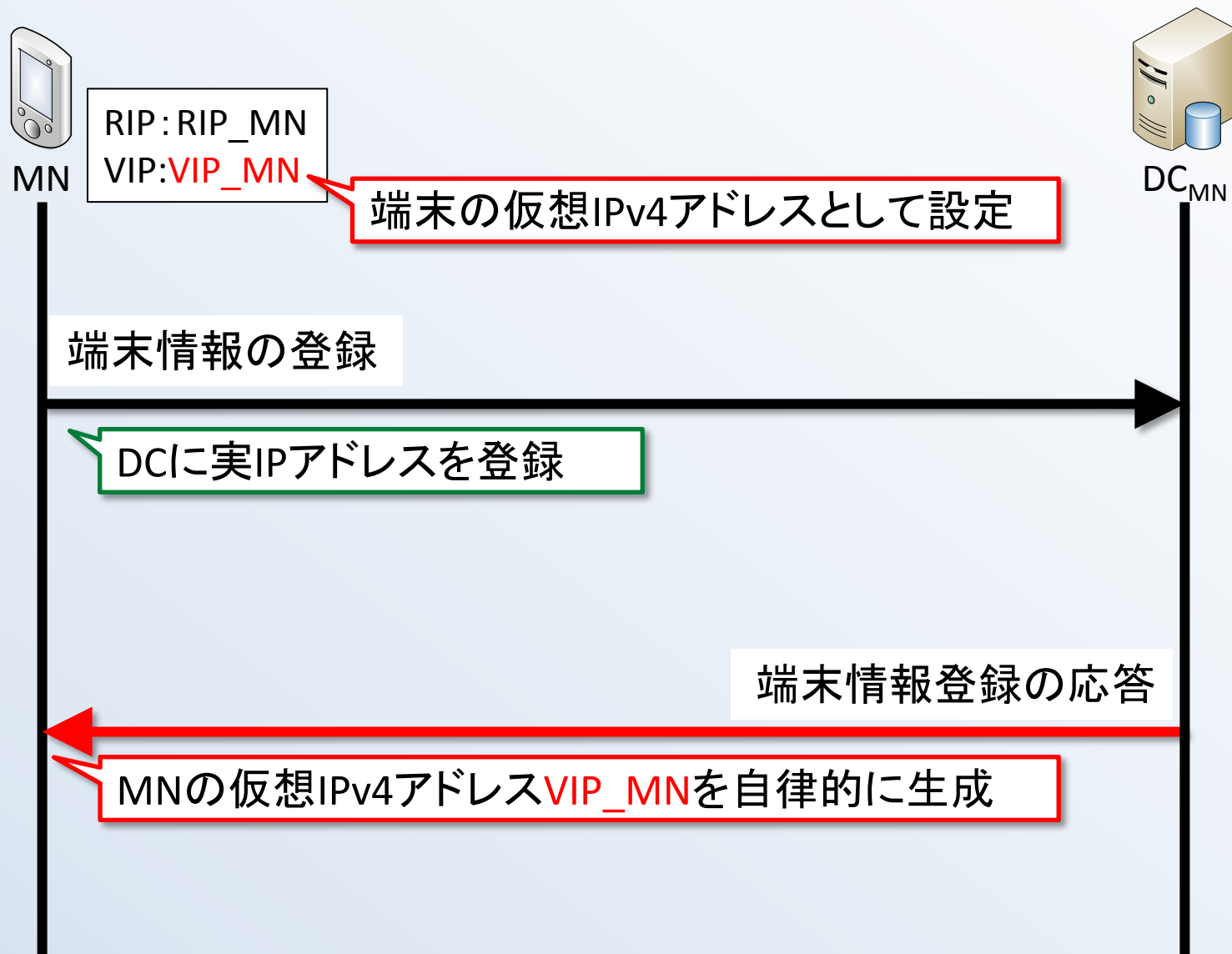


NTMobileのシステム全体で仮想IPv4アドレスの節約が可能

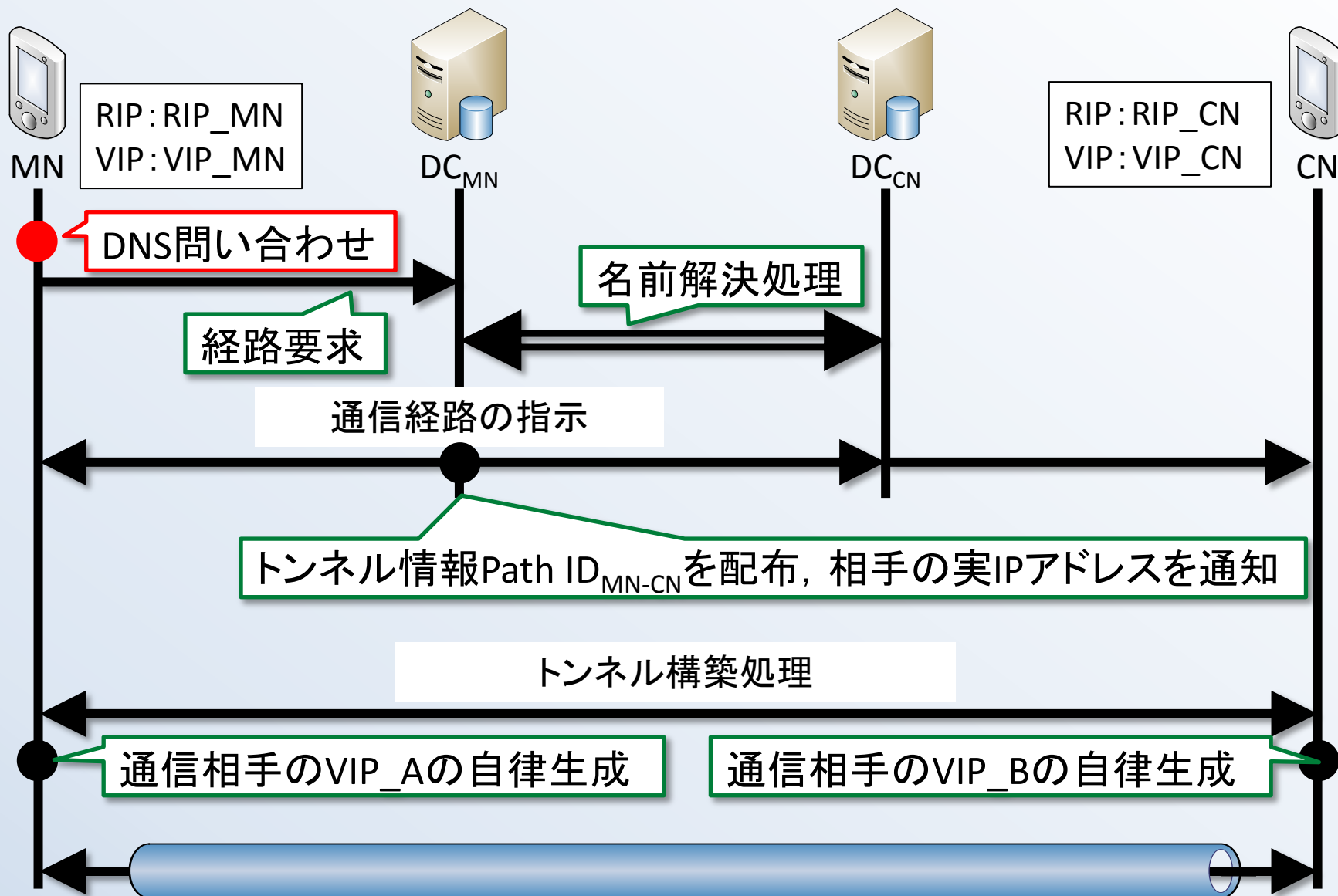
\*<sup>2</sup>: DCが通知するNTMobileの通信を一意的に識別する通信識別子

