

OSに依存しないNTMobile通信ライブラリの提案

岡本 昂輝^{†*}, 鈴木 秀和[†], 内藤 克浩[‡], 渡邊 晃[†] ([†]名城大学, [‡]愛知工業大学)

Proposal of NTMobile Communication Library which does not depend on OSes

Koki Okamoto^{†*}, Hidekazu Suzuki[†], Katsuhiro Naito[‡], Akira Watanabe[†] ([†]Meijo University, [‡]Aichi Institute of Technology)

1 はじめに

インターネットが急速に普及し、社会のインフラとなっている。しかし、現在の IP ネットワークには、様々な制約があり自由に通信ができない。これらの制約を解決するために我々は、NTMobile(Network Traversal with Mobility) を提案している。NTMobileは、通信相手を名前で指定すると必ず通信経路を確立することができ、また移動透過性の実現が可能である。NTMobileは、Linux と Android で実装と検証がされている。しかし、これまで OS の違いや利用用途ごとにライブラリを修正してきたため、NTM ライブラリが複数存在している。そこで本研究では、NTM ライブラリから OS 依存の部分を抜き出し、あらゆる OS での利用を可能にする方式を提案する。

2 NTMobileの概要

NTMobile[1] は、インターネット上に通信経路を指示する DC(Direction Coordinator) とエンドツーエンドの直接通信が行えない場合にパケットの中継を行う RS(Relay Server) を設置する。ユーザは NTMobile 用アプリケーションをエンド端末に搭載するだけで NTMobile の利用が可能となる。NTMobile では、ネットワークを切り替えることが可能な移動透過性の実現、およびエンドツーエンド通信のセキュリティの保証が可能である。NTMobile は、NTM 端末に対して位置に依存しない仮想 IP アドレスを割り当て、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う。この仮想 IP アドレスに基づくパケットは、端末の実 IP アドレスでカプセル化され通信相手に送信される。

これまで NTMobile は、カーネル改造方式、ライブラリ方式、TUN/TAP 方式の 3 つの方式で実現してきた。しかし、カーネル改造方式ではカーネルの変更に管理者権限が必要であり、普及が難しい。ライブラリ方式では、アプリの新規開発が必要であるという課題がある。TUN/TAP 方式では、このような課題がなく、上記 2 つの方式に比べ有用である。Linux および Android では、TUN/TAP 方式を採用し動作検証がされている。しかし、Linux と Android は統合されておらず異なる NTM ライブラリを使用している。これは、OS に依存する処理を明確に分離できていなかったことが原因である。

3 提案方式

TUN/TAP 方式の NTM ライブラリを統合し、Linux, Android, iOS, Windows で共通に利用できる NTM ライブラリを提案する。そのため、NTMobile としての機能のみを NTM ライブラリに入れ、OS 固有の機能を外に抜き出す。提案する NTM ライブラリの構成を Fig. 1 に示す。VNIC(Virtual Network Interface Card) は、OS が提供する仮想的通信インタフェースであり、PNIC(Physical NIC) は、物理インタフェースである。NTM アプリケーションは OS 上に一般アプリケーションと同じようにインストールされる。NTMobile を利用する一般アプリケーションは、VNIC 宛にパケットを送信することにより、IP パケットが生成される。VNIC の機能により IP パケットが NTM アプリに渡され、NTM アプリケーショ

