NTMobileを用いたアプリケーション層での移動透過性の実現

130441001　赤堀　蒼磨
渡邊研究室

1. はじめに

モバイル端末の普及により, 手軽にインターネットに接続ができるようになった. モバイル端末においては, 接続するネットワーク環境にかかわらず通信を開始できる通信接続性と通信中にネットワークを切り替えることができる移動透過性の需要が高い. 我々は, 両者を同時に実現可能とする NTMobile(Network Traversal with Mobility)[1] を提案している. これまでは Linux カーネルにて NTMobile の機能を実装していたが, モバイル端末で利用するにはアプリケーション層での実装が必要になるため移植を行っている. 本稿では, アプリケーション層での実装のうち, 特に移動透過性に係る部分の実装及び性能評価について報告する.

2. NTMobile

NTMobile はインターネット上に, DC(Direction Coordinator) と呼ぶ装置を設置する. NTMobile を搭載した端末 (NTM 端末) がログインした際に, DC は重複しない仮想 IP アドレスを生成し NTM 端末に配布する. NTM 端末間ではこの仮想 IP アドレスにて通信セッションを確立する. DC は NTM 端末の位置をすべて把握しており, 通信の開始または NTM 端末の移動を検出した際, NTM 端末に対し UDP トンネル構築の経路指示を行う. NTM 端末間の通信パケットは全て実 IP アドレスでカプセル化してトンネル通信を行う. NTMobile は当初 Linux カーネル上にて実装を行い動作検証を行った. このシステムを iOS や Android 端末で動作させるためアプリケーション層に移植し, 実用化を目指しているところである.

3. 従来の実装方式

これまでの NTMobile はパケットの暗号化とカプセル化処理, アドレス変化検出処理を Linux カーネル空間で実装している. この実装では, NTM 端末のネットワークが切り替わった時の処理が未実装であった. DC は通信開始時に通信識別子である Path ID を生成し, MN と CN に対して通知する. MN と CN はこの Path ID で処理を識別する方法をとっていた.

4. 新たな実装方式

新たな実装方式ではすべての処理をアプリケーション層で提供する. ネットワーク切り替え時の処理を実装するため, 実 IP アドレスと仮想 IP アドレスの関係を記述したトンネルテーブルをダイナミックに書き換える必要がある. そのためにトンネルテーブルの読み込み及び書き込みにより, データの整合性が取れない状況を避けるため排他処理の実装が必要になる. 新たな実装方式では複数の Key で一つのデータを参照することができるハッシュ機能を利用してトンネルテーブルを生成している. このハッシュ機能の利用を前提とすると, 一連の通信に対し 1 つの Path ID を利用し続ける必要がある. 従来の実装では Path ID を DC で生成していたため, NTM 端末ネットワーク切替時に別の Path ID が割り当てられることとなり, トンネルテーブルの更新が適切に行えなかった. そこで, 新たな実装方式では Path ID をエンド端末側で生成するように見直した. また, 従来の実装ではアドレス変化検出処理をカーネル内で行っていたが, アプリケーション側から 1 秒に一回カーネルに問い合わせる形に変更をした. これらの処理によりカーネル空間で実装していた機能を全てアプリケーション上で実現し, カーネル空間がブラックボックスとなっている OS でも NTMobile を利用することが可能となった.

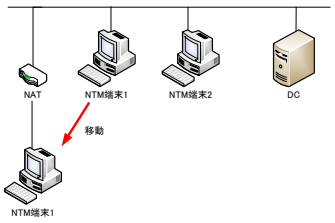


図 1: 評価用のネットワーク構成

表 1: 処理時間

全体処理時間	7.001702
アドレス取得時間	0.007133
アドレス変化検出時間	6.910756
ハンドオーバ処理時間	0.083813

5. 実装と評価

移動部分に係る処理についての実装を行った. タイマーによりアドレスを監視するスレッドを作成した. また, 排他処理を伴うトンネルテーブルの書き換え処理を実装した. 以上の実装により安定した移動透過性の実現ができることを確認した.

次に実装結果を評価するため, 処理時間の内訳を調査した. 評価用ネットワーク構成を図 1 に示す. NTM 端末 1, NTM 端末 2, DC, NAT を同一ネットワークに設置する. 通信開始後 NTM 端末 1 を NAT 配下に移動する. その後 NTM 端末 1 が NAT からアドレスを取得するまでの時間, アドレス確定から NTMobile がアドレス変化を検出するまでの時間, アドレスを検出してからハンドオーバ処理が終わるまでの時間を 10 回計測しそれぞれの平均値を算出した. 実験により得られた結果を表 1 に示す. 実験結果より, アドレスを検出するまでの時間が処理の大部分を占めていることが分かった.

6. まとめ

NTMobile によりアプリケーション上で移動透過性を実現することに成功した. 今後はアドレス変化検出時間を短縮させる方法を検討していく必要性がある.

参考文献

[1] 内藤克浩, 西尾拓也, 水谷智大, 鈴木秀和, 渡邊晃, 森香津夫, 小林英雄:　NTMobile における移動透過性の実現と実装, DICOMO2011 論文集, Vol. 2011, pp.1349-2359 (2011).

NTMobile を用いたアプリケーション層での移動透過性の実現

学籍番号:130441001

渡邊研究室　赤堀 蒼磨

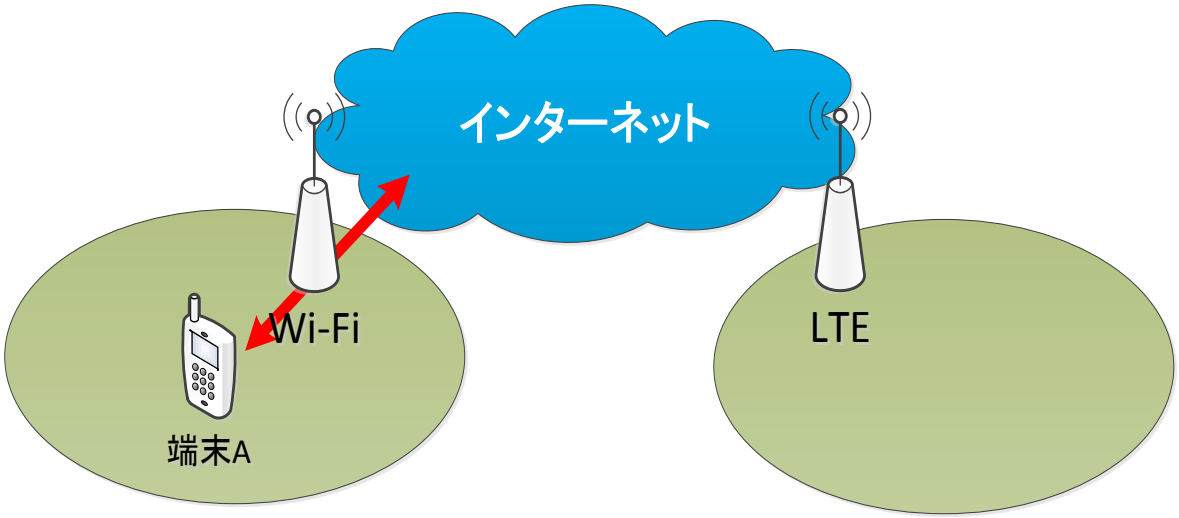
Watanabe Lab.

