NAT-f を利用した SIP の NAT 越え通信の提案

三浦　健吉†　　鈴木　秀和‡　　渡邊　晃†

名城大学理工学部†　　名城大学理工学部研究科‡

1.　はじめに

いつでもどこからでもネットワークにアクセスできるユビキタスネットワークの需要が広まっている．しかし，ホームネットワークは一般的にプライベートアドレスで構築されるため，インターネット側の外部ノードからホームネットワーク内の内部ノードに対して通信を開始できないという NAT 越え問題がある．

我々は，外部ノードと NAT ルータが連携することにより，NAT 越え問題解決する NAT-f（NAT-free protocol）を提案している[1]．しかし，現在の NAT-f は SIP（Session Initiation Protocol）に対応できないという課題があった．

そこで本稿では，NAT-f を利用した SIP の NAT 越え手法について提案する．

2.　NAT-f

図 1 に NAT-f の概要を示す．外部ノード EN（External Node）と NAT ルータには NAT-f 機能が実装されており，内部ノード IN （Internal Node）及び DDNS（Dynamic DNS）サーバは既存のものを利用する．DDNS サーバには，IN の名前とそれに対応する NAT ルータのアドレスを登録しておく．なお，プライベート IP アドレスを P*n*，グローバル IP アドレスを G*n* と表記する．

EN は IN へ通信を開始する際，DDNS サーバへ名前解決を依頼する．DDNS サーバは NAT ルータの IP アドレス G2 を返答する．EN は IP 層において取得した NAT ルータの IP アドレスを仮想アドレス V1 へ書き換えてアプリケーションに渡す．このとき，NAT ルータの IP アドレス，仮想 IP アドレス，及び IN の名前の関係を NRT(Name Relation Table)に保存する．この処理により，EN は通信相手の IP アドレスを V1 として認識する．

EN は宛先 IP アドレスが V1 である最初の TCP/UDP パケットを送信する際，一時的にこのパケットをカーネル内に待避させ，NAT ルータとの間で NAT-f ネゴシエーションを実行する．NAT ルータは， EN と IN 間の通信に必要な NAT テーブルを生成し，自身の IP アドレス G2 とマッピングされたポート番号 m を EN に返答する．

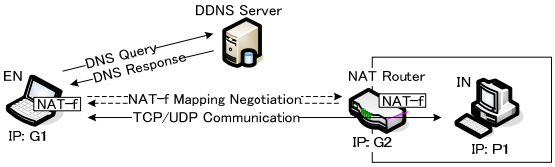


図1　NAT-f

以後，EN はネゴシエーションの結果に従い，送信するパケットの宛先 IP アドレスを V1 から G2 へ，宛先ポート番号をマッピングされたポート番号 m に変換する．NAT に届けられたパケットの宛先 IP アドレス・ポート番号は NAT テーブルに従って変換され，IN に届けられる．以上の動作により NAT 外部からの通信開始が可能となる．

3.　SIP

図 2 に SIP のシーケンスを示す．両端の UA（User Agent）が 2 台の SIP Proxy を経由してシグナリングを行う場合について述べる．

事前準備として，UA2 は SIP Proxy 2 に対して REGISTER により自身の URI （Uniform Resource Identifier）である URI2 と INVITE を受け取る際に使用するトランスポートアドレス G2:d1 の登録を要求する．SIP Proxy 2 は受信した URI とトランスポートアドレスを DB（Data Base）に登録し，UA2 に対して 200 OK を返答する．

通信開始時，UA1 は INVITE により UA2 とのセッションの確立を要求する．INVITE には，UA2 とのセッションを確立する際に UA1 が使用するトランスポートアドレス G1:s2 が記載されており，SIP Proxy 1 を中継し，SIP Proxy 2 に転送される．