# 通信シーケンス(端末起動時)

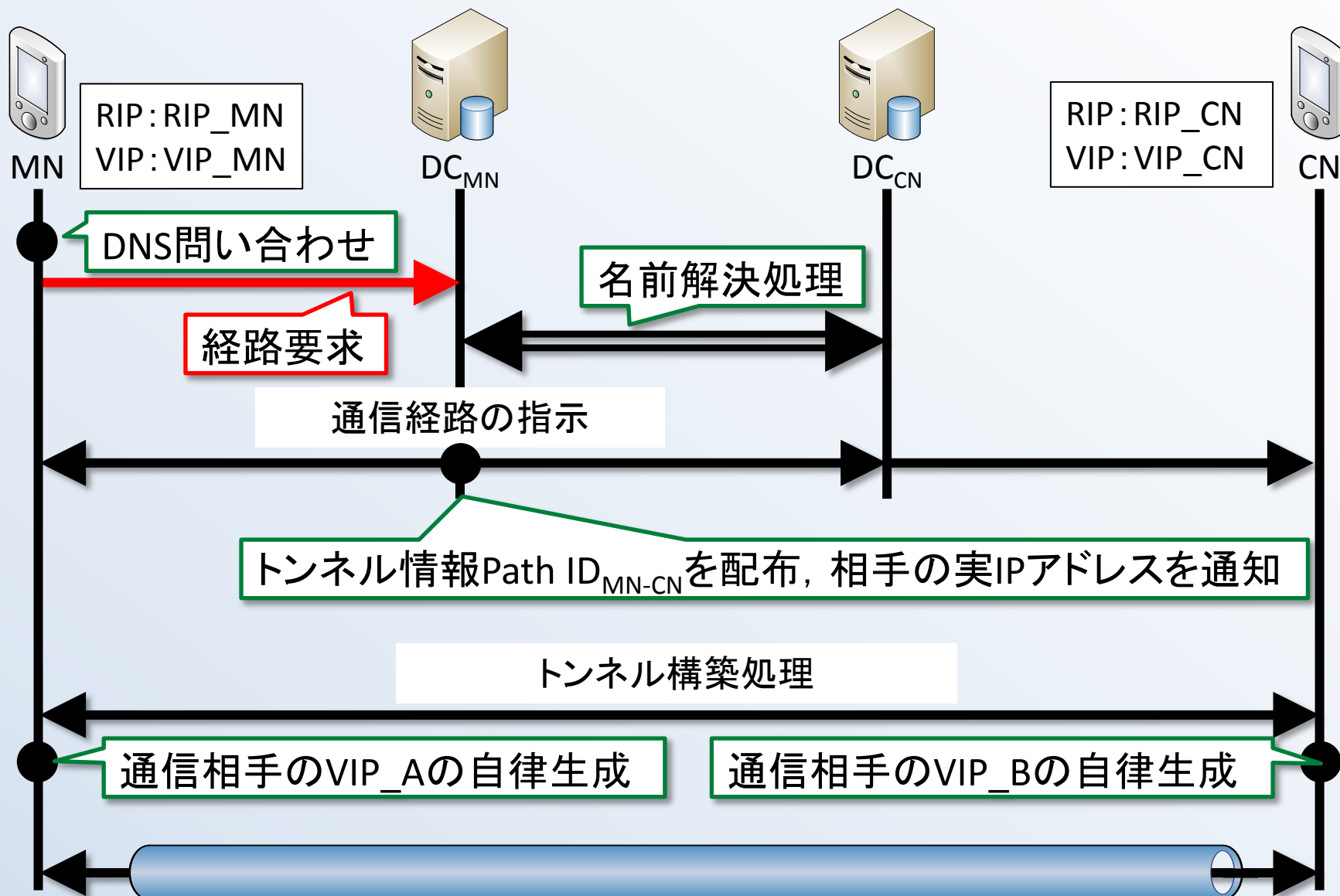




# 通信シーケンス(通信開始時)