研究背景

- スマートフォンやタブレット端末の普及
- Wi-Fi↔LTEやWi-Fi↔3Gでネットワークが切り替わると

  IPアドレスが変化　→　通信切断

  - スマートフォンでは特に切り替えが頻繁

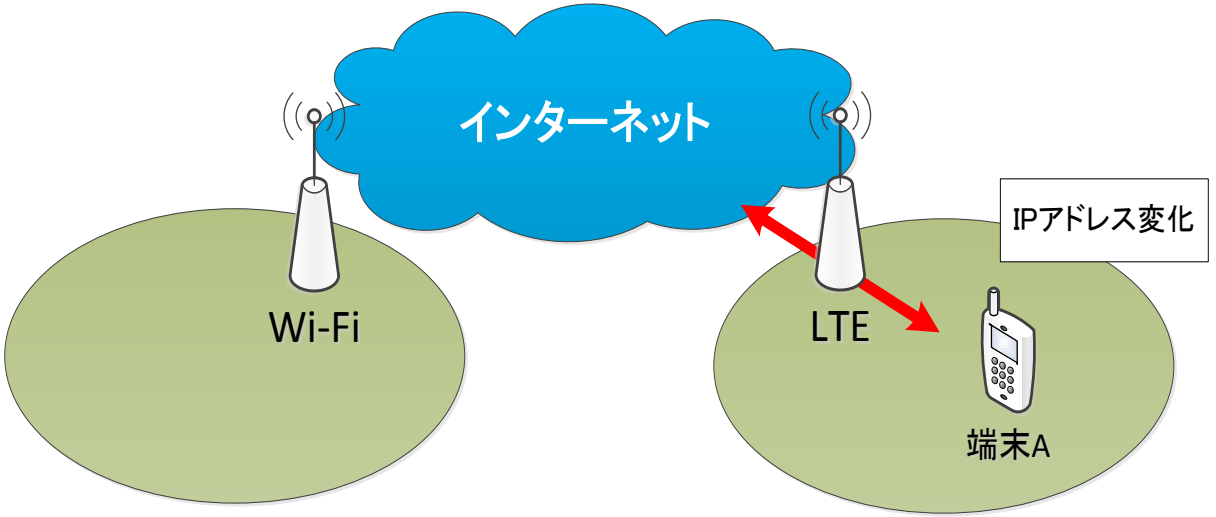


研究背景

- スマートフォンやタブレット端末の普及
- Wi-Fi↔LTEやWi-Fi↔3Gでネットワークが切り替わると

  IPアドレスが変化　→　通信切断

  - スマートフォンでは特に切り替えが頻繁

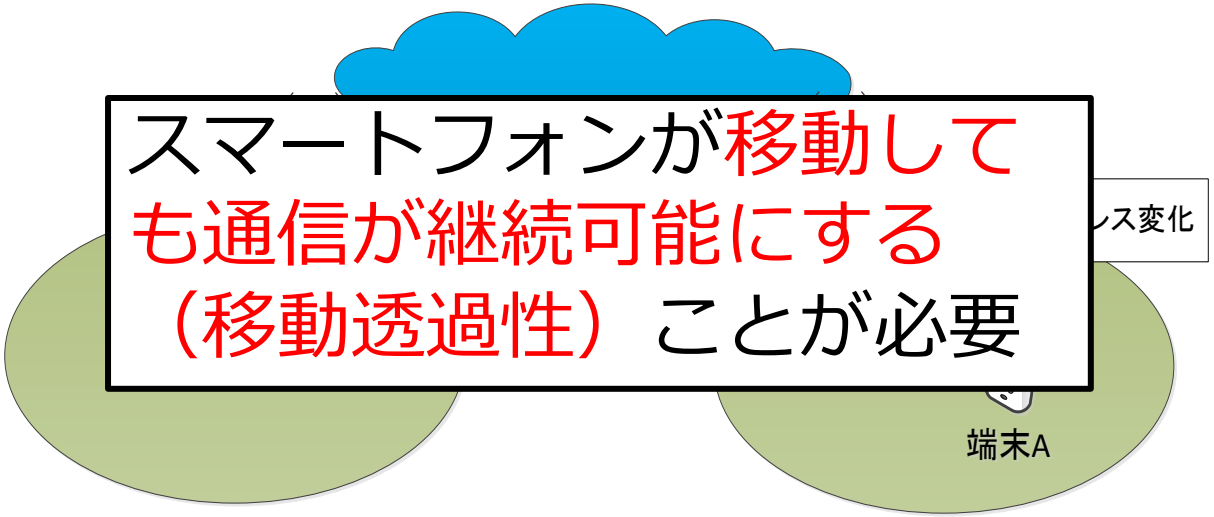


研究背景

- スマートフォンやタブレット端末の普及
- Wi-Fi↔LTEやWi-Fi↔3Gでネットワークが切り替わると

  IPアドレスが変化　→　通信切断

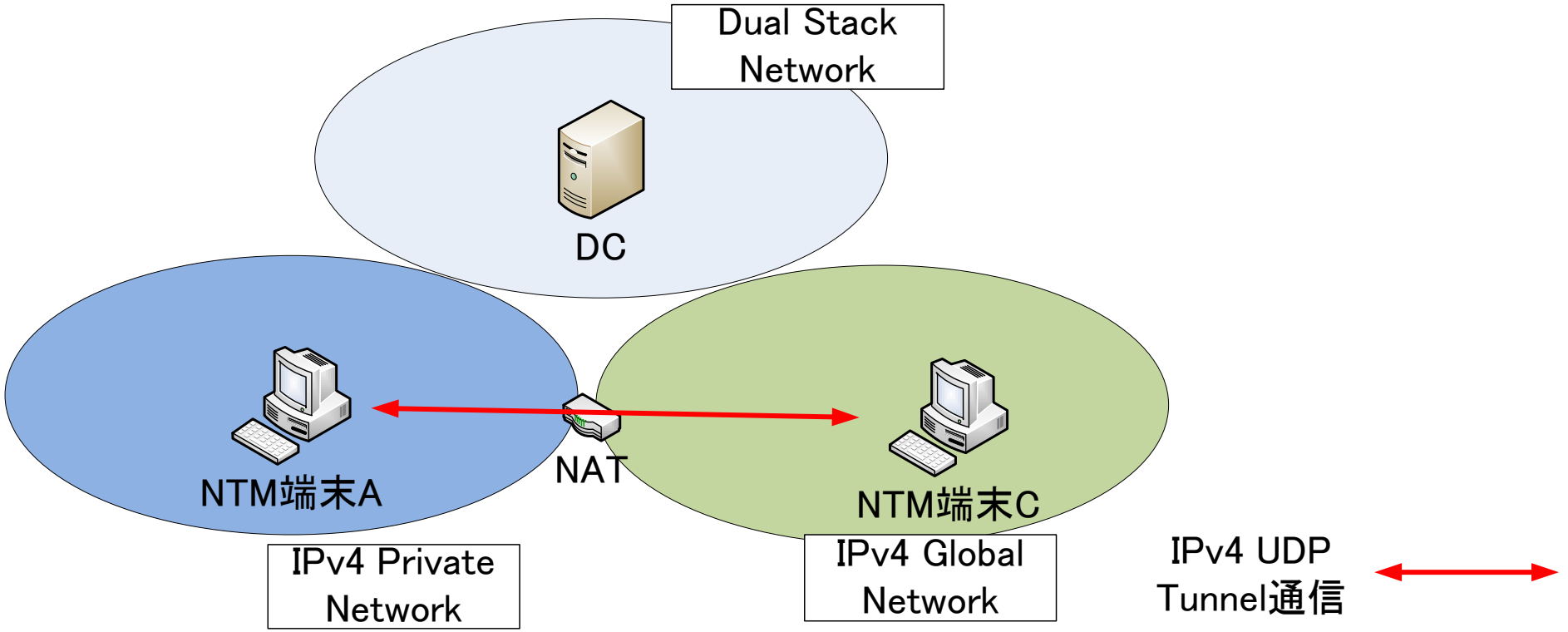
  - スマートフォンでは特に切り替えが頻繁



-NTMobile-

(Network Traversal with Mobility)

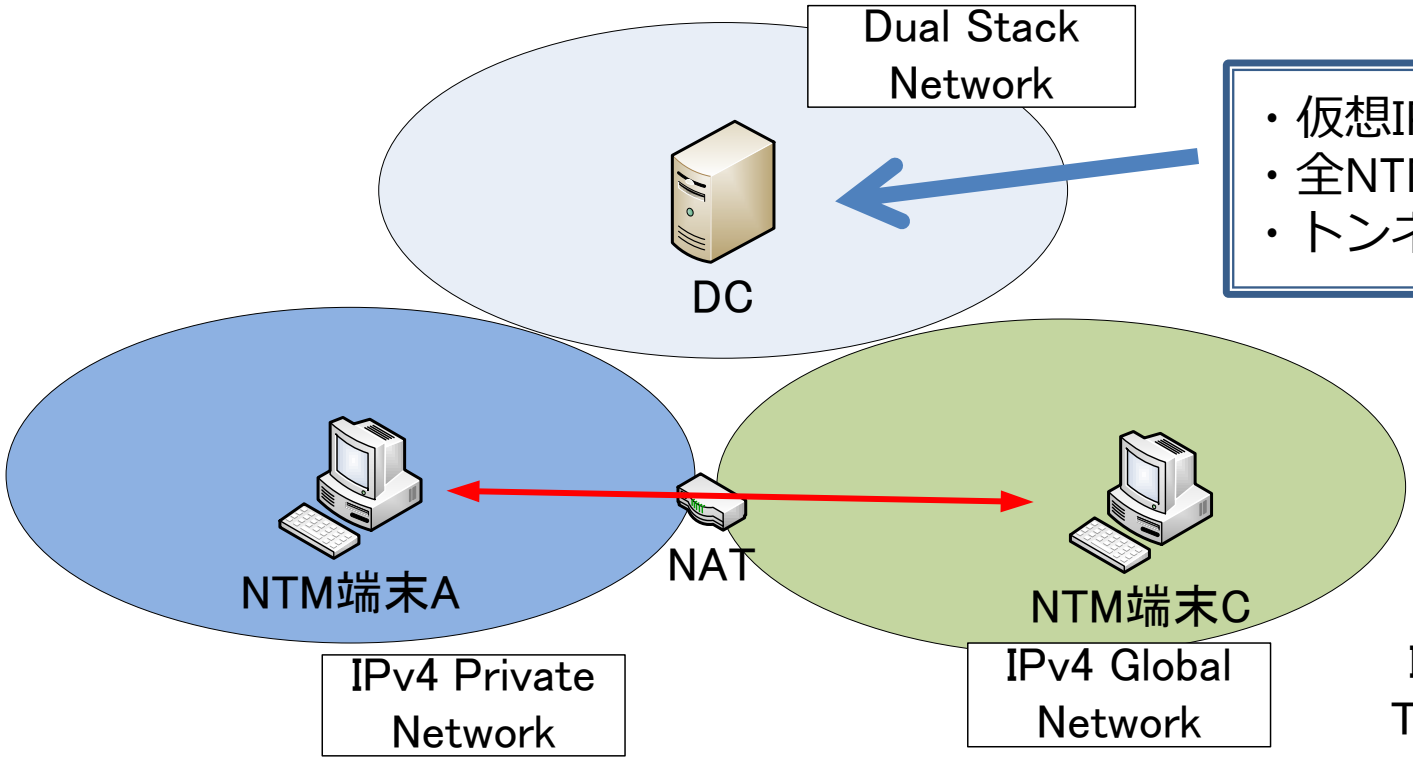
移動透過性/通信接続性を同時に実現



