

NTMobile を Android 端末に実装するための検討

赤堀 蒼磨 †*, 納堂 博史 †, 鈴木 秀和 †, 内藤 克浩 ‡, 渡邊 晃 †(†名城大学, ‡愛知工業大学)

Studies of NTMobile Implementation in Android.

Soma Akahori†*, Hiroshi Nodo†, Hidekazu Suzuki†, Katsuhiko Naito‡, Akira Watanabe†

(†Meijo University, ‡Aichi Institute of Technology)

1 はじめに

Android や iOS 等の OS を搭載したスマートフォンやタブレット端末の普及により, 手軽にインターネットに接続が可能になった. これらの移動端末において, 接続するネットワーク環境に関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必要性が高まっている. 著者らは, 両者を同時に実現可能とする次世代技術 NTMobile(Network Traversal with Mobility)[1] を提案している. この技術を Android や iOS 端末へ実装できると有用である. 本稿では, NTMobile を Android 端末上のアプリケーションとして動作させる, NTMobile フレームワークの実現方法について報告する.

2 NTMobile の概要

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 (以降 NTM 端末と記す) は, ログイン時に DC から重複しない仮想 IP アドレスの配布を受ける. アプリケーションは仮想 IP アドレスで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, 通信開始時または移動を検出した時, 両 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. 以上の動作により, 通信接続性と移動透過性を同時に実現する.

NTMobile は, 当初 Linux カーネル上での動作を検証していた. しかし Android 端末ではルート権限が必要となることから, NTMobile 普及のために望ましくない. そこで NTMobile をアプリケーションに移植し, フレームワークとして動作できるよう検討を進めてきた. これまで簡易実装により UDP レベルでの動作を確認している. 今回は TCP も実現可能なように, フレームワークの実装を見直した.

3 フレームワークの実装方法

NTMobile フレームワークは, C 言語で記述されており, 上位アプリケーションに対して NTMobile 用のソケットインタフェースを提供する. フレームワーク内で生成した TCP/IP パケットをデータとみなし, Linux が提供する UDP によりカプセル化を行う. UDP ポート番号 4330 を使うと, カーネル版との互換性を保つことができる. ただし, 4330 に固定するとフレームワークを利用する他アプリケーションとの同時動作ができないので, ポート番号をダイナミックに変更できるようになっている.

NTMobile フレームワークのモジュール構成は Fig.1 のとおりである. 仮想 TCP/IP は, TCP/IP をアプリケーション層で実現するモジュールで, オープンソースの lwIP(light weight IP)[2] を利用する. libntm は, DC と NTM 端末が共通で使用するライ

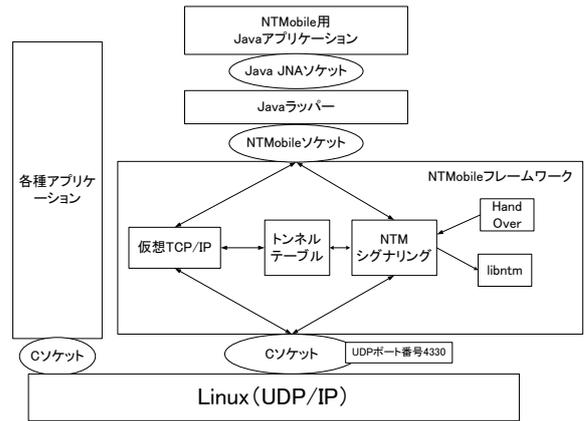


Fig. 1 Module of NTMobile Framework

ブラリである. NTM シグナリングは, 通信開始時または IP アドレス変化時に DC の指示を受けて, UDP トンネルの構築を行うモジュールである. トンネルテーブルはトンネル構築時にカプセル化処理に必要なテーブルである. Handover は IP アドレスの変化を常時監視するモジュールで, 移動透過性実現のためにフレームワーク特有の処理を行う. カーネル版では IP アドレスの変化をカーネルで検出し, トンネル構築処理を開始していた. フレームワーク版では, IP アドレスの変化をカーネルから教えてもらうインタフェースがないため, Handover モジュールにて IP アドレスの変化を定期的にチェックする. NTMobile 接続時または, IP アドレスの変化が検出されたら NTM シグナリングを呼び出し DC に経路指示を依頼する. NTMobile ソケットは, カーネルが提供する C ソケットと 1 対 1 に対応しており, 機能的に互換性を持つインタフェースである. Java ラッパーは JNA(Java Native Access) で定義された Java インタフェースを NTMobile ソケットに変換するモジュールである. NTMobile に係る動作はフレームワークですべて実行するため, Java ラッパーは単に Java と C インタフェースの変換を行うだけである.

4 まとめ

NTMobile フレームワークの実装方法について報告した. 今後はハンドオーバー部分やラッパーを中心に, 本稿のコンテンツに基づいて Android 端末へ実装する. 実装が完了次第, 動作確認及び性能評価を行う予定である.

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] lwIP - A Lightweight TCP/IP stack - Summary <<http://savannah.nongnu.org/projects/lwip/>> (accessed 2016-6-27)

NTMobile をAndroid 端末に実装するための検討

赤堀 蒼磨[†], 納堂 博史[†], 鈴木 秀和[†], 内藤 克浩[‡], 渡邊 晃[†]
[†]名城大学 理工学部
[‡]愛知工業大学 情報科学部

研究背景

- モバイル端末が急速に普及 (Android, iOS 端末等)
 - 移動端末で手軽にインターネット接続が可能
- WI-FI, LTE, 3G などの接続先が変化
 - IP アドレスの変化
 - 通信切断
- グローバルアドレスからプライベートアドレスに対して通信開始が不可 (NAT 越え問題)



通信接続性・移動透過性
の必要性

-NTMobile-

(Network Traversal with Mobility)