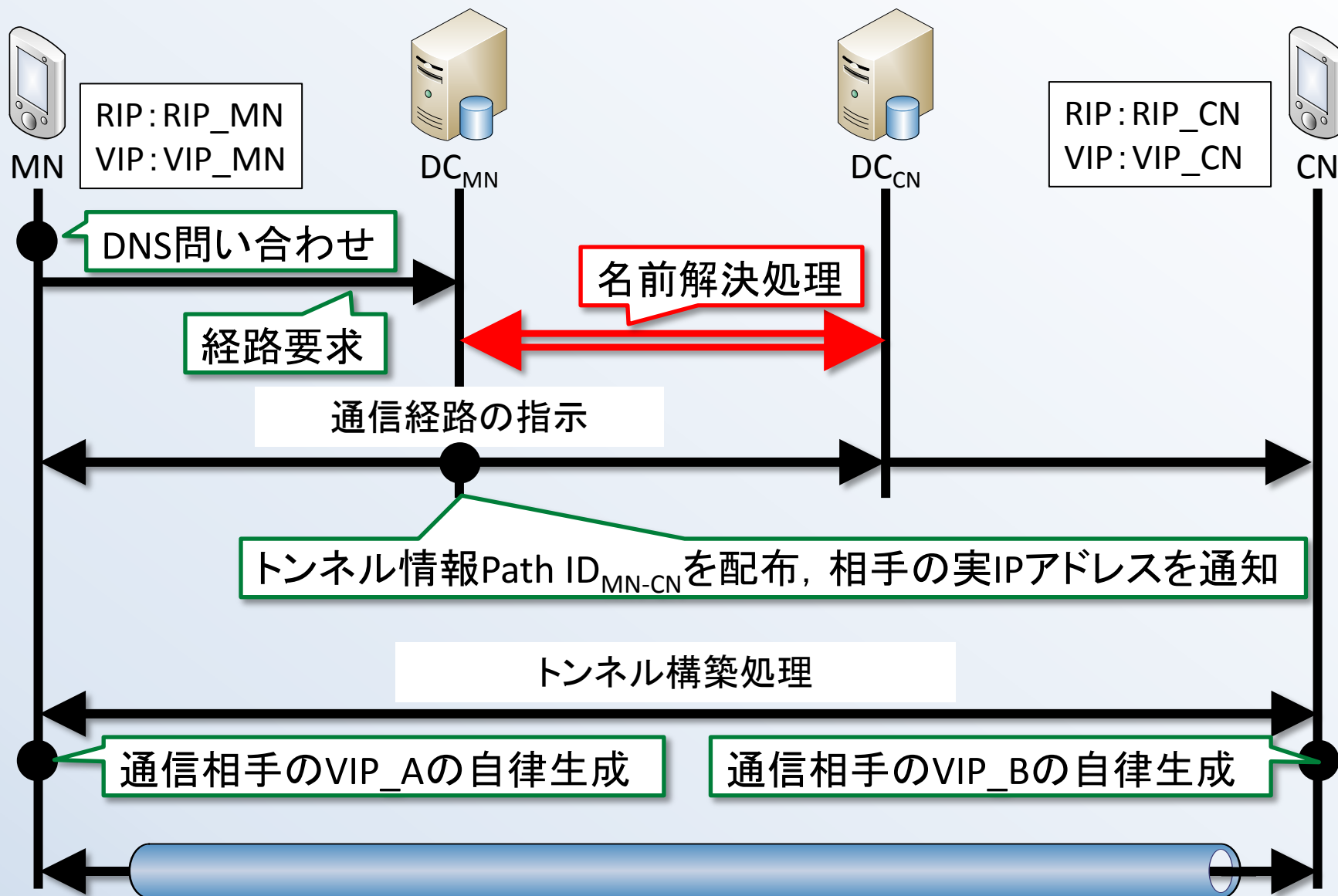


# 通信シーケンス(通信開始時)

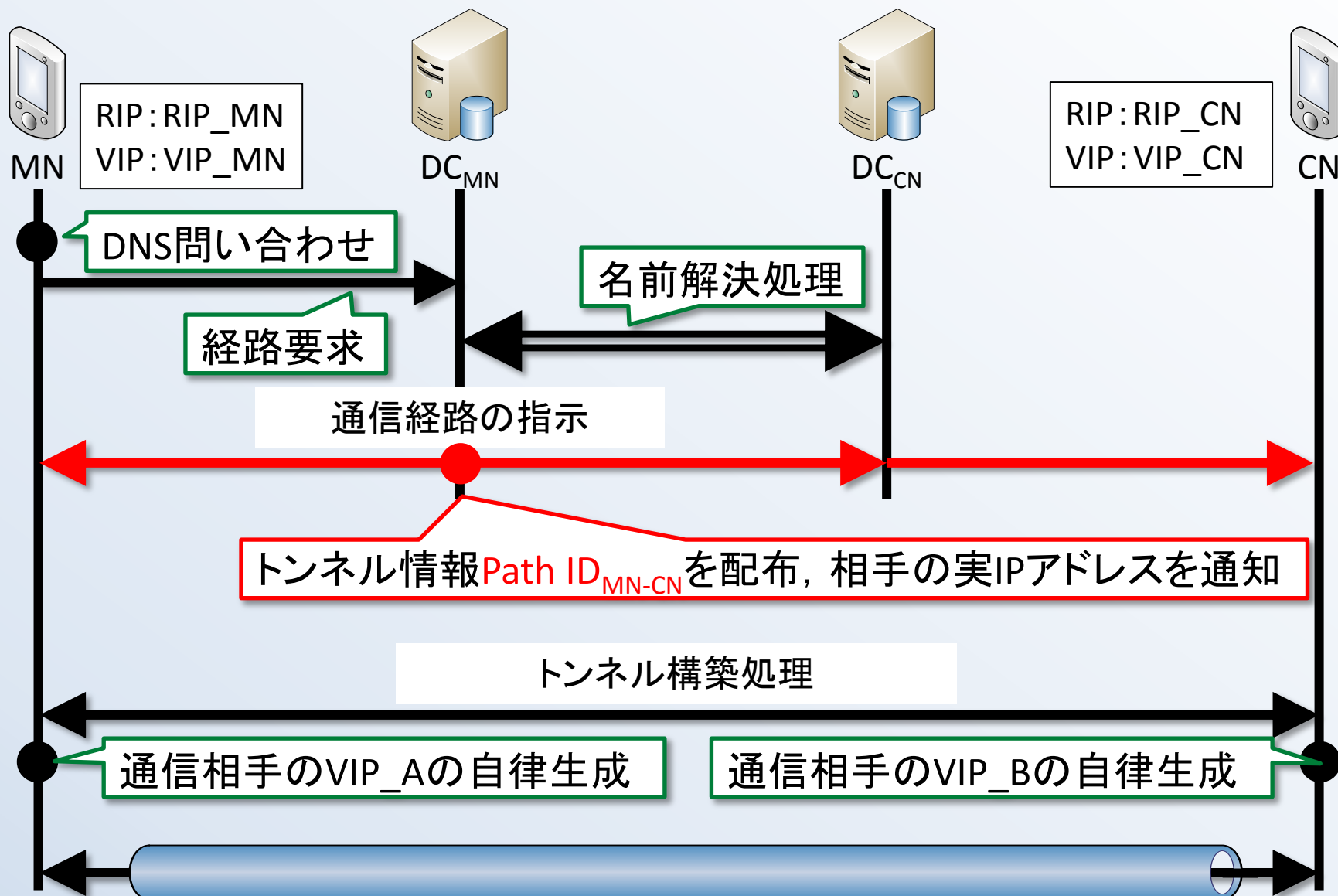