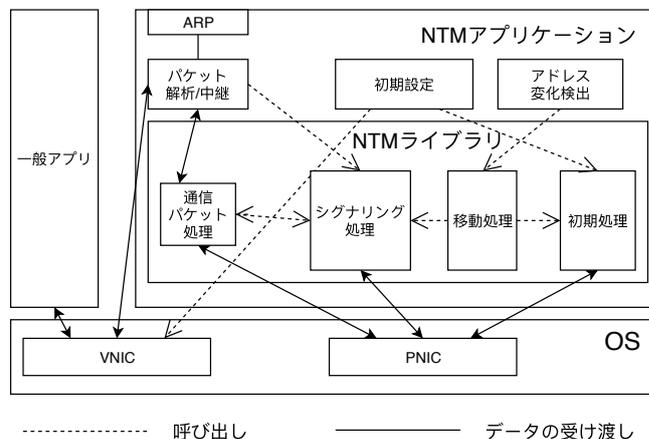


Fig. 1 NTM ライブラリの構成

ンの処理を経て PNIC から送信される。

NTM ライブラリは、DC への登録を行う初期処理、DNS クエリをトリガとしてトンネル経路を生成するシグナリング処理、一般通信パケットを NTMobile パケットに変換する通信パケットからなる。初期設定は、VNIC を定義し NTM ライブラリの初期処理を起動する。VNIC の設定方法は OS によって異なるため NTM ライブラリの外に出す。パケット解析/中継は、IP アドレスの変化を検出してシグナリング処理を起動するか通信パケット処理に渡すかを判定する。OS に依存する VNIC と NTM ライブラリとの中継役として NTM ライブラリの外に出す。アドレス変化検出は、通信中に IP アドレスの変化を検出してシグナリング処理を起動し、通信を継続する。アドレス変化の検出方法は、OS により方法が異なるため NTM ライブラリの外に出す。ARP モジュールは、Windows 特有の処理である。Windows では、Linux の TUN に相当するサービスがないため、TAP に相当するサービスを利用する必要がある。TAP は、ARP の処理が必要であるため、VNIC から ARP request が来た場合は、ARP reply を返答する必要がある。NTM ライブラリは性能と移植性を考慮し、C で記述する。NTM アプリケーションの記述言語は、Linux は C、Android は Java、iOS は SWIFT、Windows は C とする。Java から NTM ライブラリを呼ぶ際にはラッパーが必要となる。SWIFT は、C と互換性があるため、ラッパーは不要である。以上の機能分担により、NTM ライブラリを統合できる。

4 まとめ

本稿では、NTM ライブラリを共通化し、あらゆる OS に適用できる方式を提案した。今後は、Windows への移植方法を中心に検討を進める。

文献

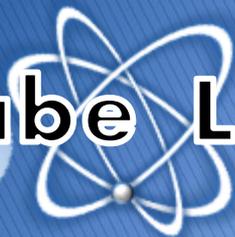
[1] 納堂. 他: エンドツーエンド通信をアプリケーションレベルで可能にする通信ライブラリの実現と評価, 情報処理学会論文誌 Vol.9, No.1, pp.1-11, Jan.2019.



OSに依存しないNTMobile 通信ライブラリの提案

岡本 昂輝⁺, 鈴木 秀和⁺, 内藤 克浩[‡], 渡邊 晃⁺
⁺名城大学 工学部
[‡]愛知工業大学 情報科学部

Watanabe Lab.