-NTMobile-

(Network Traversal with Mobility)

DC（Direction　Coordinator)



・仮想IPアドレスの配布
・全NTM端末の位置の把握
・トンネル構築指示

IPv4 UDP
Tunnel通信

仮想IPアドレス

- IPアドレスの役割

IPアドレス
• 位置情報 • 通信識別子

IPアドレス一つに
２つの役割がある

仮想IPアドレス

- IPアドレスの役割

IPアドレス
• 位置情報　　接続先とともに変化 • 通信識別子　通信識別子も変化

IPアドレス一つに
２つの役割がある

仮想IPアドレス

- IPアドレスの役割

IPアドレス
• 位置情報　→　**実IPアドレス** • 通信識別子　→　**仮想IPアドレス**

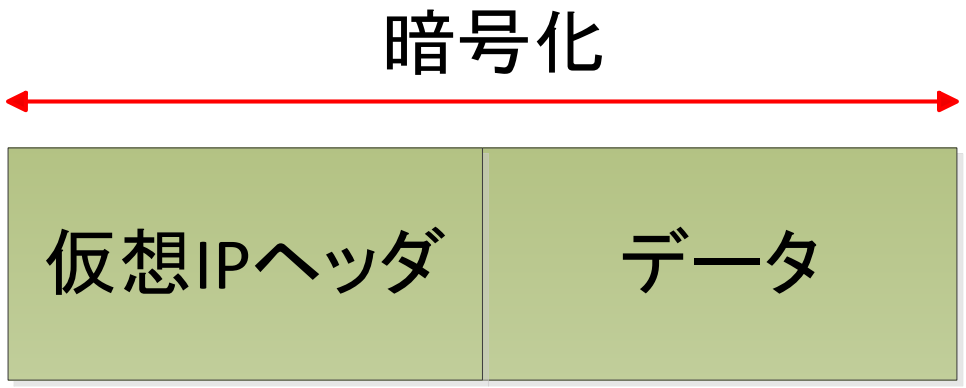
２つのIPアドレスに一つずつの役割を付与
仮想IPアドレスは不変のため通信の継続が可能

UDPトンネル通信

- DCから配布された仮想IPアドレスでパケットを生成
- 実IPアドレスで仮想IPパケットを**UDPカプセル化**
  - カプセル内部は暗号化
    - 通信路での盗聴対策
  - 実ネットワークをそのまま活用可能