■ 移動透過性と通信接続性を実現

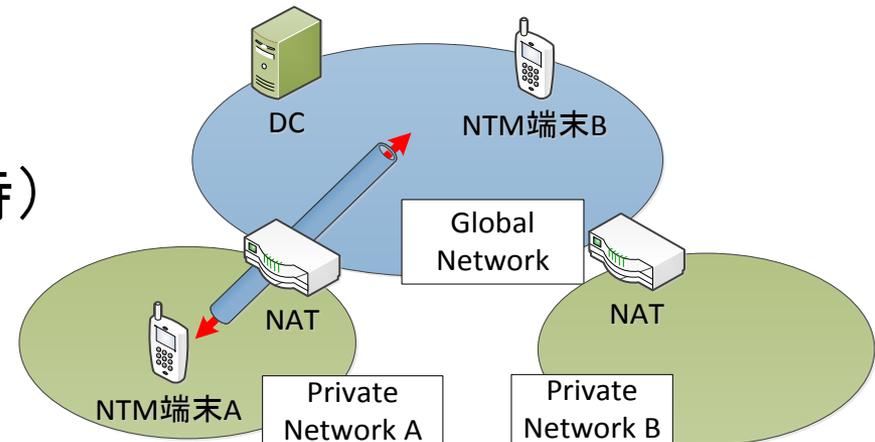
- DC(Direction Coordinator)
 - 仮想IPアドレスの配布
 - 通信セッションの確立
- 全通信を実IPアドレスでUDPカプセル化(移動透過性)
- DCが通信端末間のトンネル構築を指示(通信接続性)

| | | | |
|--------------------|--------|---------------------|--|
| IPヘッダ (実IPアドレス) | UDPヘッダ | IPヘッダ (仮想IPアドレス) | |
|--------------------|--------|---------------------|--|

NTMobileパケットの構成

■ 実IPアドレス変化時(移動検出時)

- 仮想IPアドレスは不変
 - 通信の継続が可能



-NTMobile-

(Network Traversal with Mobility)

■ 移動透過性と通信接続性を実現

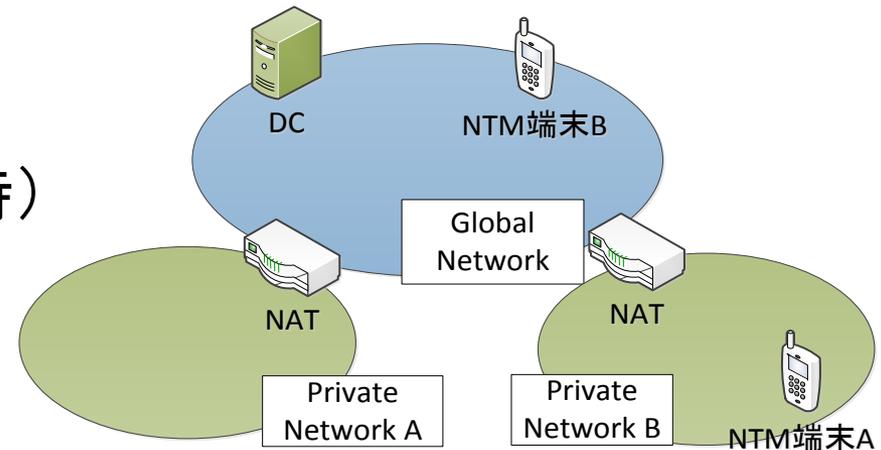
- DC(Direction Coordinator)
 - 仮想IPアドレスの配布
 - 通信セッションの確立
- 全通信を実IPアドレスでUDPカプセル化(移動透過性)
- DCが通信端末間のトンネル構築を指示(通信接続性)

| | | | |
|--------------------|--------|---------------------|--|
| IPヘッダ (実IPアドレス) | UDPヘッダ | IPヘッダ (仮想IPアドレス) | |
|--------------------|--------|---------------------|--|

NTMobileパケットの構成

■ 実IPアドレス変化時(移動検出時)

- 仮想IPアドレスは不変
 - 通信の継続が可能



-NTMobile-

(Network Traversal with Mobility)

■ 移動透過性と通信接続性を実現

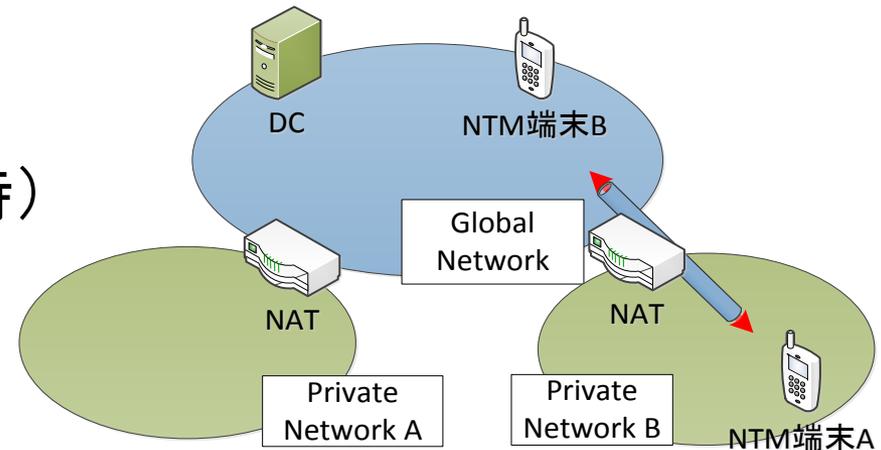
- DC(Direction Coordinator)
 - 仮想IPアドレスの配布
 - 通信セッションの確立
- 全通信を実IPアドレスでUDPカプセル化(移動透過性)
- DCが通信端末間のトンネル構築を指示(通信接続性)

| | | | |
|--------------------|--------|---------------------|--|
| IPヘッダ (実IPアドレス) | UDPヘッダ | IPヘッダ (仮想IPアドレス) | |
|--------------------|--------|---------------------|--|

NTMobileパケットの構成

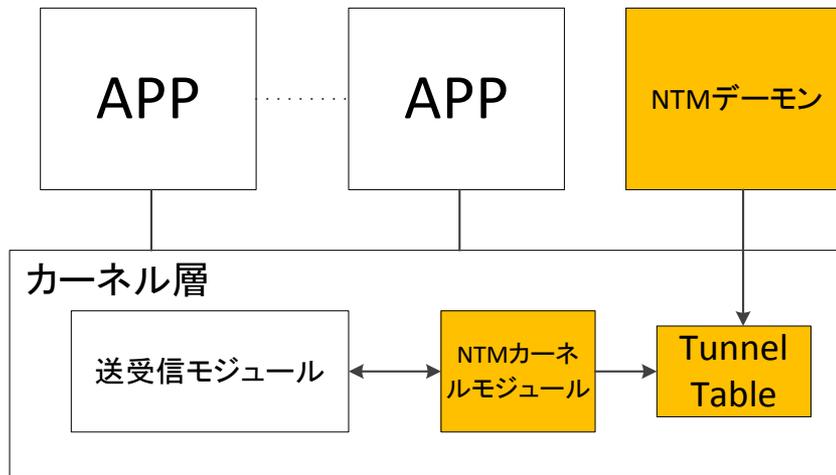
■ 実IPアドレス変化時(移動検出時)