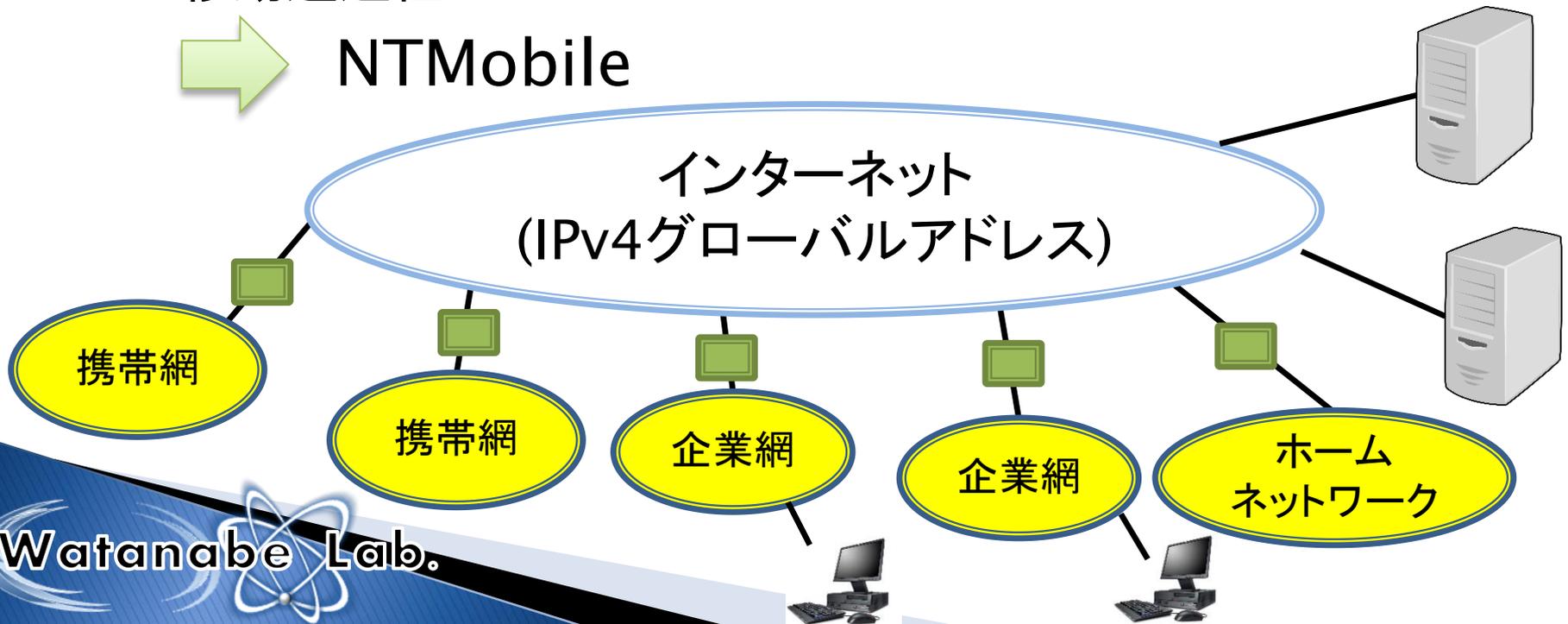


研究背景

- ▶ インターネットが普及し様々な制約が発生している
 - IPアドレス枯渇問題
 - NAT越え問題
 - IPv4アドレスとIPv6アドレスの非互換性
 - 移動透過性