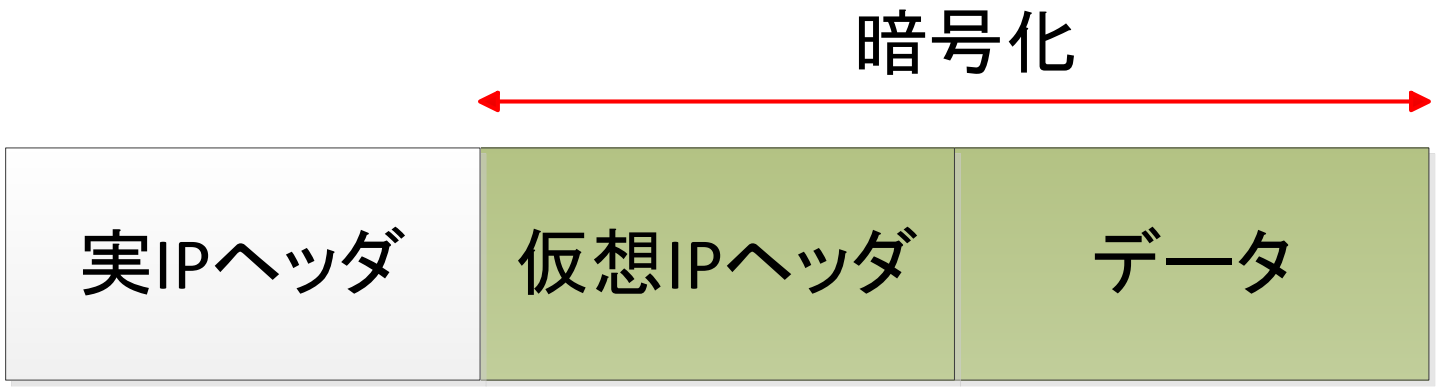
UDPトンネル通信

- DCから配布された仮想IPアドレスでパケットを生成
- 実IPアドレスで仮想IPパケットを**UDPカプセル化**
  - カプセル内部は暗号化
    - 通信路での盗聴対策
  - 実ネットワークをそのまま活用可能



UDPトンネル通信

- DCから配布された仮想IPアドレスでパケットを生成
- 実IPアドレスで仮想IPパケットを**UDPカプセル化**
  - カプセル内部は暗号化
    - 通信路での盗聴対策
  - 実ネットワークをそのまま活用可能



フレームワーク組込型NTMobile

移動透過性の実現に
カーネル実装が主流

- モバイル端末はカーネル空間が**ブラックボックス**なOSが多い
- カーネルに合わせた**カスタマイズ**が必要

全てのモジュールを
アプリケーション層
で実装

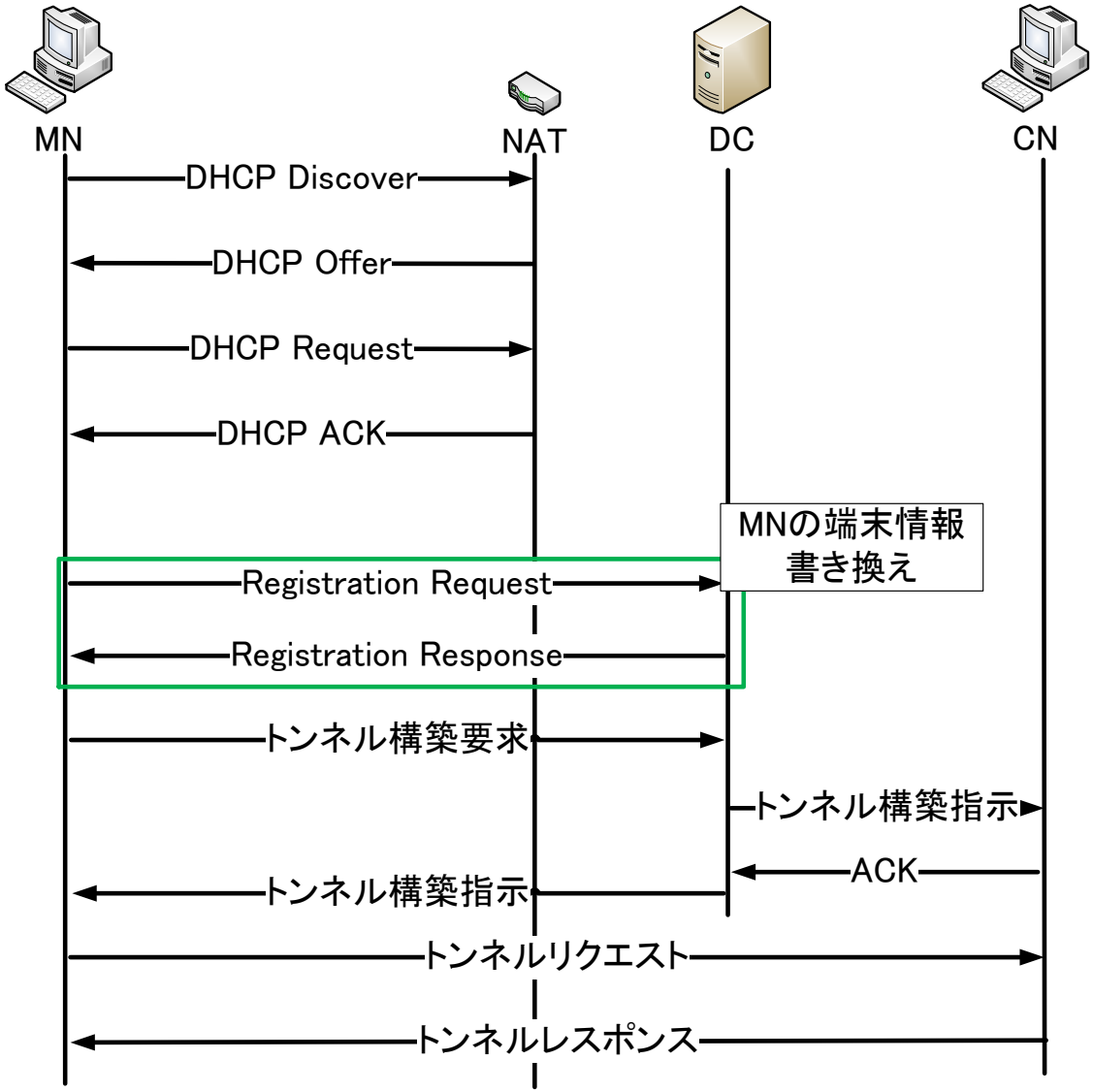
- OSに**依存しない**
- **アップデート不要**

フレームワーク組込型NTMobile

実装済機能	未実装機能
• パケット暗号化/カプセル化 • サーバ群との通信	• 移動検出 • 移動時の排他処理

- **移動透過性**に関わる重要なモジュールが未実装

移動時のシーケンス


MN
NAT
DC
CN
DHCP Discover
DHCP Offer
DHCP Request
DHCP ACK
MNの端末情報
書き換え
Registration Request
Registration Response
トンネル構築要求
トンネル構築指示
ACK
トンネル構築指示
トンネルリクエスト
トンネルレスポンス