- 仮想IPアドレスは不変
 - 通信の継続が可能

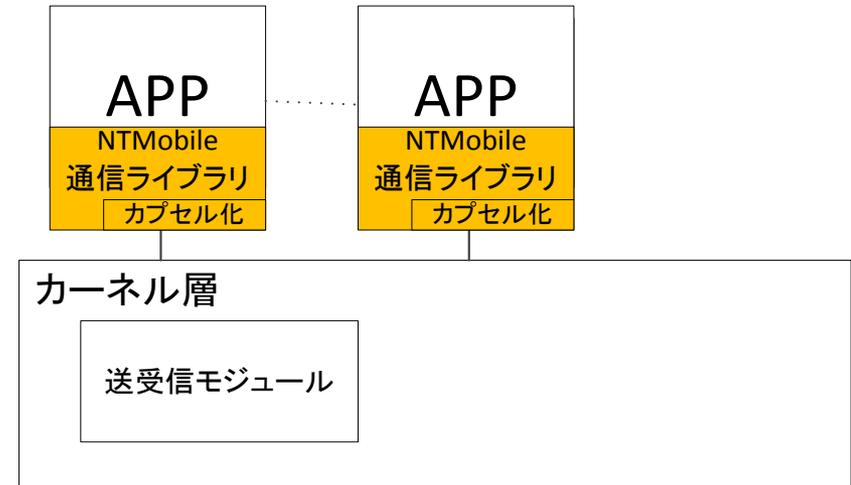


NTMobileフレームワークの必要性

カーネル版NTMobile



フレームワーク版NTMobile

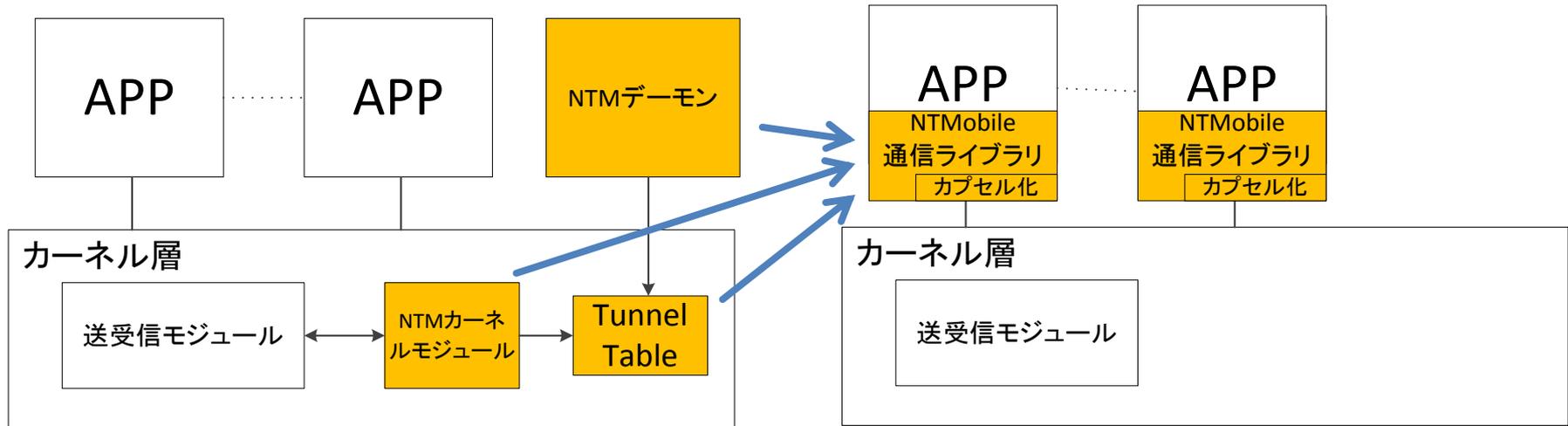


| | 通信ライブラリ | アップデート | ソフトウェア | 対応可能OS |
|----------|-----------|--------|--------|-----------------------|
| カーネル版 | Linuxカーネル | 必要あり | 変更必要なし | Linux Android(一部) |
| フレームワーク版 | アプリケーション層 | 必要なし | 変更必要あり | Linux, Android iOS |