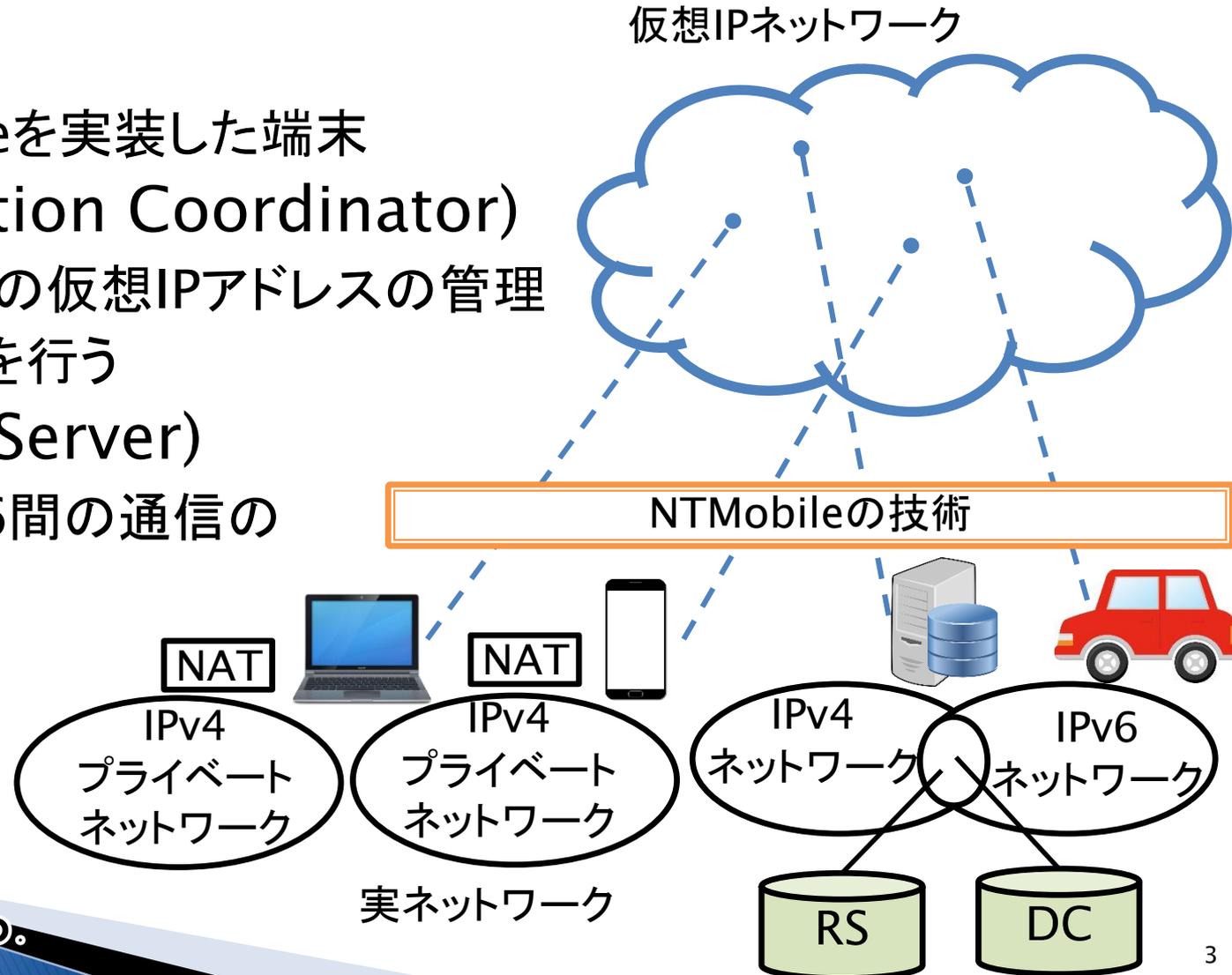


NTMobile



NTMobileの構成

- ▶ NTM端末
 - NTMobileを実装した端末
- ▶ DC(Direction Coordinator)
 - NTM端末の仮想IPアドレスの管理
 - 経路指示を行う
- ▶ RS(Relay Server)
 - IPv4-IPv6間の通信の補助



NTMobileの現状

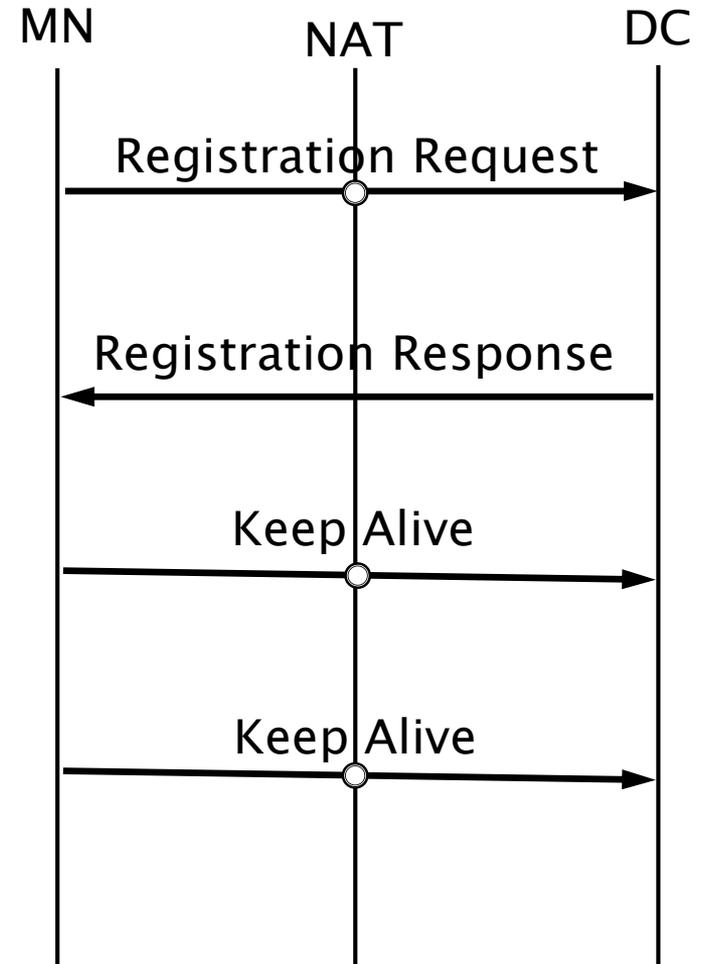
- ▶ Linux及びAndroidで動作検証済み
 - 各々の用途にあわせて開発してきたため、統合しきれていない
- ▶ 他のOSに容易に実装することが困難
 - Windows及びiOSへの導入が望まれている