実装

- **移動検出機能**（Moving Detector）を実装
  - 10秒に1回自身のアドレスを確認
  - 変化検出後はトンネル再構築処理の実行
- トンネルテーブル書き換え
  - **排他処理**

排他処理

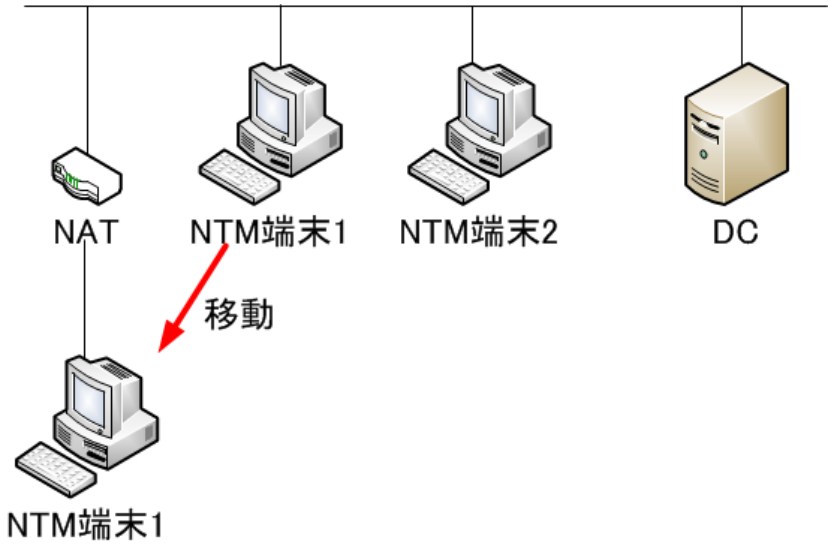
- トンネルテーブル
  - 実IPアドレス、仮想IPアドレス等の関係を記述したテーブル
  - 移動後にトンネルテーブルの書き換えが必要
    - 書き換えにともなう**排他処理**

Keys	トンネルデータ
1	トンネルデータ1
2	トンネルデータ2
3	トンネルデータ3

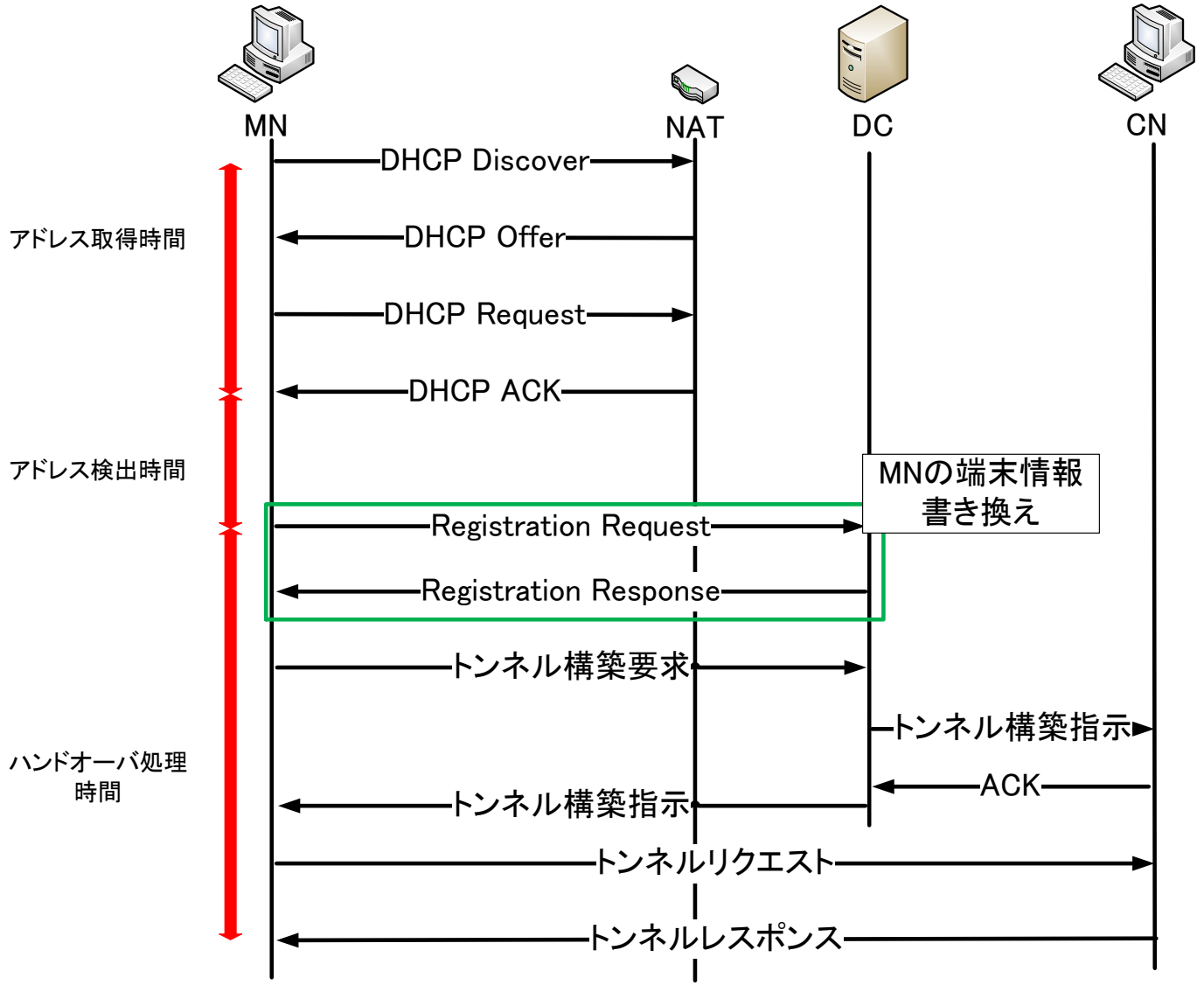
トンネルデータ1	
実IPアドレス	←書き換え
仮想IPアドレス	
（省略）	
Mutex	←ロック/アンロック

性能評価

- NTM端末１をDCと同じネットワークからNAT配下へ移動、その際のトンネル再構築時間を計測
- 全体の処理時間を三つのフェーズに分割しそれぞれ１０回計測した平均、最小値、最大値を計測する。



性能評価


MN
NAT
DC
CN
DHCP Discover
DHCP Offer
DHCP Request
DHCP ACK
アドレス取得時間
アドレス検出時間
MNの端末情報
書き換え
Registration Request
Registration Response
トンネル構築要求
トンネル構築指示
ACK
トンネル構築指示
ハンドオーバ処理
時間
トンネルリクエスト
トンネルレスポンス

実験結果

- アドレス変化取得時間が全体の**約９８．７%**かかっている
  - **１０秒に1回**自身のアドレスを読み取るタイマーを使用

	AVE[s]	MAX[s]	MIN[s]
全体処理時間	7.002	9.274	1.510
アドレス取得時間	0.007	0.008	0.006
アドレス変化検出時間	**6.911**	**9.201**	**1.477**
ハンドオーバ処理時間	0.084	0.130	0.026