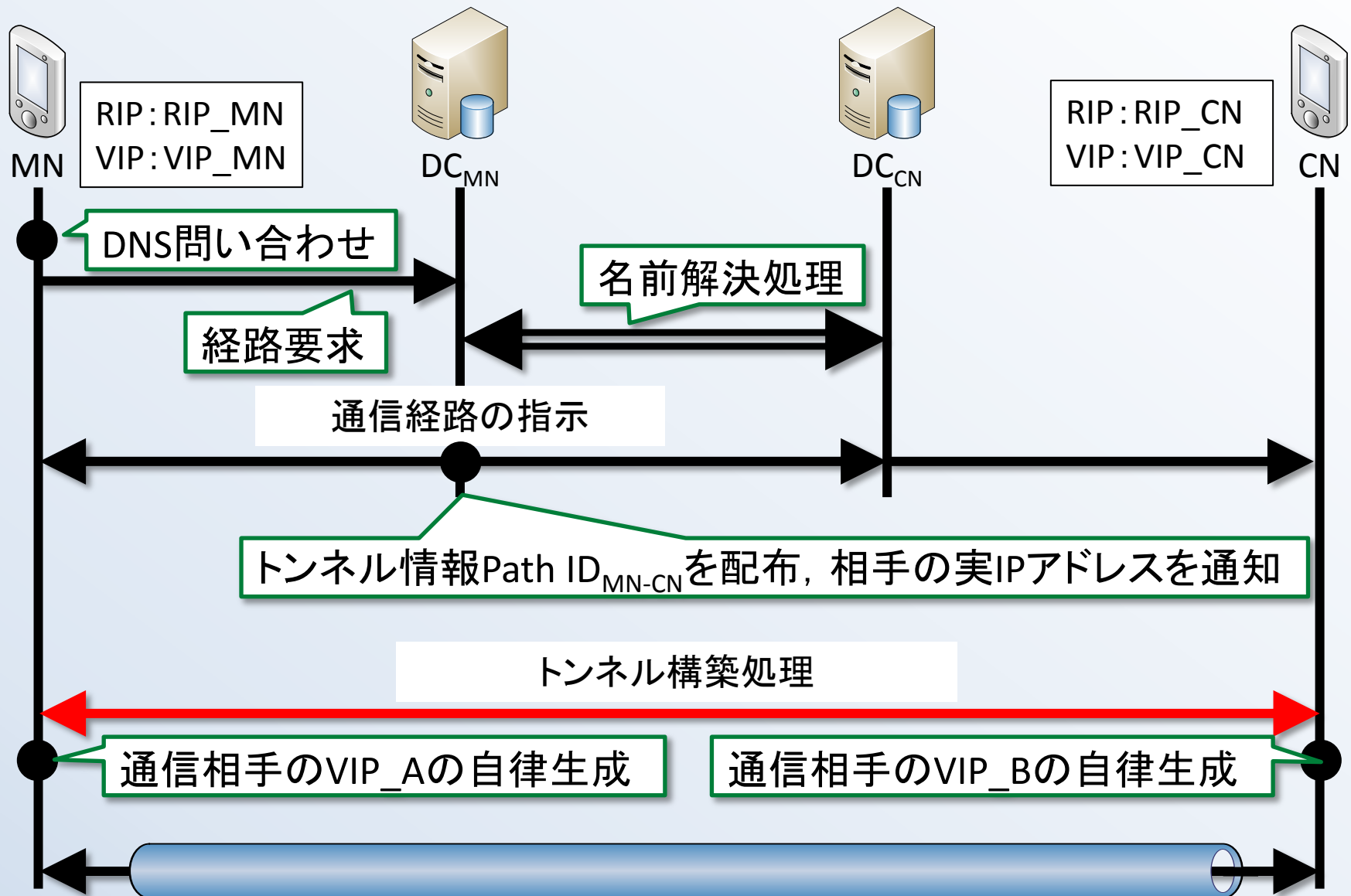
# 通信シーケンス(通信開始時)