NTMobileフレームワークの必要性

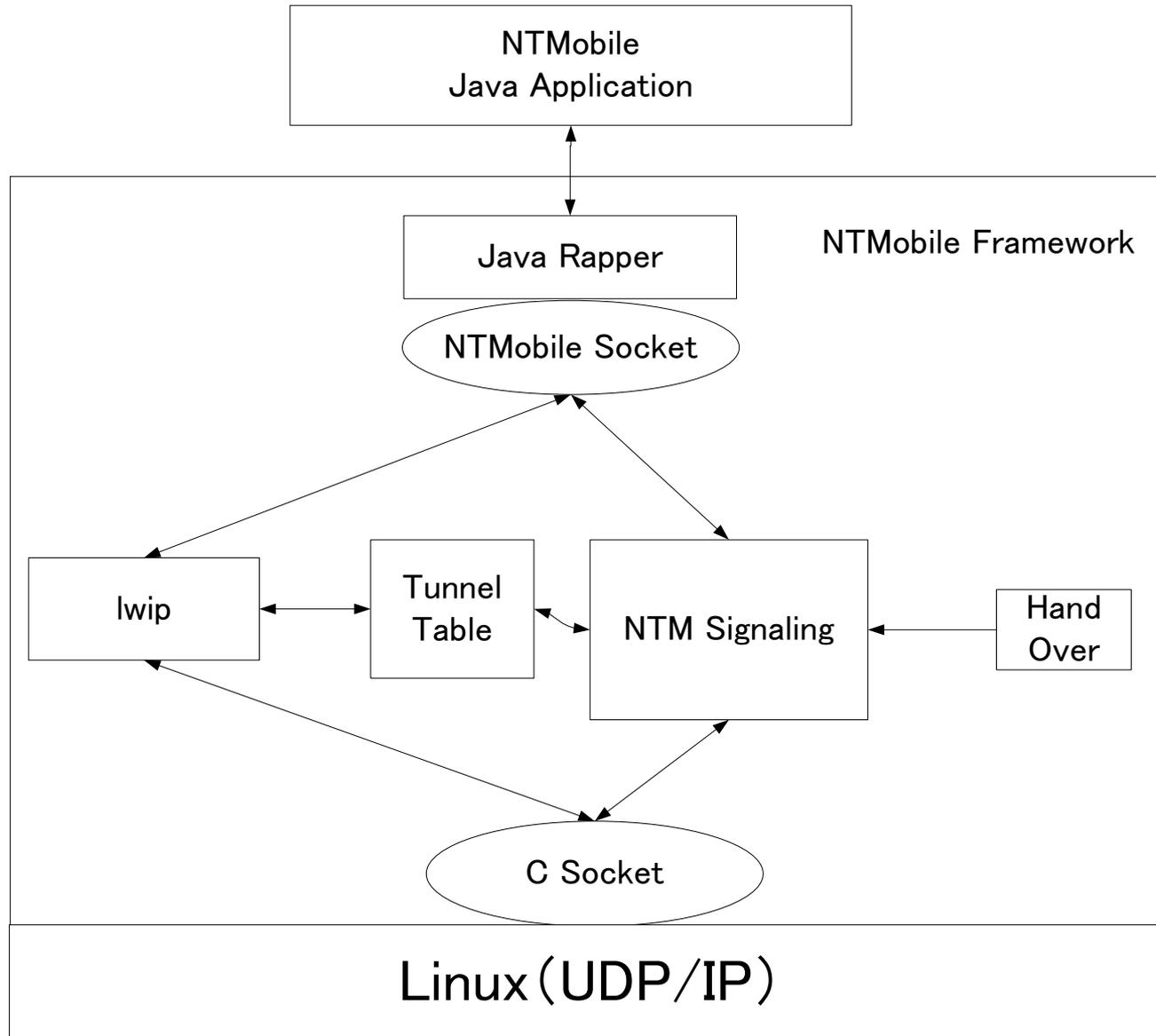
カーネル版NTMobile

フレームワーク版NTMobile



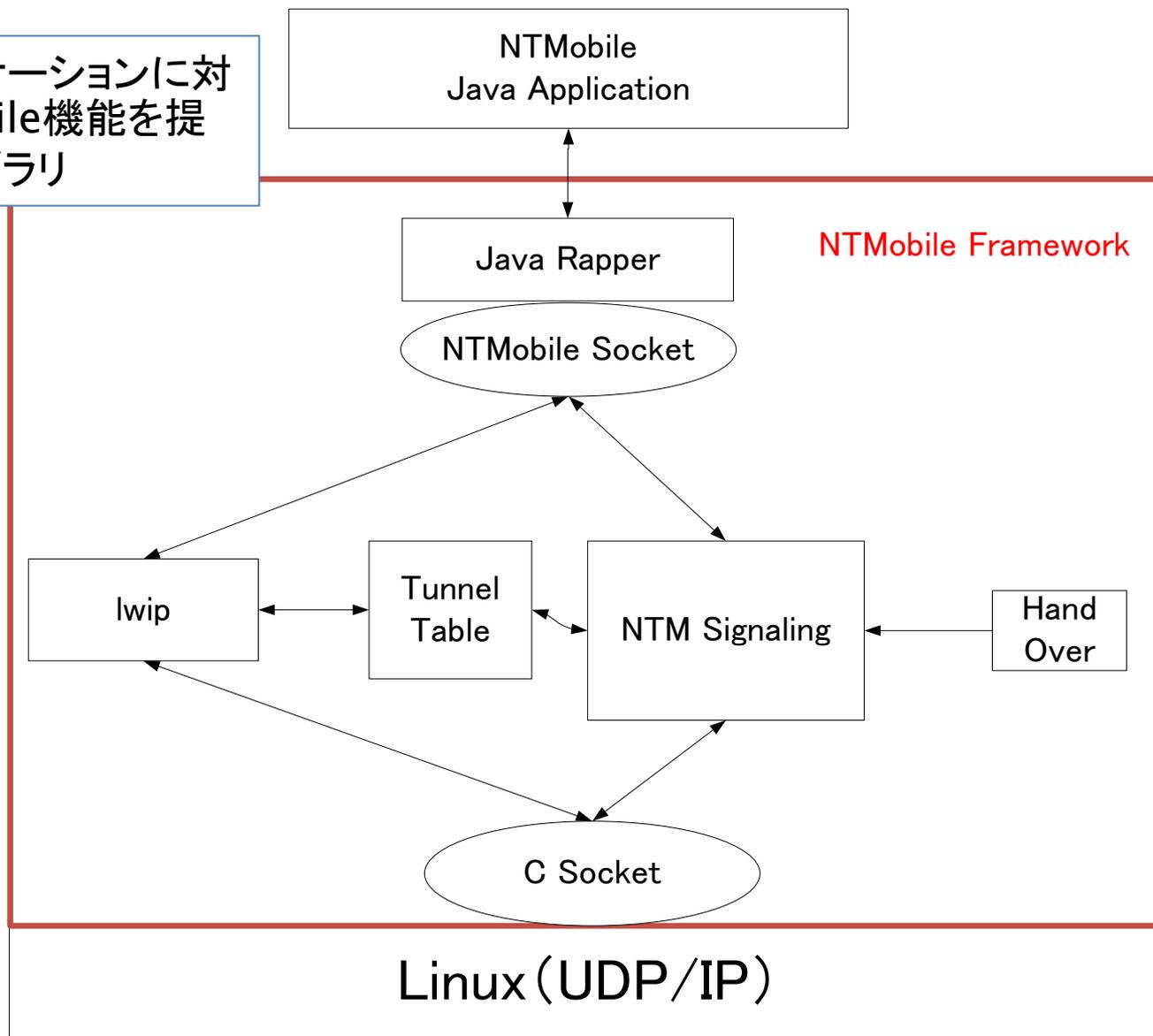
| | 通信ライブラリ | アップデート | ソフトウェア | 対応可能OS |
|----------|-----------|--------|--------|-----------------------|
| カーネル版 | Linuxカーネル | 必要あり | 変更必要なし | Linux Android(一部) |
| フレームワーク版 | アプリケーション層 | 必要なし | 変更必要あり | Linux, Android iOS |