まとめ

- フレームワークの実装
  - 各モジュールの機能
  - ハンドオーバ処理
- 今後の予定
  - アドレス変化検出時間の短縮

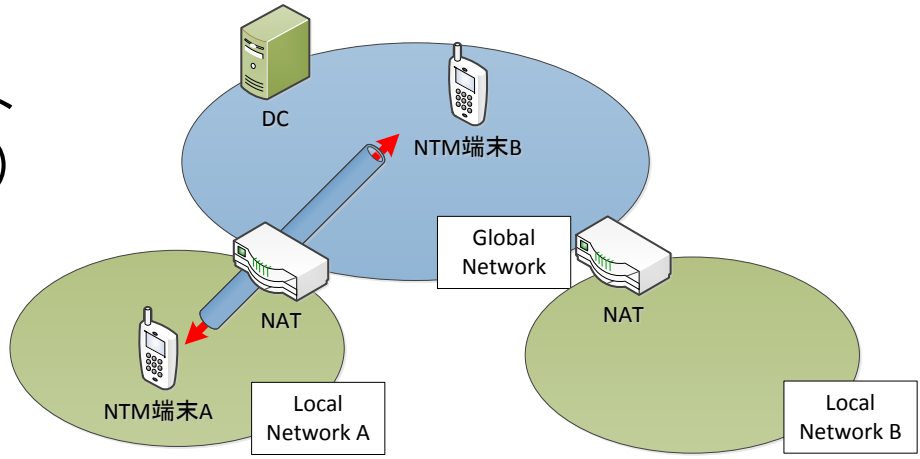
付録

- NTMobile
- フレームワークモジュール図
- PathID

-NTMobile-

(Network Traversal with Mobility)

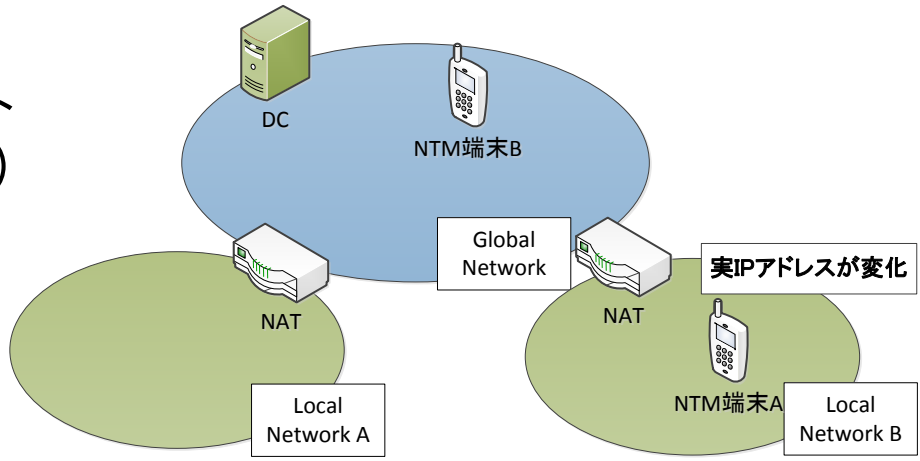
- 移動透過性と通信接続性を実現
  - 仮想ＩＰアドレスの使用
  - 実ＩＰアドレスでUDPカプセル化**(移動透過性)**
    - **カプセル内部は暗号化**
  - DCが通信端末間のトンネル構築を指示**（通信接続性）**
- IPアドレス変化時（移動検出時）
  - DC内にあるNTM端末情報によりトンネル再構築（ハンドオーバ処理）



-NTMobile-

(Network Traversal with Mobility)

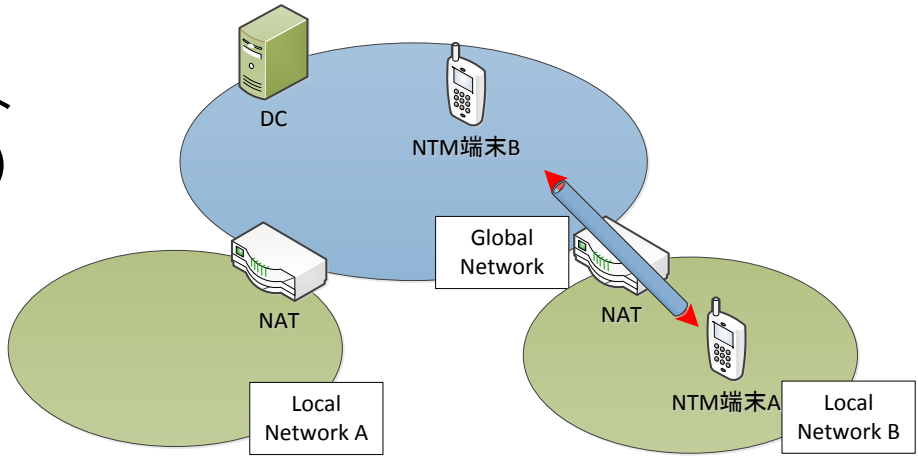
- 移動透過性と通信接続性を実現
  - 仮想ＩＰアドレスの使用
  - 実ＩＰアドレスでUDPカプセル化**(移動透過性)**
    - **カプセル内部は暗号化**
  - DCが通信端末間のトンネル構築を指示**（通信接続性）**
- IPアドレス変化時（移動検出時）
  - DC内にあるNTM端末情報によりトンネル再構築（ハンドオーバ処理）



-NTMobile-

(Network Traversal with Mobility)