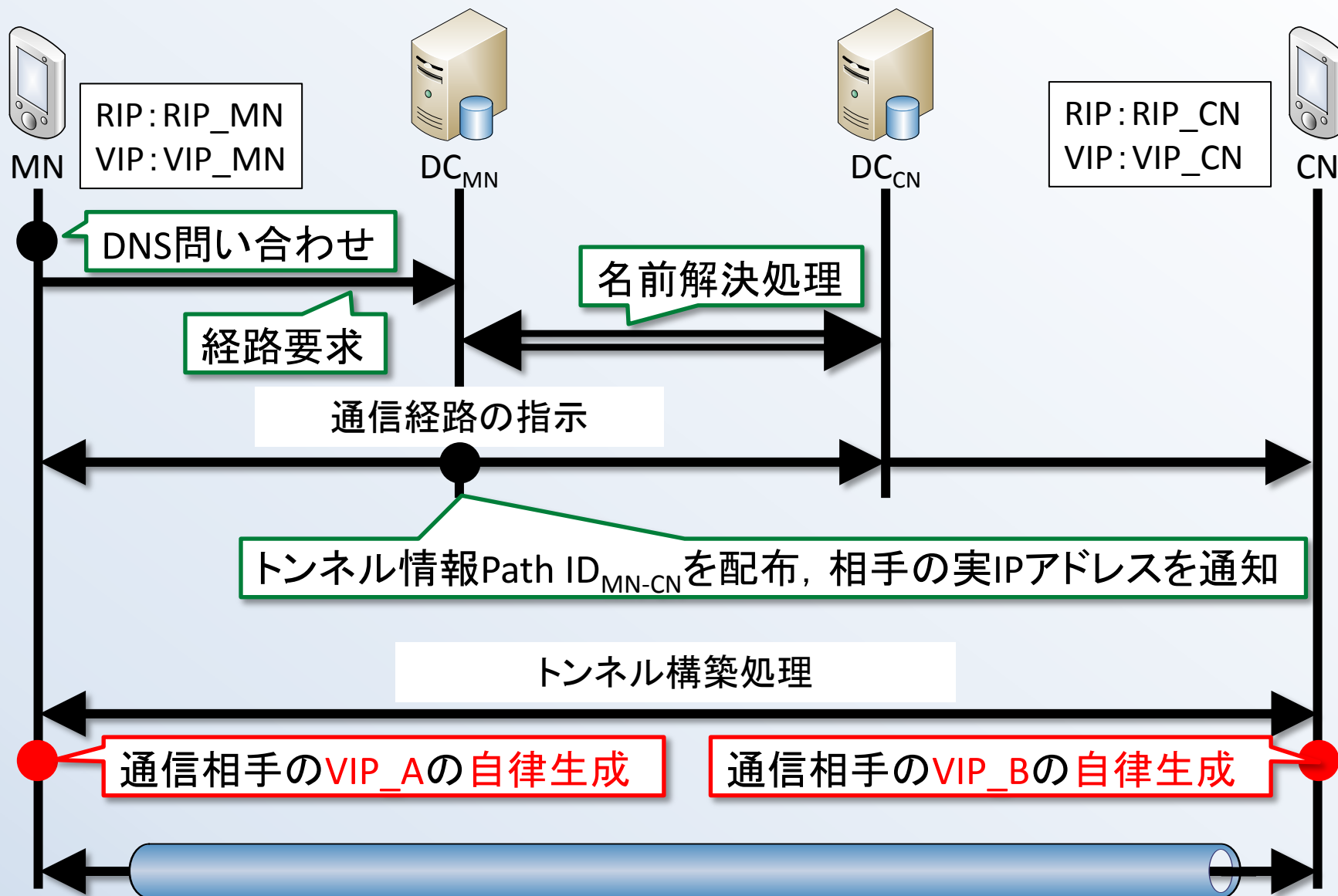
# 通信シーケンス(通信開始時)



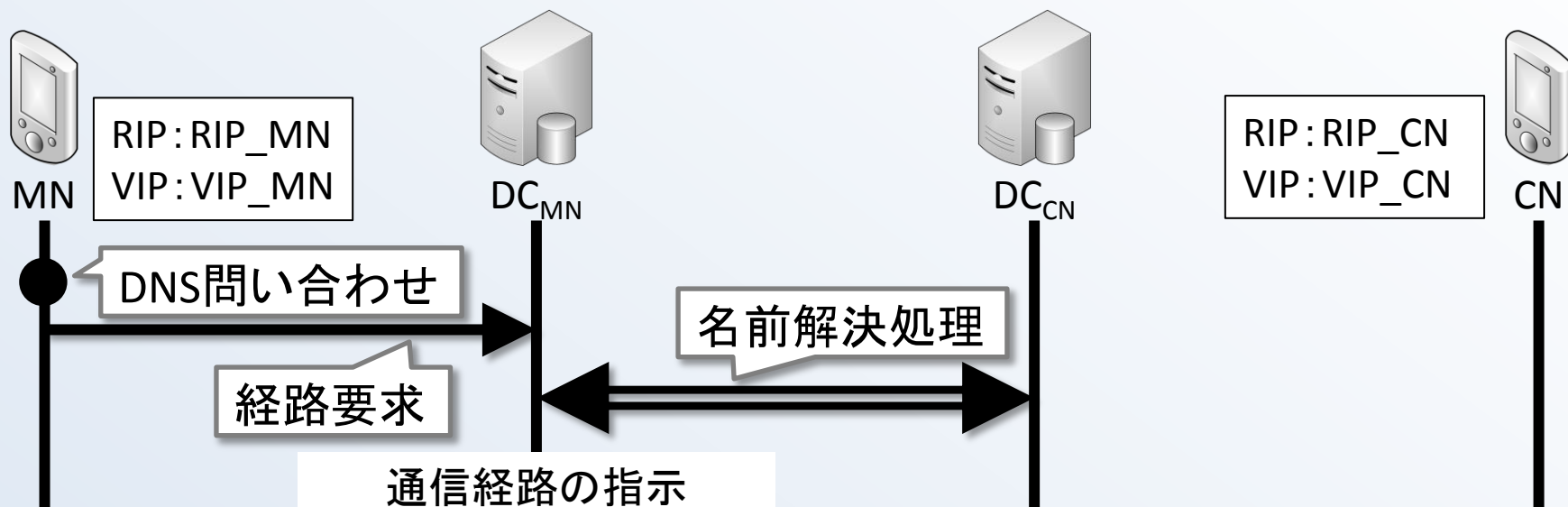
# 通信シーケンス(通信開始時)