フレームワークの構成

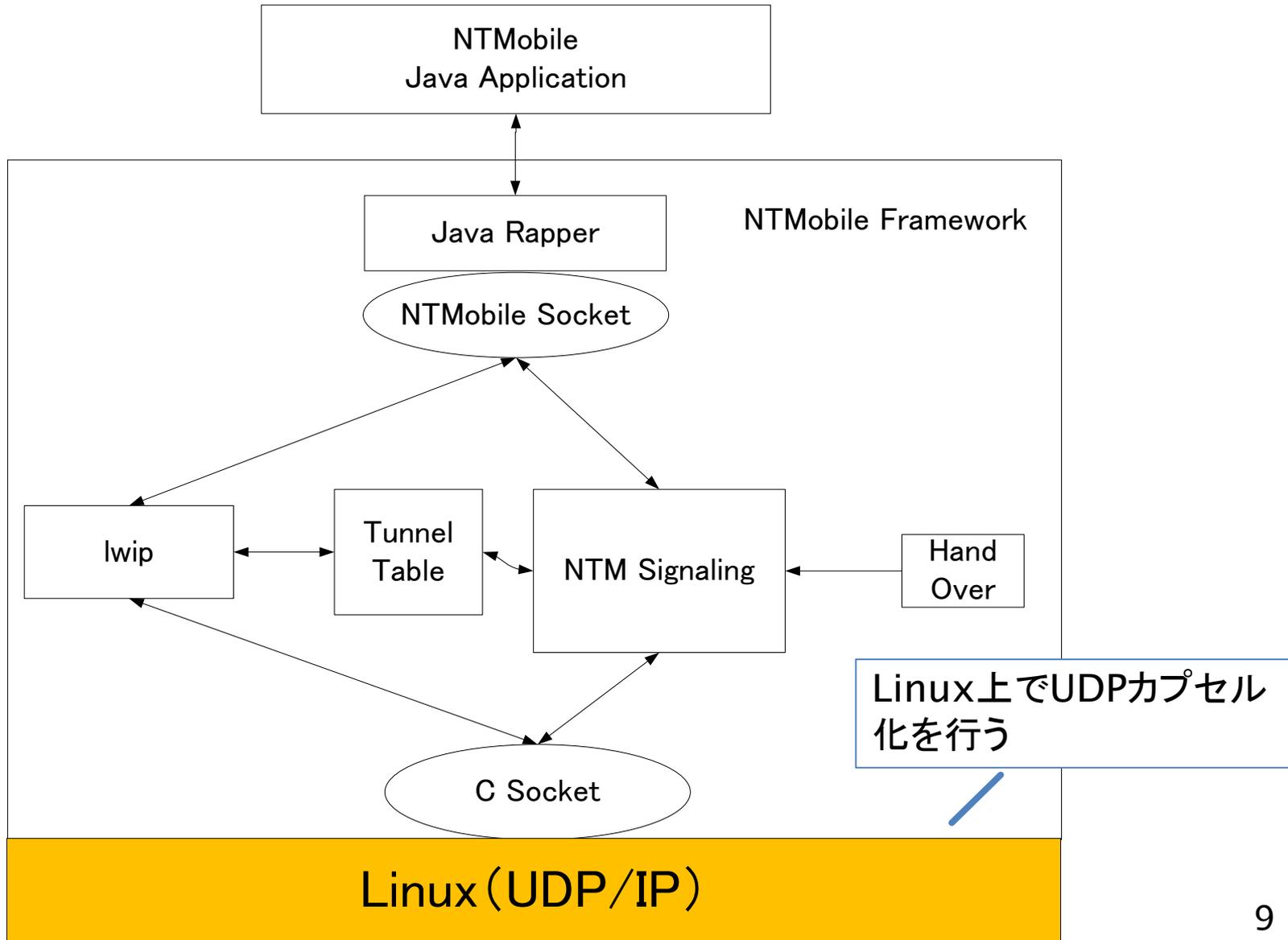


フレームワークの構成

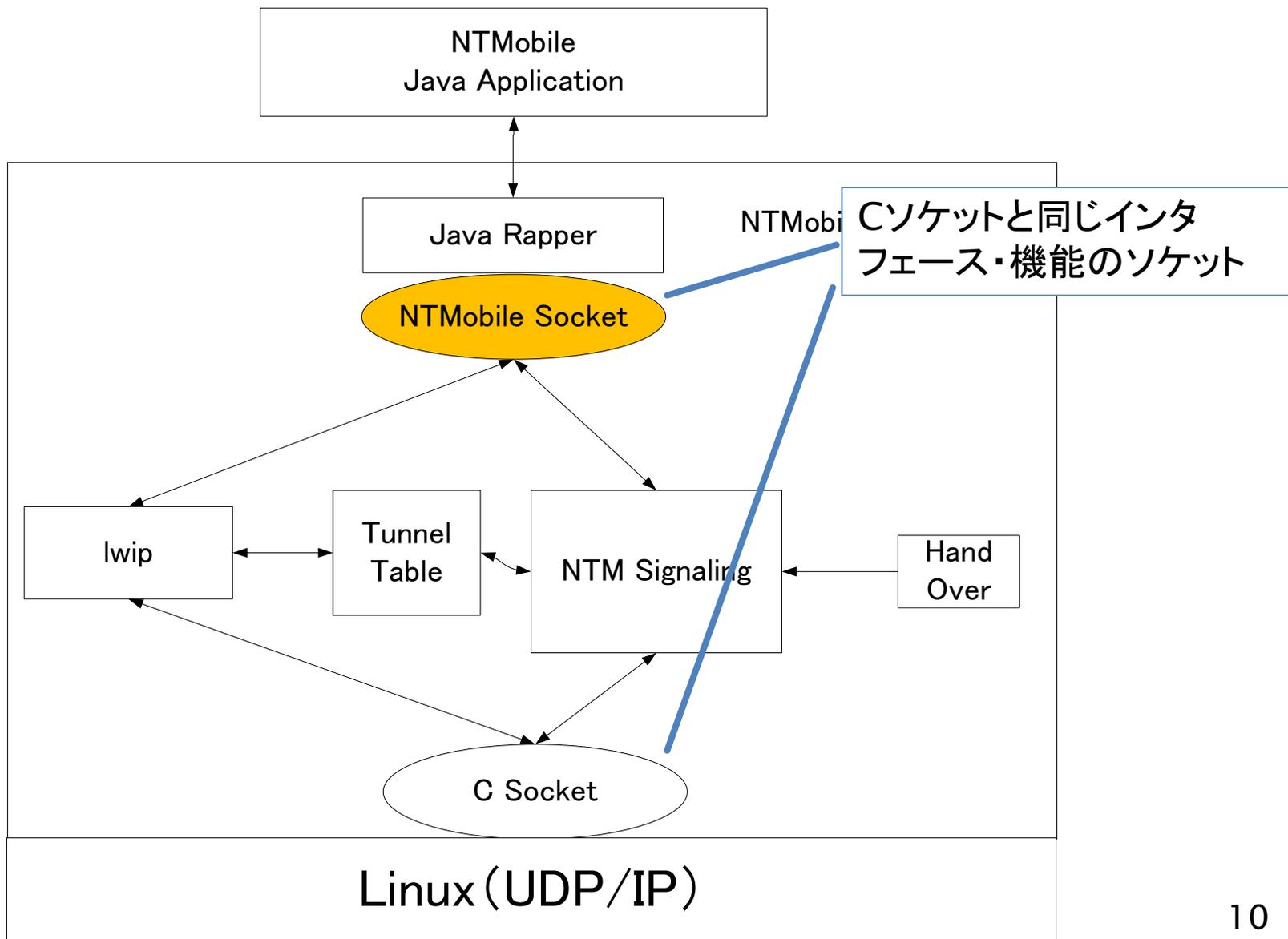
上位アプリケーションに対してNTMobile機能を提供するライブラリ



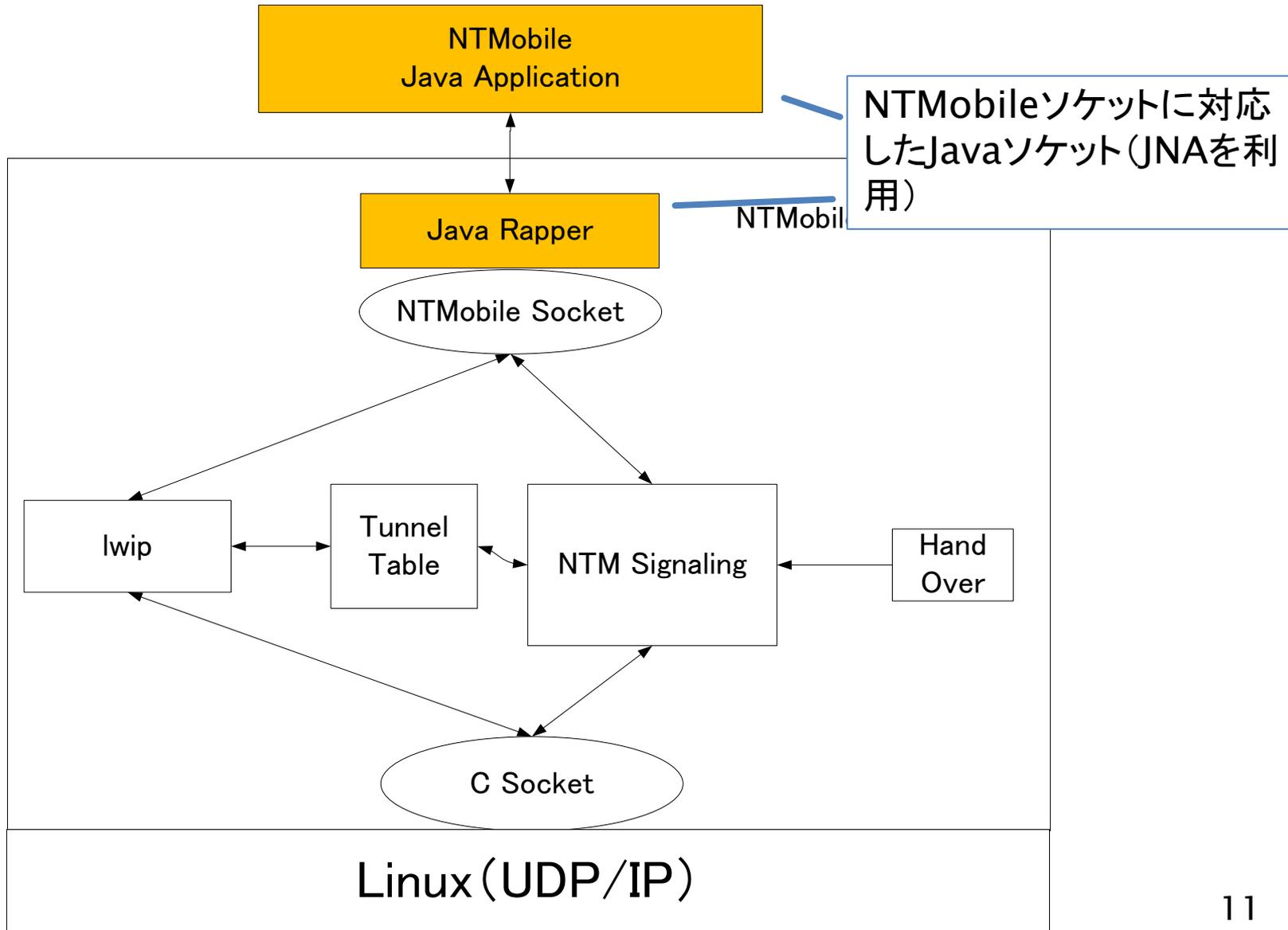
フレームワークの構成