NTMライブラリの統合

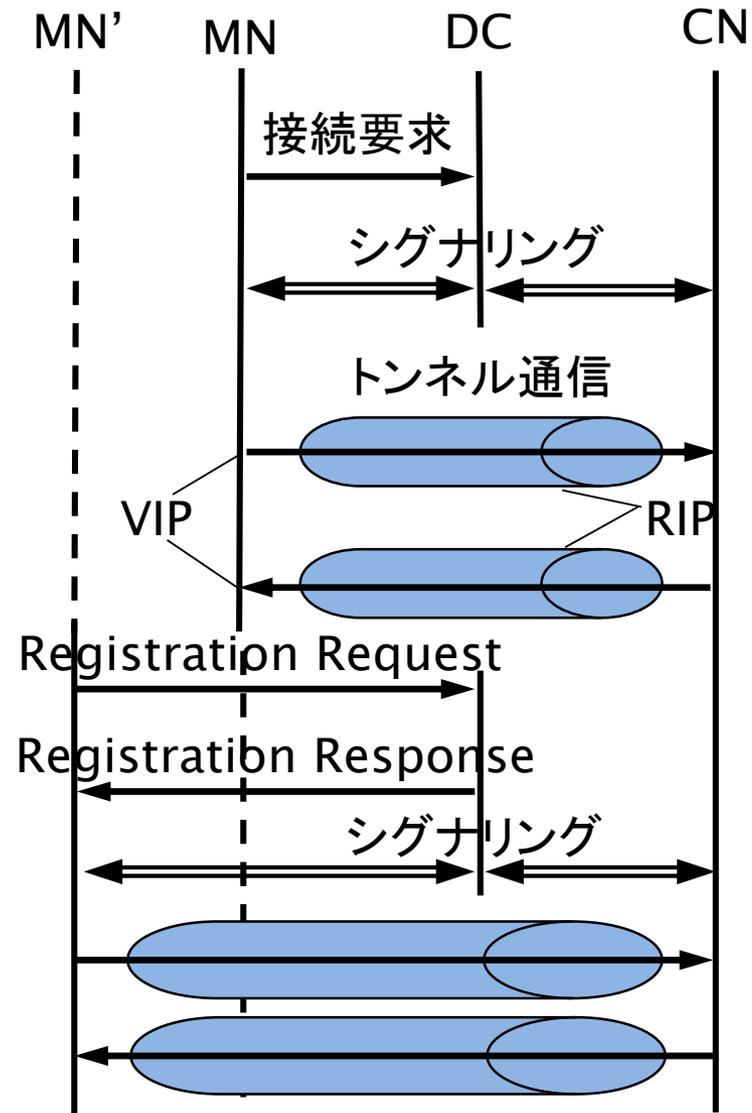
NTMobileの動作

- ▶ 立ち上げ処理
 - MNはプライベートネットワーク
 - DCはグローバルネットワーク
- ▶ Keep Alive
 - NATテーブルの維持
 - 20秒ごとに送信