# 通信シーケンス(通信開始時)



# 通信シーケンス(通信開始時)



Path IDと対応する仮想IPv4アドレスが決定

MNが管理する情報

| トンネル情報                   | MNのVIP | CNのVIP |
|--------------------------|--------|--------|
| Path ID <sub>MN-CN</sub> | VIP_MN | VIP_A  |

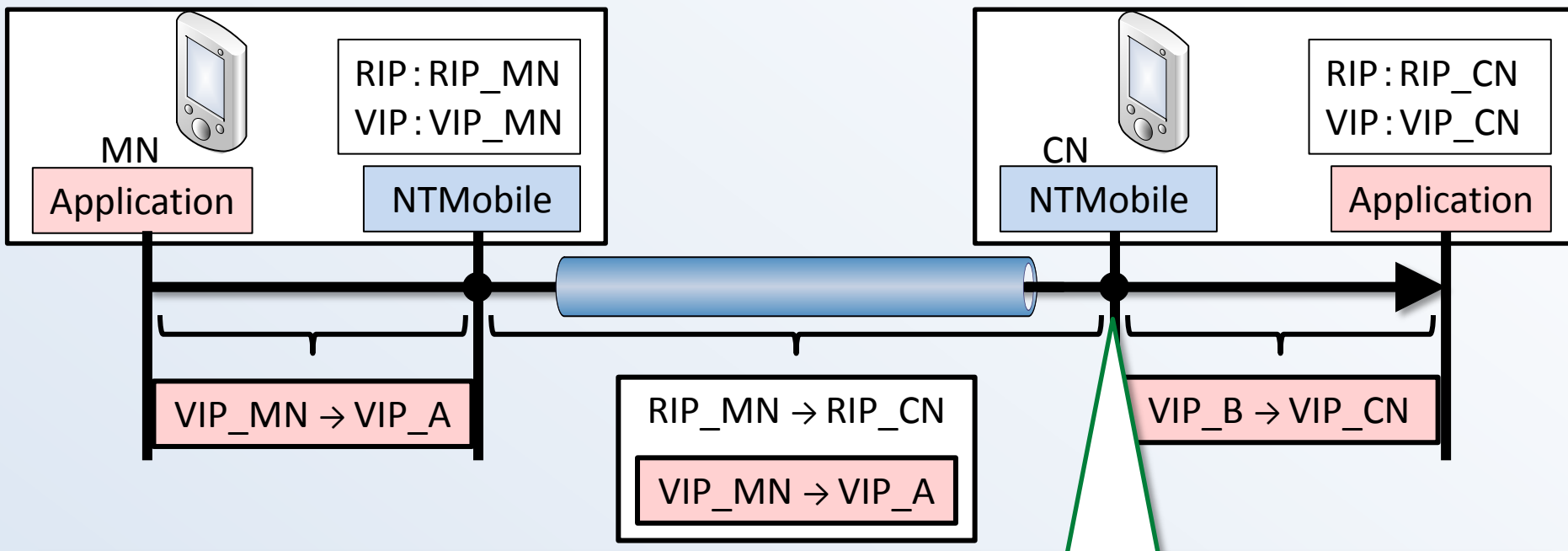
CNが管理する情報

| トンネル情報                   | MNのVIP | CNのVIP |
|--------------------------|--------|--------|
| Path ID <sub>MN-CN</sub> | VIP_B  | VIP_CN |