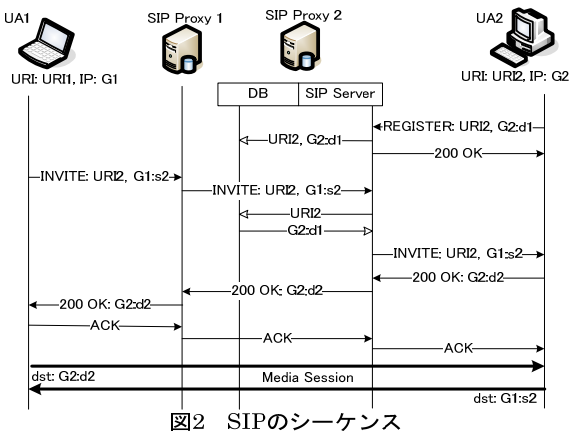


図2　SIPのシーケンス

"Proposal of NAT traversal for SIP utilizing NAT-f"
† Kenkichi Miura and Akira Watanabe
Faculty of Science and Technology, Meijo University
‡ Hidekazu Suzuki
Graduate School of Science and Technology,Meijo University

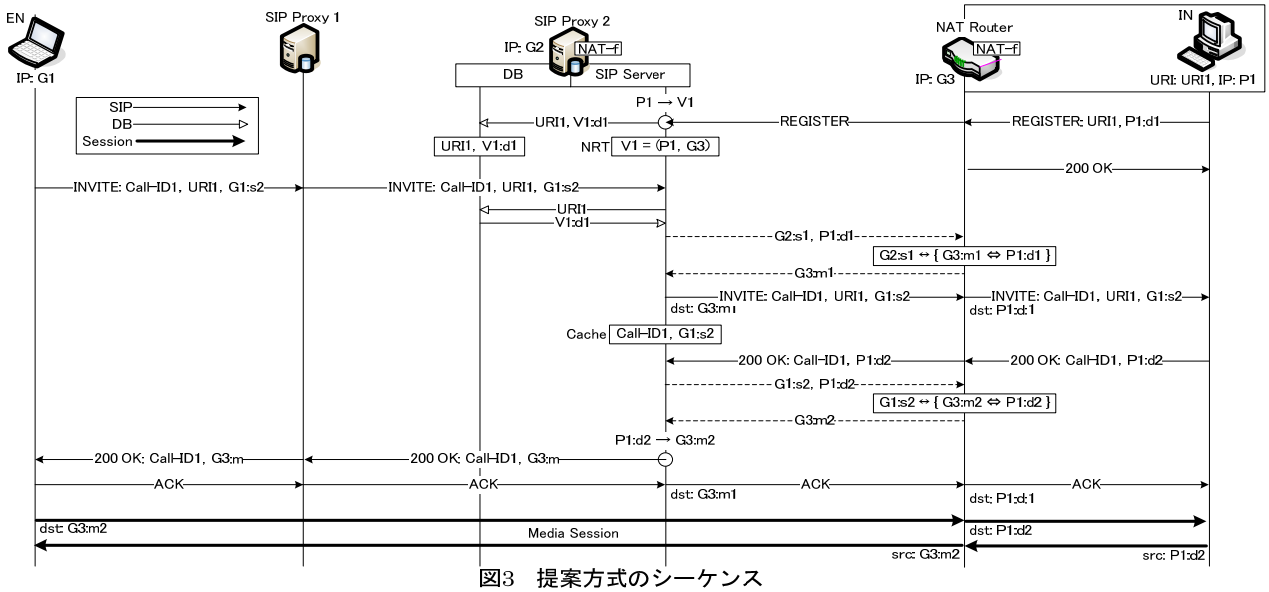


図3　提案方式のシーケンス

SIP Proxy 2 は，URI2 の名前解決を行い，INVITE を UA2 へ転送する．

INVITE を受信した UA2 は，200 OK を返答する．200 OK には，UA2 が使用するトランスポートアドレス G2:d2 が記載されており，2 台の SIP Proxy を経由して UA1 まで転送される．

UA1 は ACK を返答した後，交換したトランスポートアドレスを用いて，UA2 と直接メディアセッションを確立する．

ここで，SIP において NAT-f を利用することを考える．SIP では両端の UA が自身の IP アドレスを TCP/UDP パケットの IP ペイロード部分に記載し，お互いに交換することで，通信相手の IP アドレスを認識する．そのため，NAT-f のように，IP 層で DNS 応答を書き換え，アプリケーションに仮想アドレスを認識させる手法が適用できない．