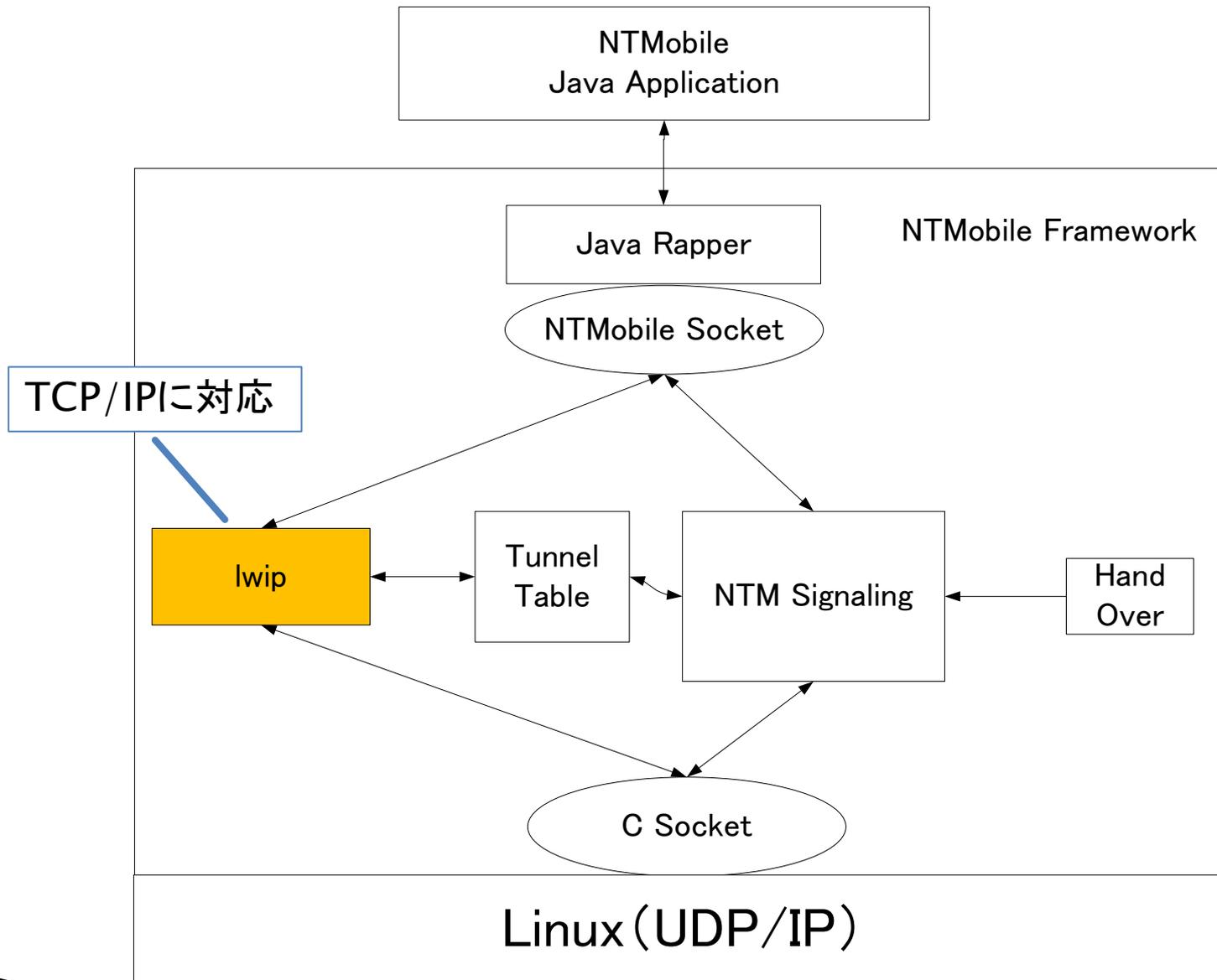
フレームワークの構成



フレームワークの構成



フレームワークの構成



lwIP(lightweight IP)

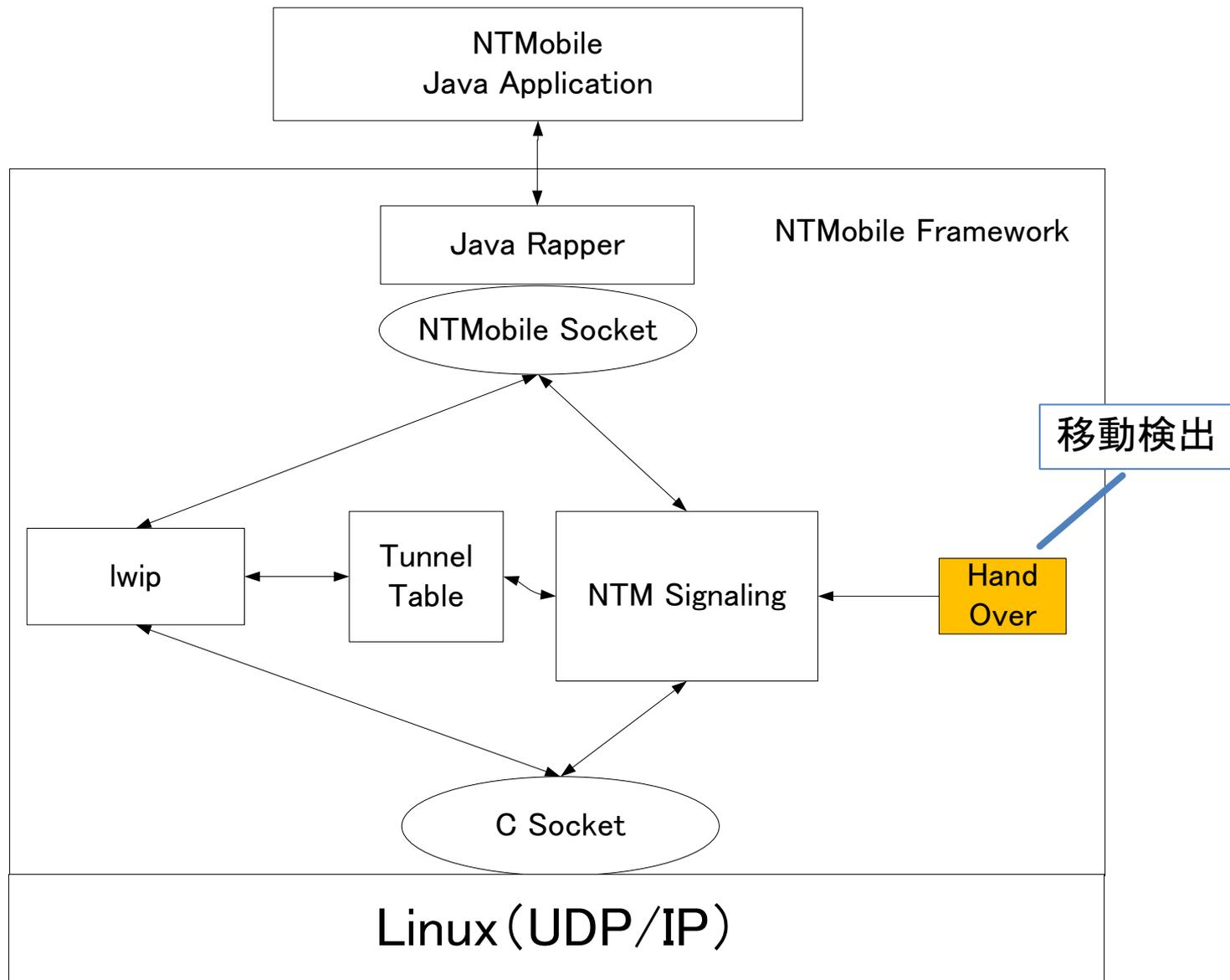
- アプリケーションでTCP/IPを実現するオープンソースソフトウェア
- 主に組み込み系システムで使用
- 本システムではカプセル内部のTCP/IPの機能に利用

| | | | |
|------------------------|--------|-------------------------|--|
| IPヘッダ (実IPアド レス) | UDPヘッダ | IPヘッダ (仮想IPア ドレス) | |
|------------------------|--------|-------------------------|--|