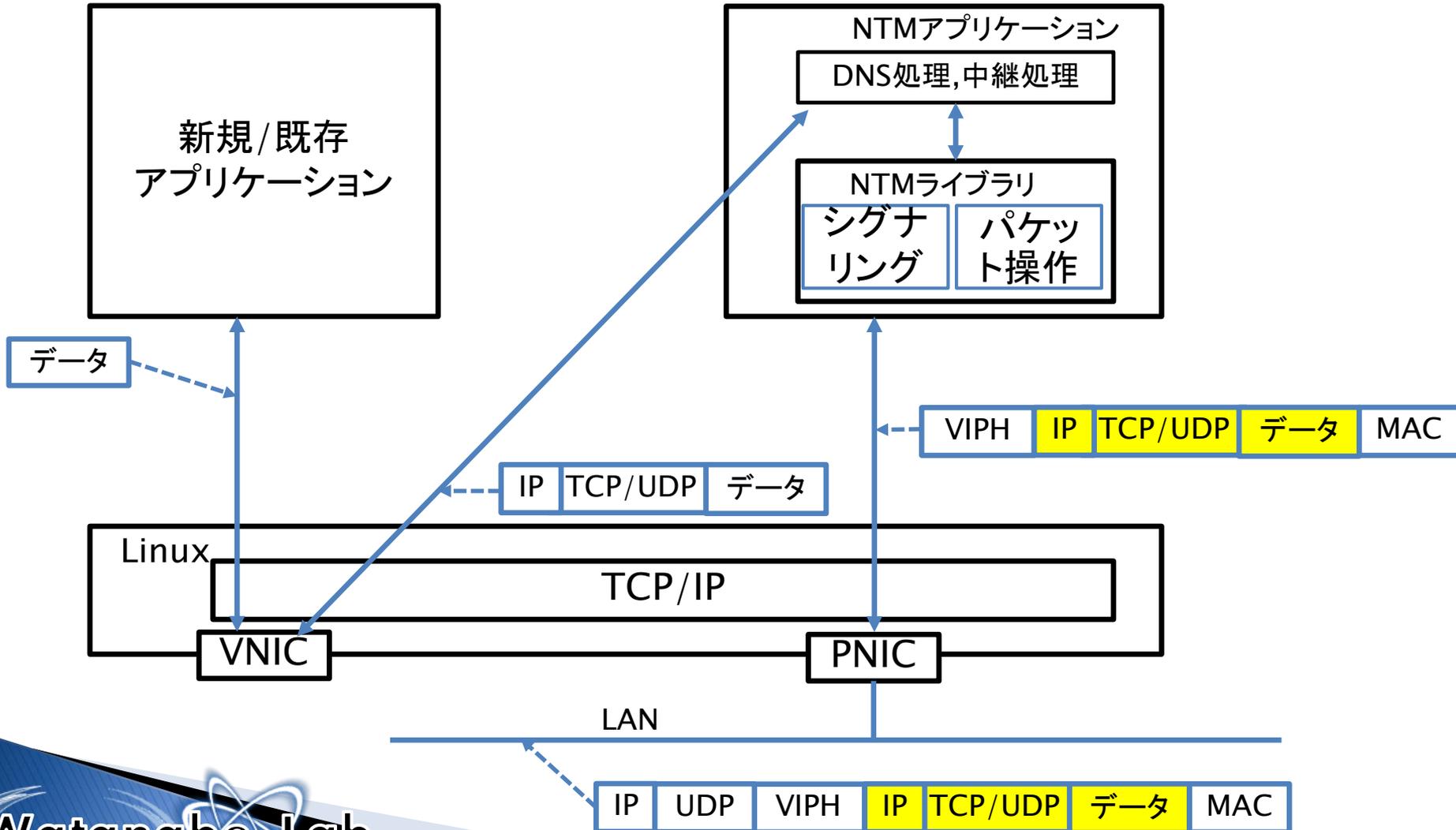


NTMobileの動作

- ▶ 接続要求
 - CNのFQDNを指定
- ▶ シグナリング
 - MNとCNに経路を指示
- ▶ トンネル通信
 - エンドツーエンドの暗号通信
- ▶ 移動時の処理