4.　提案技術

EN は SIP 機能を持つ一般端末とする．SIP Proxy 2 側に拡張した NAT-f を実装する．そして，SIP Proxy 2 が NAT ルータに対して NAT-f ネゴシエーションを実行することにより EN と IN 間の通信に必要な NAT テーブルを生成する．

図 3 に NAT-f を利用した SIP の NAT 越え通信のシーケンスを示す．NAT ルータには NAT-f 機能が実装されているものとする．

SIP Proxy 2 は REGISTER 受信時に，記載されている IP アドレスを P1 から仮想アドレス V1 へ書き換えてから DB に対して登録を実行する．また，SIP Proxy 2 は V1，P1，G3 の関係を NRT に保存し，200 OK を返答する．

UA1 からの通信開始時，SIP Proxy 2 は EN からの INVITE を受信すると DB の内容により，INVITE の転送先となる V1:d1 を取得する．ここで，宛先 IP アドレス V1 が仮想アドレスであるため，SIP Proxy 2 と NAT ルータ間で NAT-f ネゴシエーションを実行する．SIP Proxy 2 は INVITE の送信元トランスポートアドレス G2:s1，宛先ポート番号 d1，及び NRT に保存した IN のアドレス P1 の情報を NAT ルータに送信する． NAT ルータはこれらの情報を用いて，SIP Proxy 2 と IN の SIP ネゴシエーションに必要な NAT テーブルを生成後，マッピングされた G3:m1 を返答する．SIP Proxy 2 はこのマッピングされたポートに向けて INVITE を送信する．また，SIP Proxy 2 は一連のシーケンスで共通する Call-ID と EN が使用するトランスポートアドレス G1:s2 を対応付けてキャッシュする．

SIP Proxy 2 は，INVITE に対する 200 OK 受信すると，先ほど作成したキャッシュの情報を Call-ID1 で検索し，NAT ルータに対して再度 NAT-f ネゴシエーションを実行する．SIP Proxy 2 は EN のトランスポートアドレス G1:s2 と IN のトランスポートアドレス P1:d2 を通知して EN と IN 間の通信で必要な NAT テーブルを生成する．すなわち，SIP Proxy 2 の指示により，NAT ルータ内に EN と IN 間の通信に必要となる NAT テーブルが生成される．SIP Proxy 2 は 200 OK に記載されているトランスポートアドレスを P1:d2 から G3:m2 へ書き換え，転送する．

200 OK を受け取った EN は，ACK 返答した後，交換したトランスポートアドレスに従い，メディアセッションを確立することができる．

5.　まとめ

NAT-f を利用した SIP の NAT 越えに手法ついて検討した．今後は，実装と動作検証を行う．

参考文献

[1] 鈴木 秀和，宇佐見 庄五，渡邊 晃，“外部動的マッピングにより NAT 越え通信を実現する NAT-f の提案と実装”，情報処理学会論文誌，Vol.48，No.12，pp.3949-3961，Dec.2007