通信相手のVIP\_Aの自律生成

通信相手のVIP\_Bの自律生成

## ■ IPパケットデカプセル化時に仮想IPv4アドレスを変換



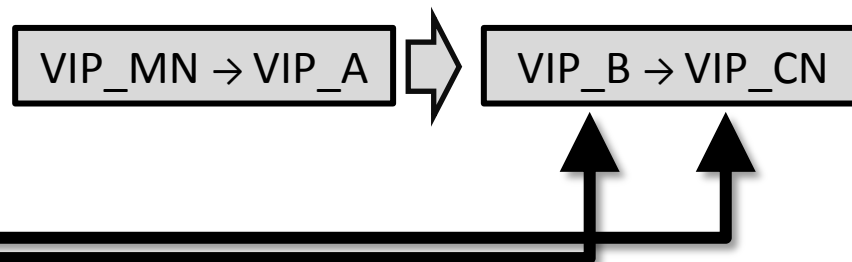
### ① 仮想IPv4アドレスの検索

Path IDから仮想IPv4アドレスを検索

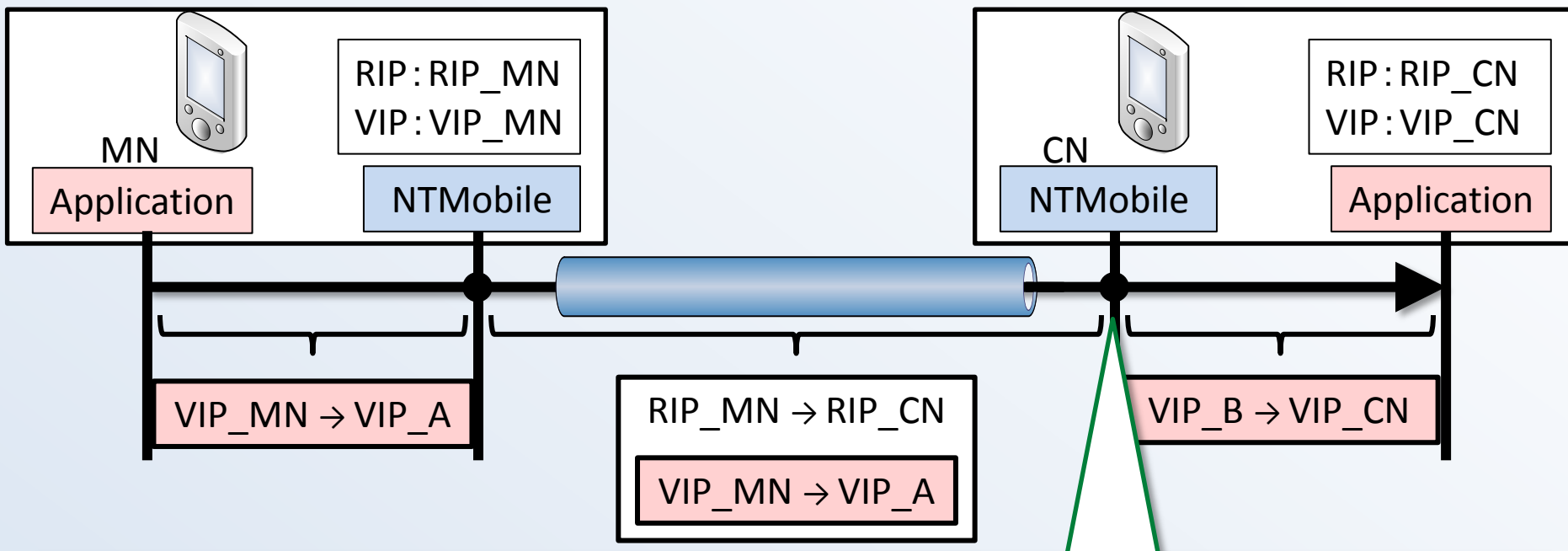
| CNが管理する情報                |        |        |
|--------------------------|--------|--------|
| トンネル情報                   | MNのVIP | CNのVIP |
| Path ID <sub>MN-CN</sub> | VIP_B  | VIP_CN |

### ② 仮想IPv4アドレスの変換

検索結果を元に仮想IPアドレスを変換



## ■ IPパケットデカプセル化時に仮想IPv4アドレスを変換



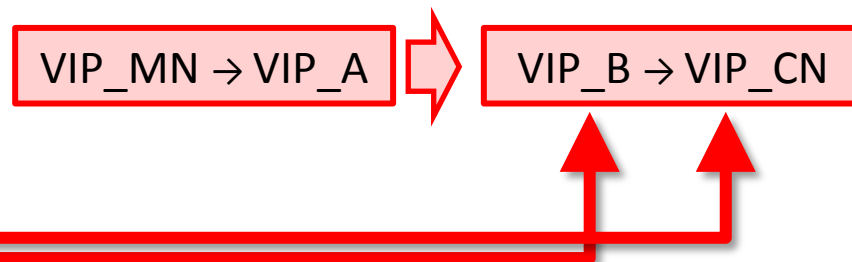
### ① 仮想IPv4アドレスの検索

Path IDから仮想IPv4アドレスを検索

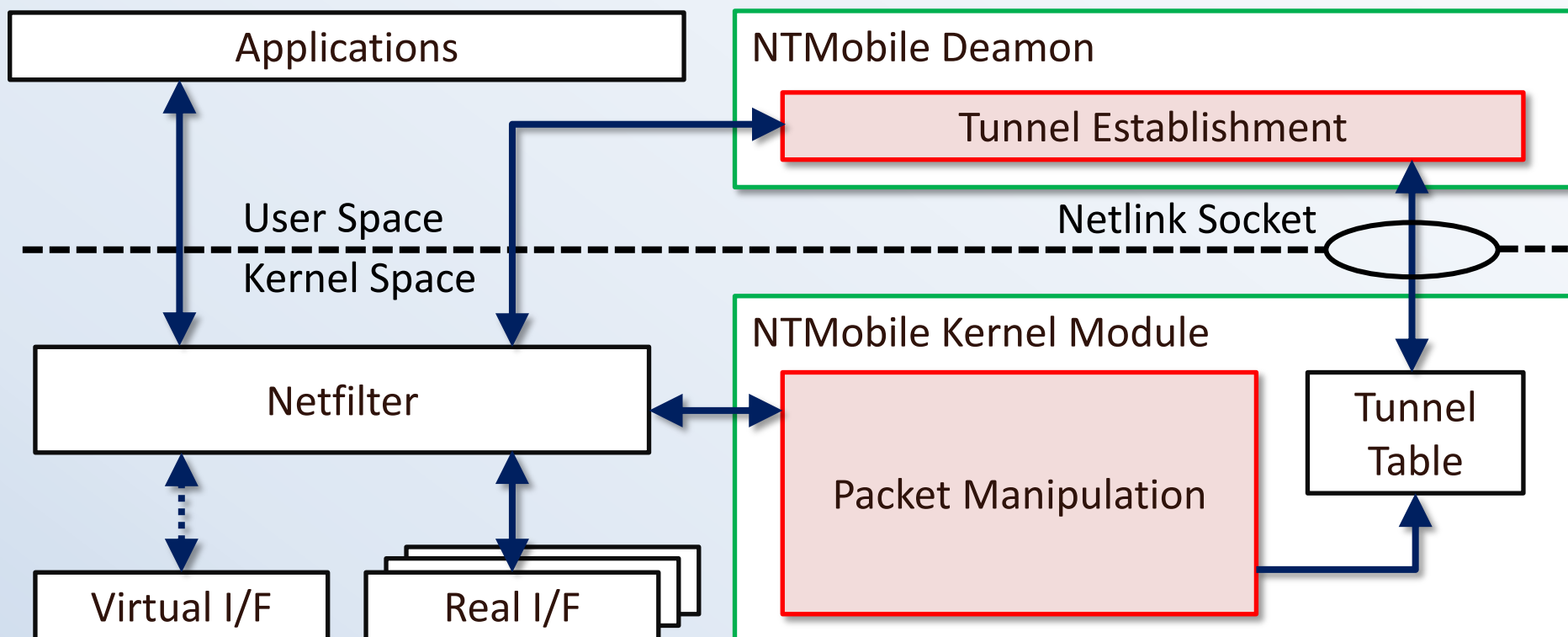
| CNが管理する情報                |        |        |
|--------------------------|--------|--------|
| トンネル情報                   | MNのVIP | CNのVIP |
| Path ID <sub>MN-CN</sub> | VIP_B  | VIP_CN |