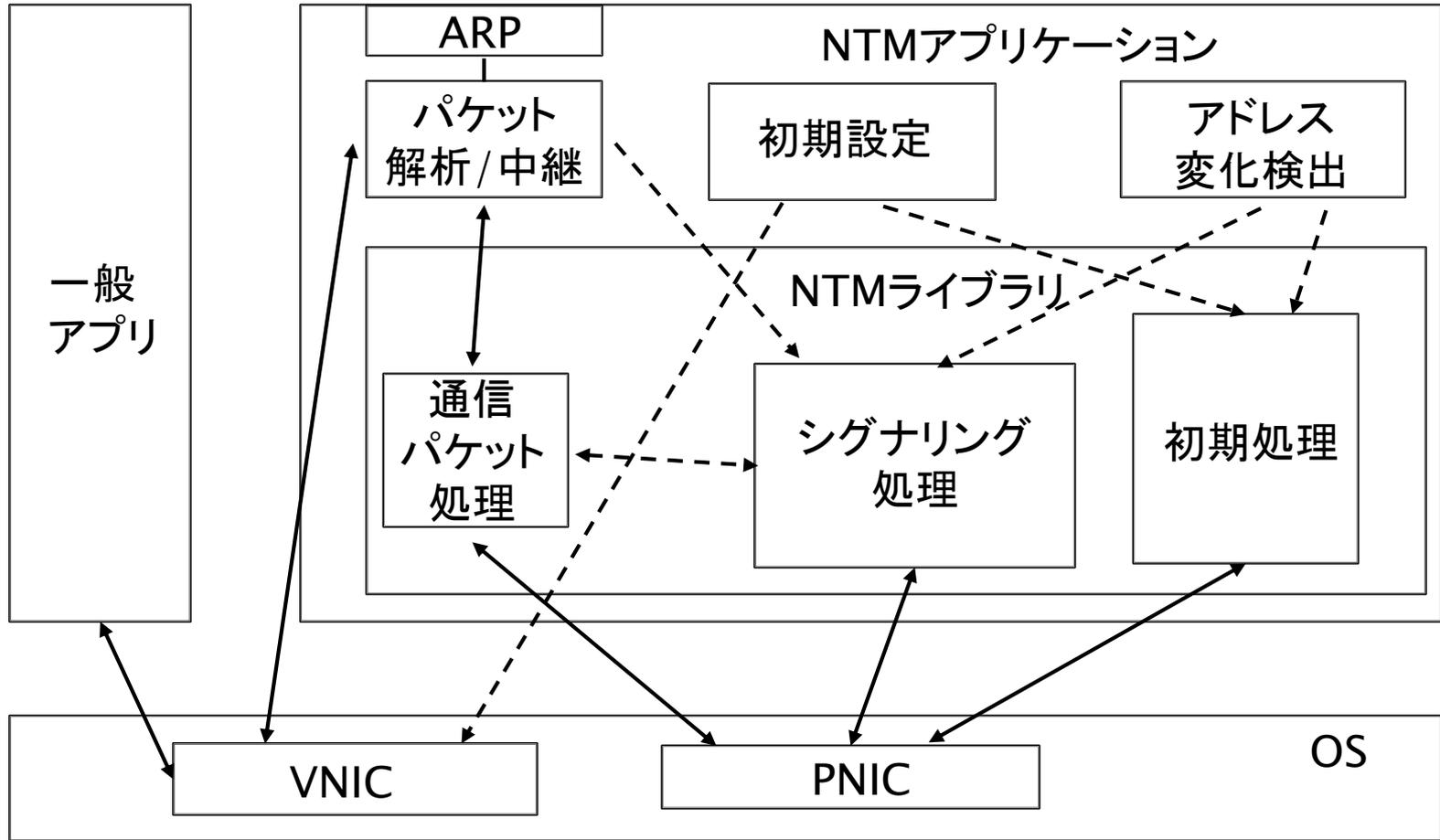
NTMobileの実現方式



統合に向けての検討

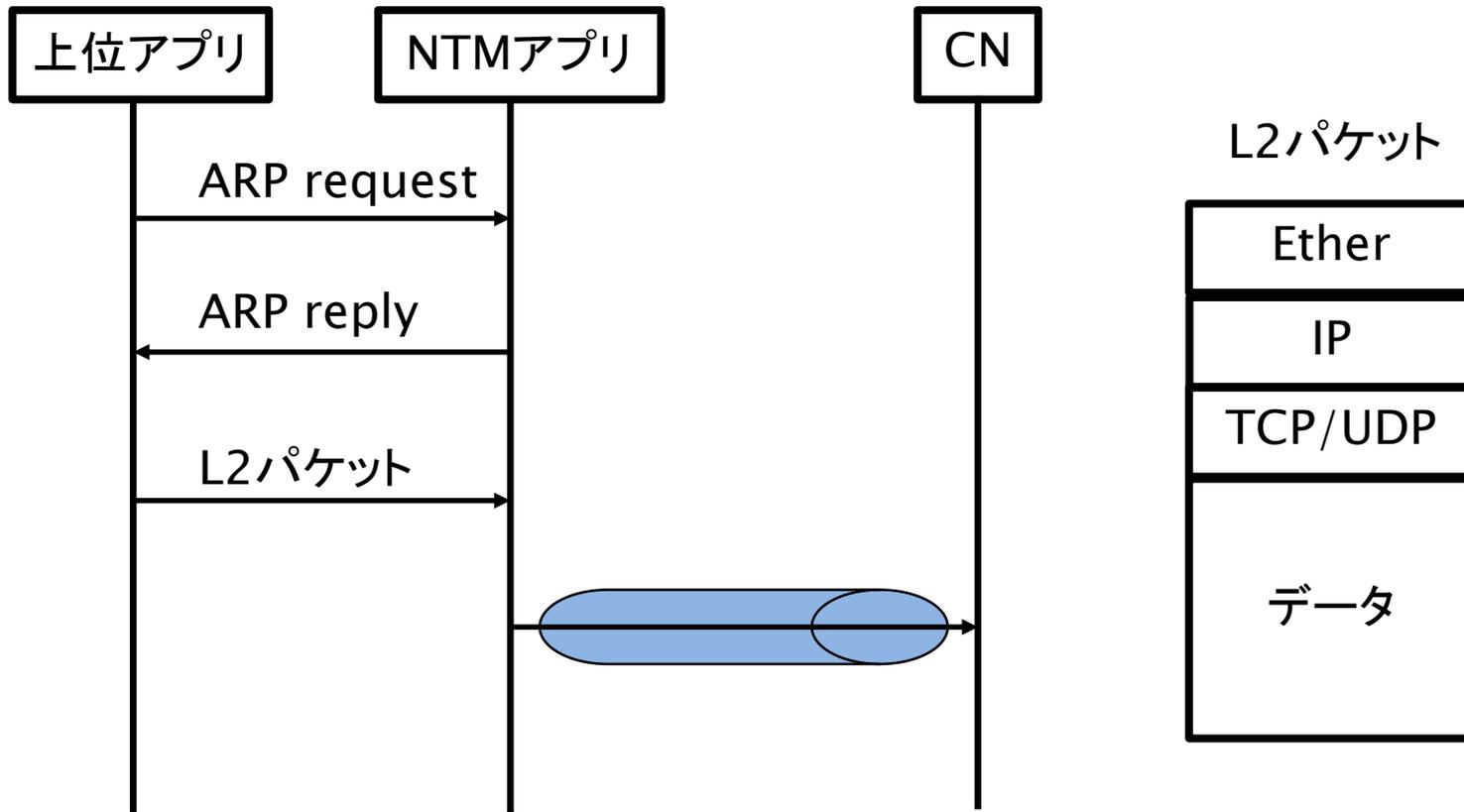
- ▶ NTMobileを各OS固有の部分と共通の部分に分ける
 - OSに依存する処理を明確にする
 - NTMobileとしての機能をNTMライブラリに入れる
 - Windows特有の処理を検討

モジュール構成の詳細



Windows特有の処理(ARP)

▶ ARPの処理



まとめ

- ▶ OSに依存しないNTMobile通信ライブラリを検討
 - OSに依存する部分
 - NTMobileとしての機能

- ▶ 今後の方針
 - Windowsへの移植を中心に検討
 - ライブラリ型との統合を検討