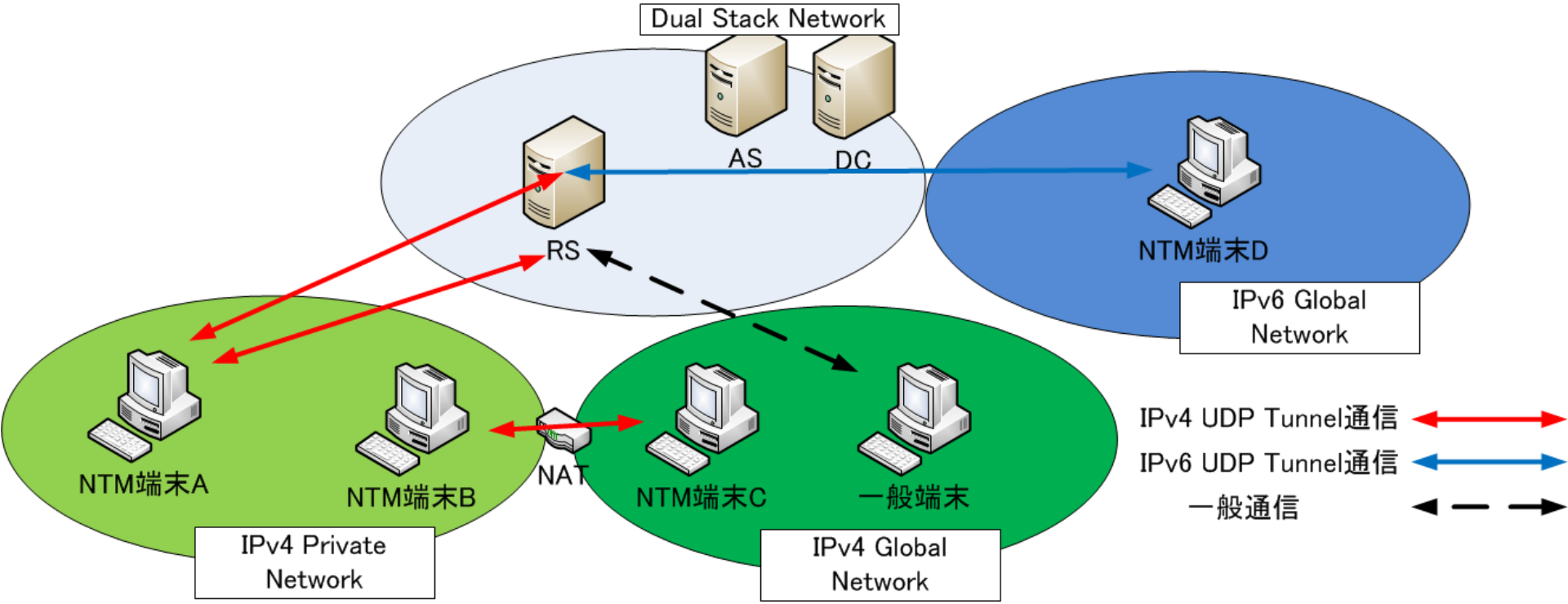
- 移動透過性と通信接続性を実現
  - 仮想ＩＰアドレスの使用
  - 実ＩＰアドレスでUDPカプセル化**(移動透過性)**
    - **カプセル内部は暗号化**
  - DCが通信端末間のトンネル構築を指示**（通信接続性）**
- IPアドレス変化時（移動検出時）
  - DC内にあるNTM端末情報によりトンネル再構築（ハンドオーバ処理）



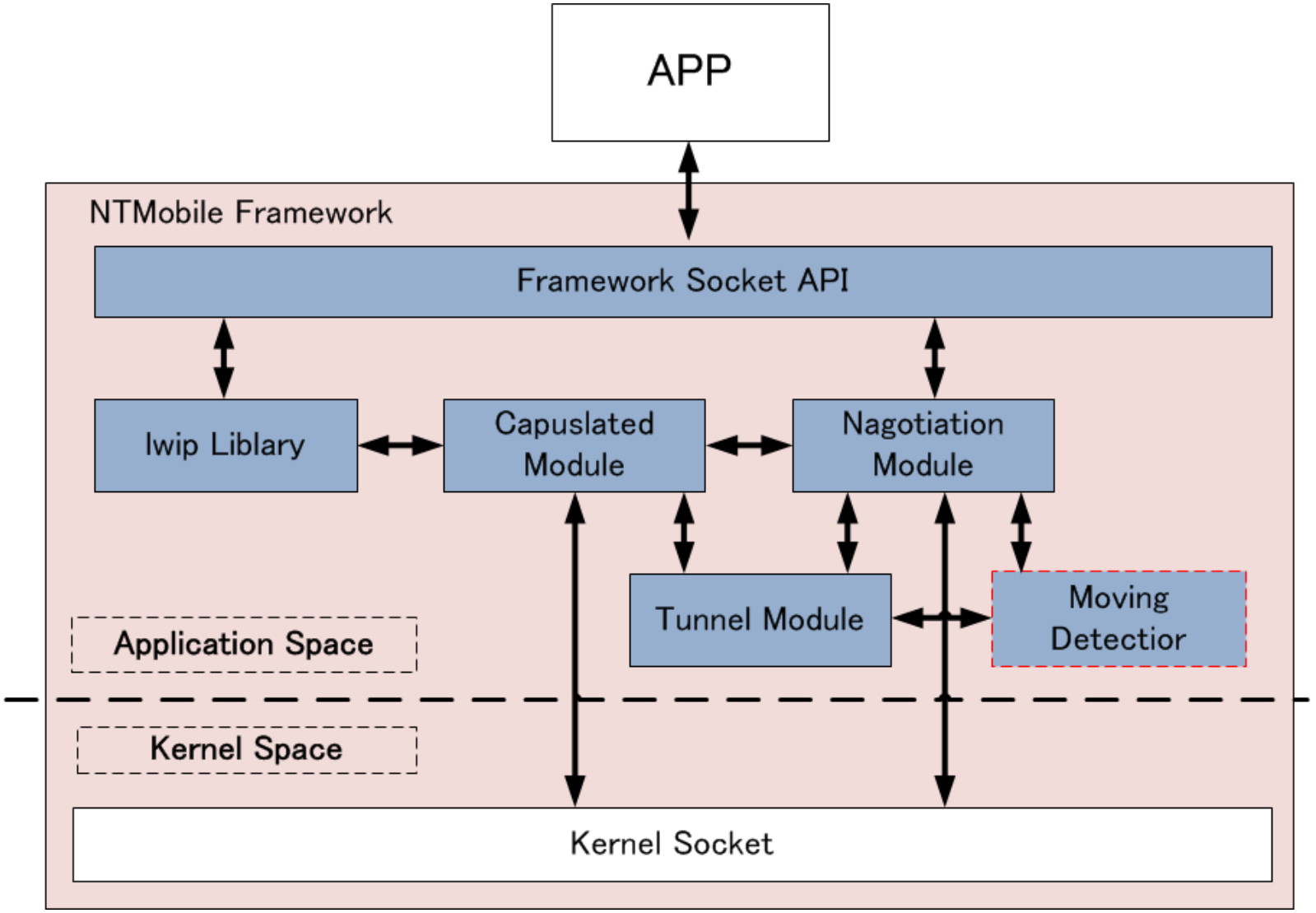
-NTMobile-

(Network Traversal with Mobility)