### ② 仮想IPv4アドレスの変換

検索結果を元に**仮想IPアドレス**を変換



- NTMobileデーモン
  - 仮想IPv4アドレスの生成処理
  - Path IDによる仮想IPv4アドレスの登録処理
- NTMobileカーネルモジュール
  - 仮想IPv4アドレスの変換処理



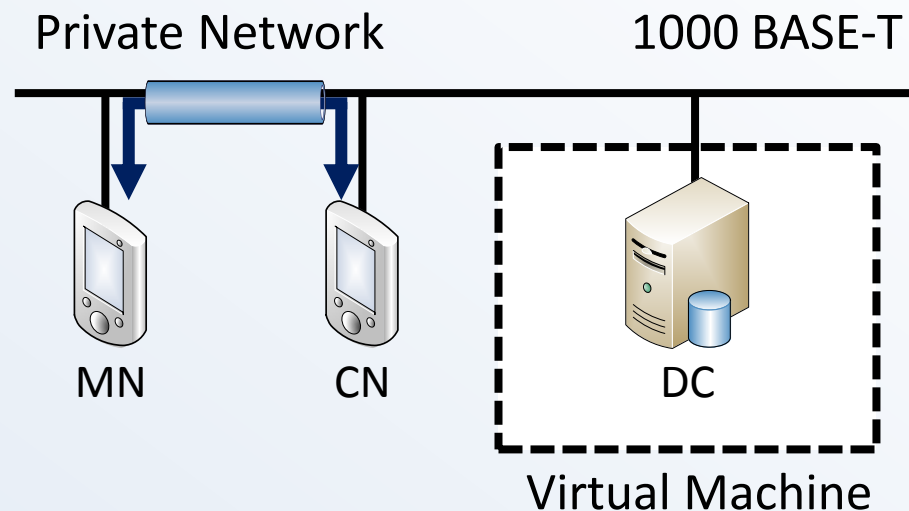


## ■ 提案方式をLinux上に実装

- NTM端末間のトンネル通信の動作を確認

## ■ 測定環境

- Linux PC上にMN, CNを実装
- 仮想マシン上にDCを構築
- 1000 BASE-Tの有線接続



|        | MN                         | CN                         | DC<br>(Host PC)             |
|--------|----------------------------|----------------------------|-----------------------------|
| OS     | Ubuntu 10.04               | Ubuntu 10.04               | Windows 7                   |
| CPU    | Intel Core i7-860(2.80GHz) | Intel Core i7-930(2.80GHz) | Intel Core i7-2600(3.40GHz) |
| Memory | 3GB                        | 3GB                        | 8GB                         |

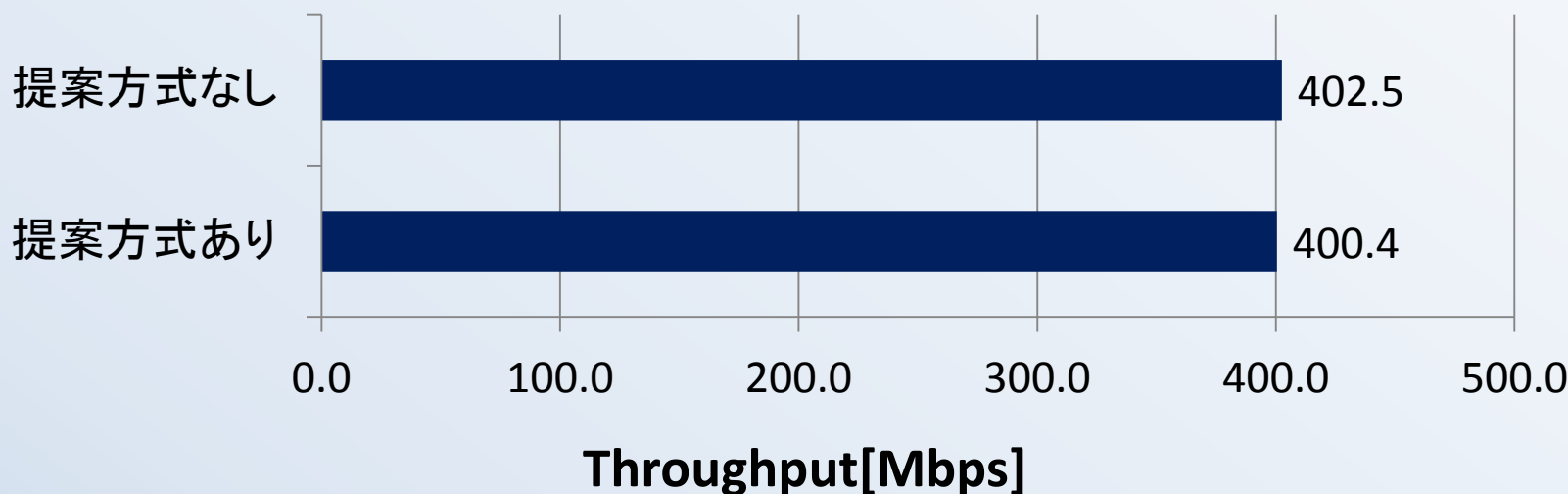
|             | DC<br>(VM)   |
|-------------|--------------|
| OS          | Ubuntu 10.04 |
| CPU<br>割り当て | 1コア          |
| Memory      | 1GB          |

## ■ トンネル通信のスループットを測定

- iperfを用いたTCP通信を行い, MN~CN間のスループットを測定

## ■ スループットの低下率は0.5%

- アドレス変換処理の影響はほとんどないことを確認



- 仮想IPv4アドレスの運用
  - 端末内で仮想IPv4アドレスを自律生成・管理
  - Path IDを用いて, 仮想IPv4アドレスを対応付け
  
- 実装と評価
  - Linux上で提案方式の動作を検証
  - 性能が劣化しないことを確認
  
- 今後の予定
  - 提案方式と一般端末との通信の検討