作成者 : Adam Dunkels

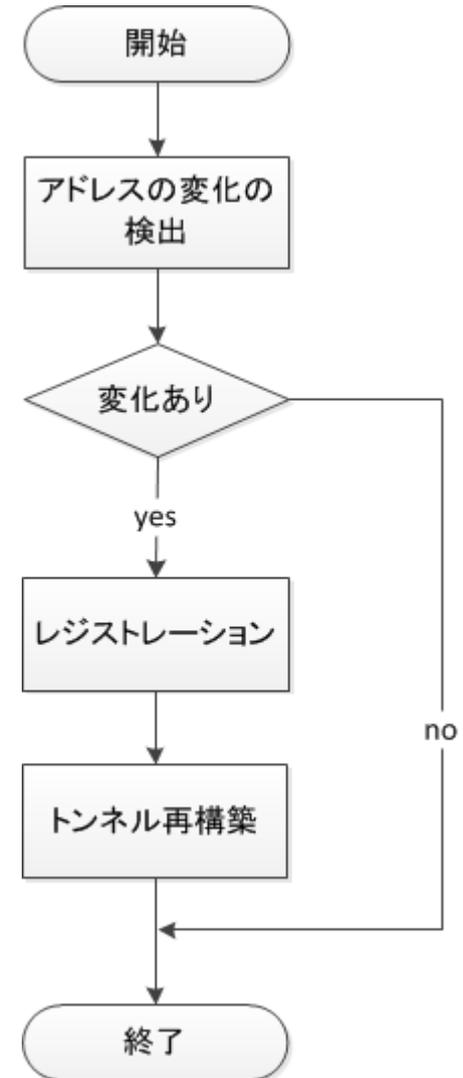
lwIP wiki : <http://savannah.nongnu.org/projects/lwip/>

フレームワークの構成

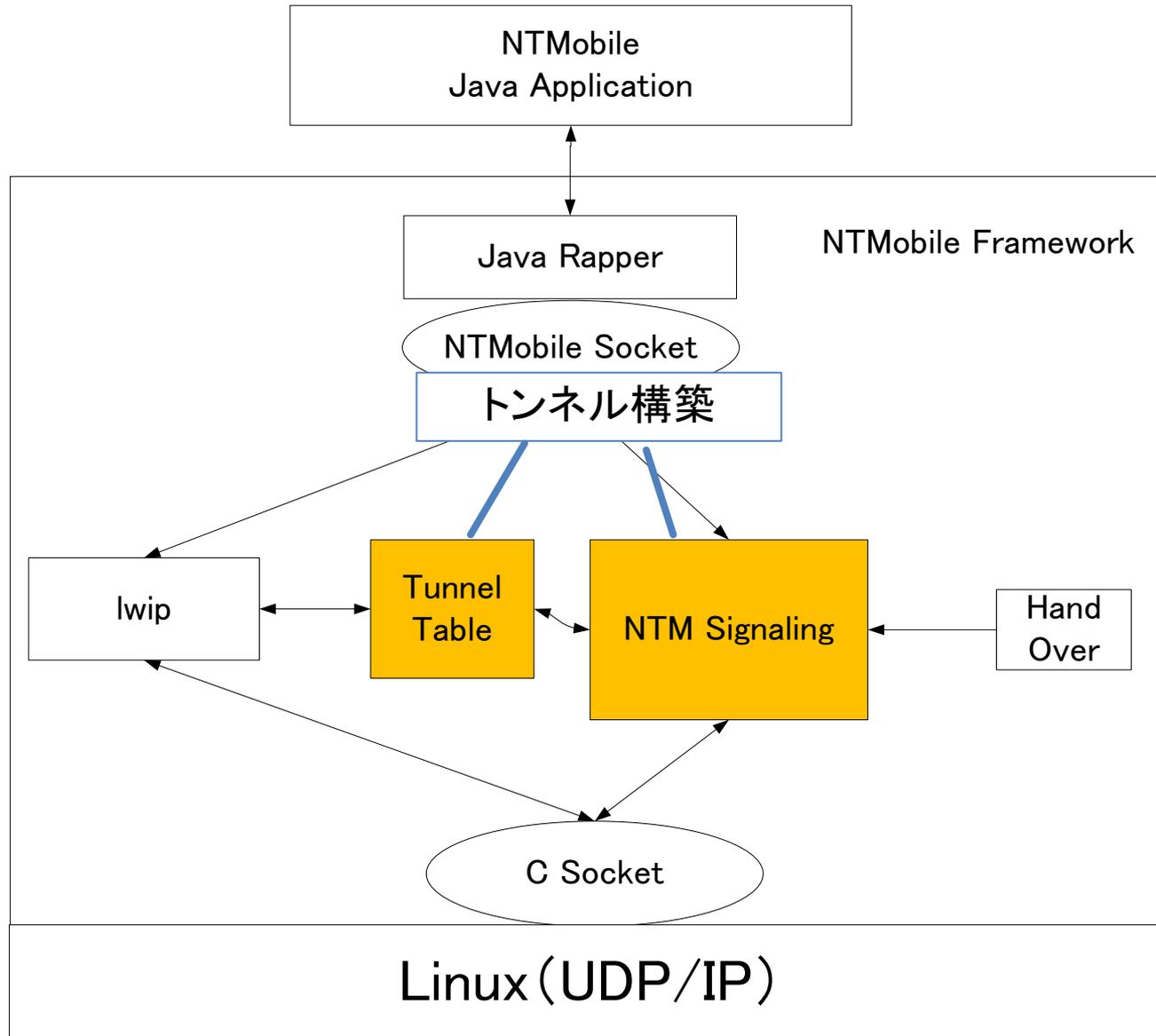


Handoverモジュール

- 一秒に一回の呼び出し
- アドレス変化を検出した場合DCへ、端末情報の上書き(レジストレーション)とNTMobile端末間のトンネルの再構築依頼

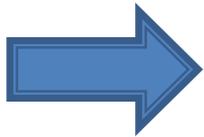


フレームワークの構成



まとめ

- フレームワークの実装
 - 各モジュールの機能
 - Javaラッパー
 - ハンドオーバ処理



Android端末への実装

- 今後の予定
 - フレームワークの試験・性能評価