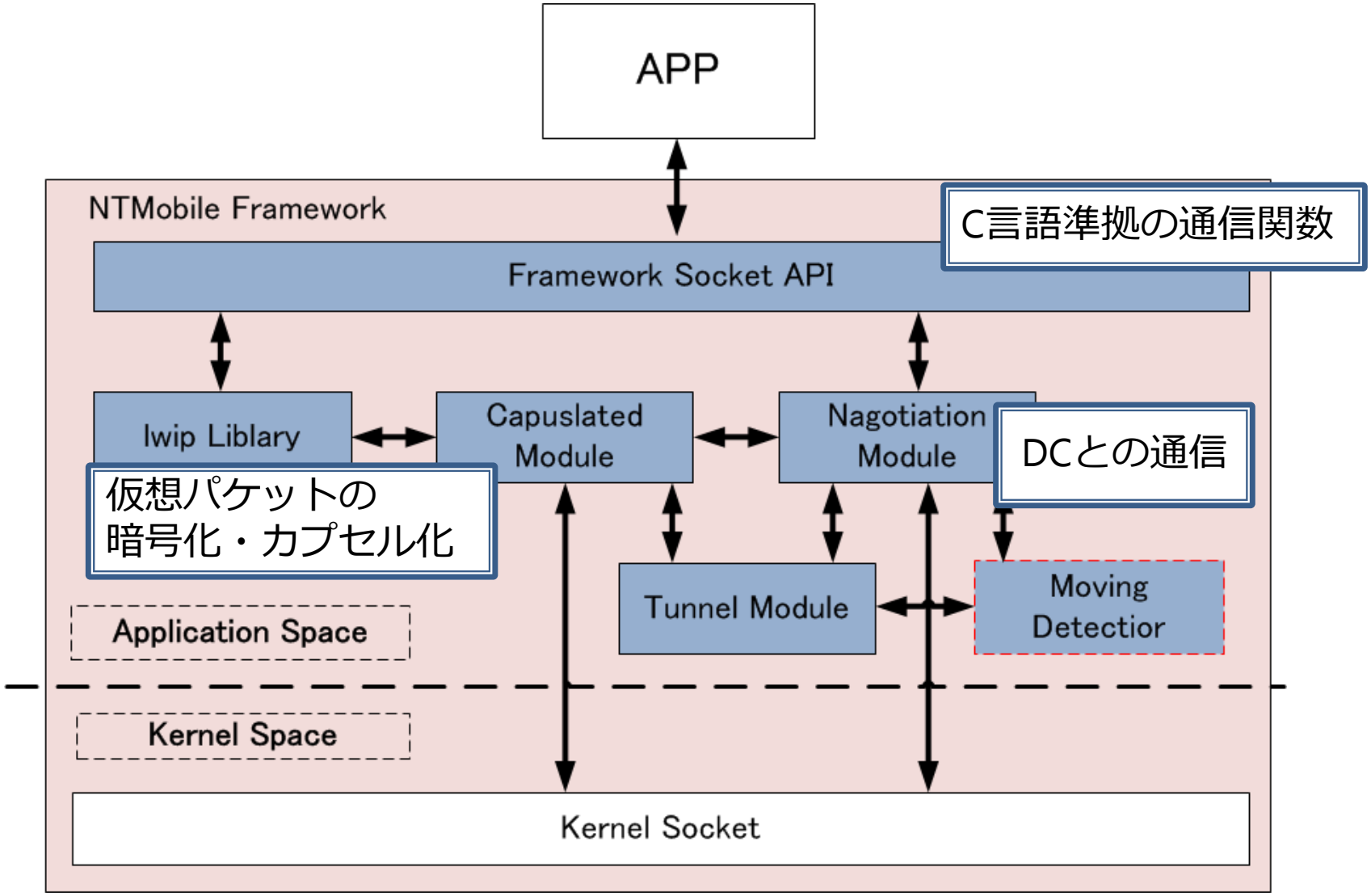
移動透過性/通信接続性を同時に実現



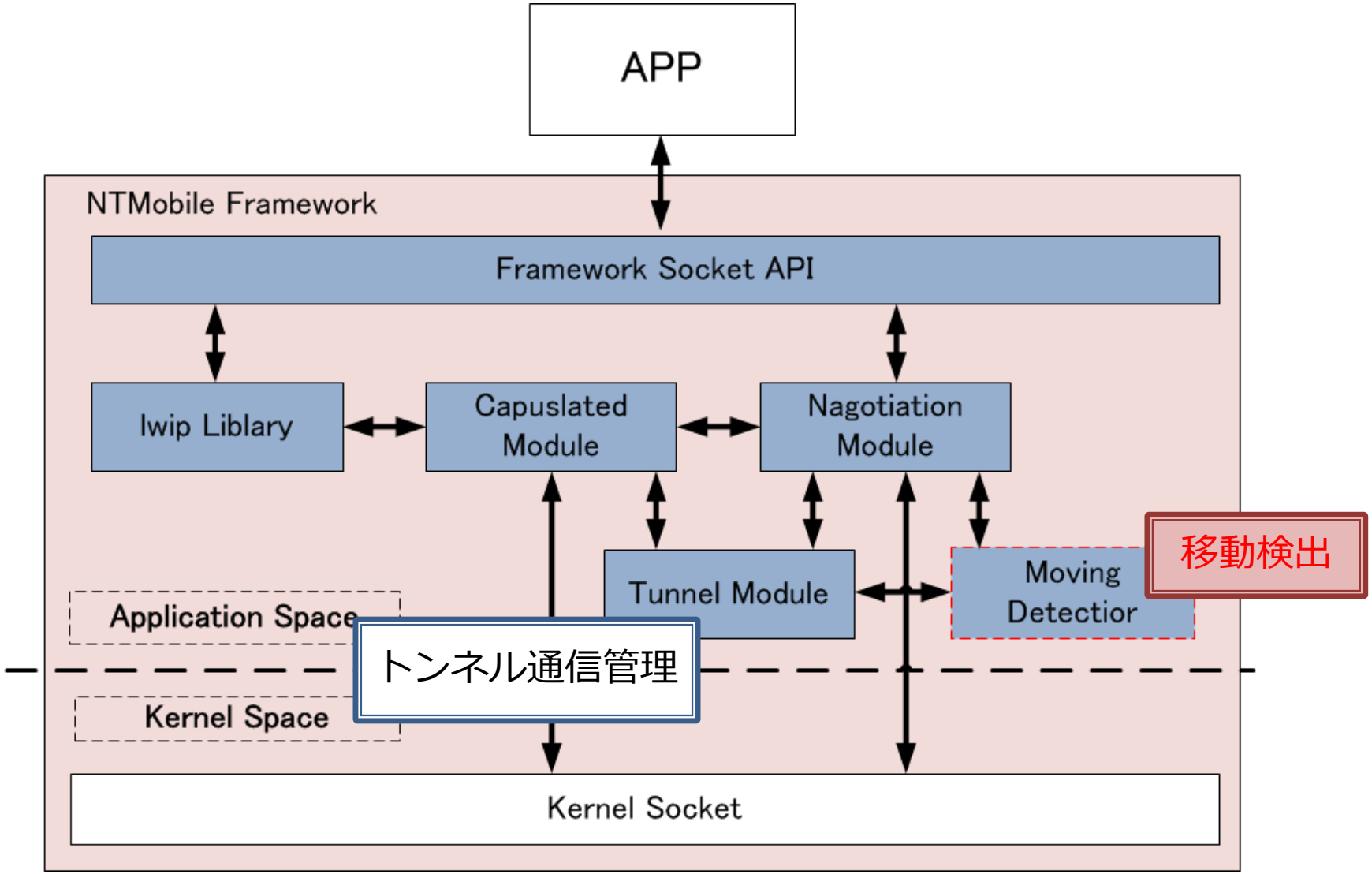
フレームワーク組込型NTMobile



フレームワーク組込型NTMobile



フレームワーク組込型NTMobile



PathID

- 仮想IPアドレスと対になる**通信識別子**

✓端末情報登録時DCが生成→エンド端末へ

✓ネットワーク切り替え毎に変化

✓通信継続不可

✓エンド端末で生成しDCへ送信→各端末へ

✓同一のPath IDを**継続して使用**

✓移動透過性の実現