NAT-fを利用したSIPのNAT越え通信方式の検討

名城大学大学院　理工学研究科
三浦 健吉，鈴木秀和，渡邊晃

はじめに

- 家庭内のネットワークはプライベートIPアドレスで構築される場合が多い
  ⇒ NAT(Network Address Translator)*が使用される

- NAT
  - 1個のグローバルIPアドレスを複数のホストで共有する技術
  - 一般家庭のブロードバンドルータに搭載されている

*本スライドでは，NAPTまたはIPマスカレードを含めてNATと呼ぶ

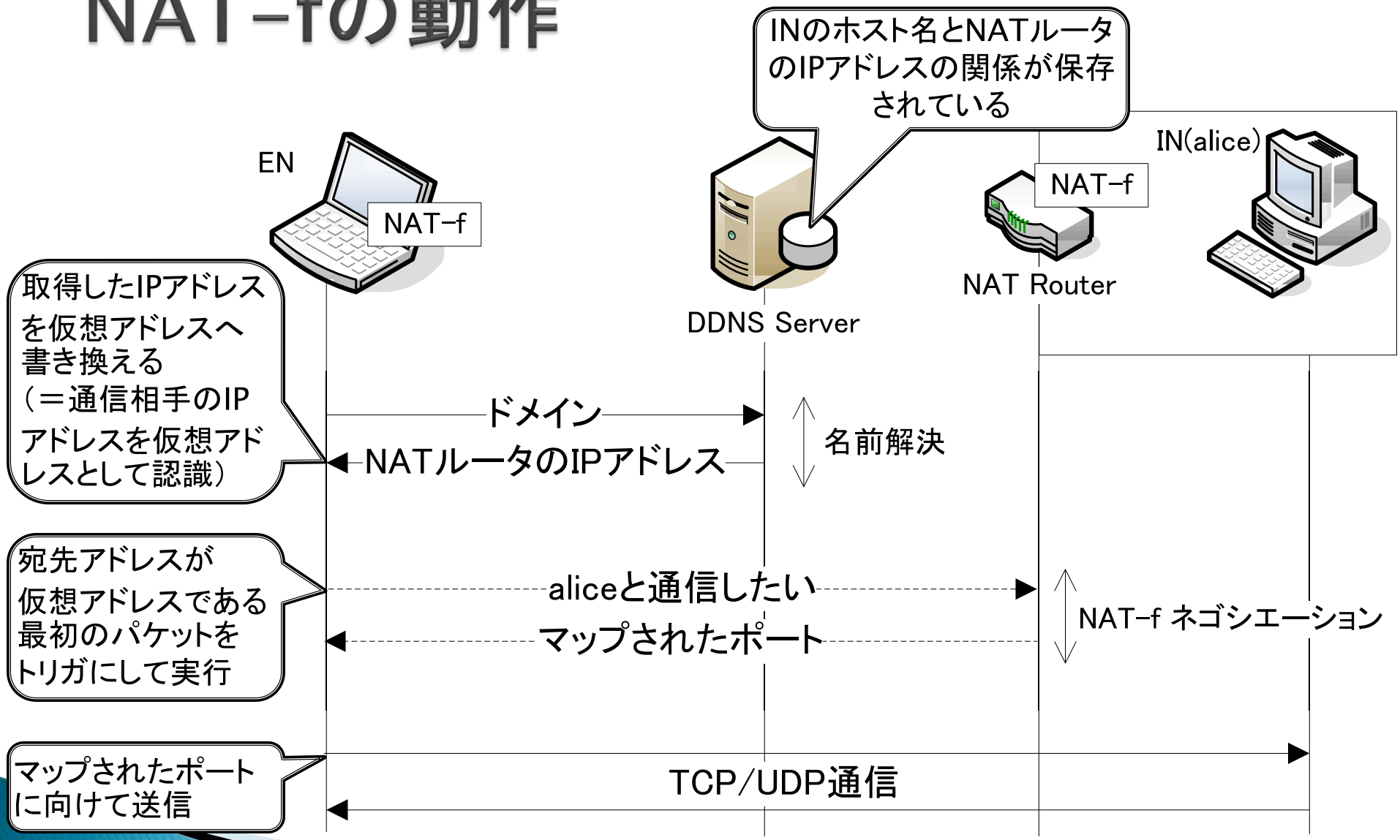
NAT越え問題とNAT-f

- NAT越え問題
  - NAT外側の端末から内側の端末に対して通信を開始できない
  - NAT外側からはNAT内側の端末が見えないため

- NAT-f(NAT-free protocol)*
  - 通信に先立ち，外側の端末からNATに対してネゴシエーションを行い，NAT内側の端末との通信に必要なNATテーブルを生成することにより，NAT越え問題を解決

*鈴木秀和，宇佐見庄五，渡邊晃：”外部動的マッピングによりNAT越えを実現するNAT-fの提案と実装”，情報処理学会論文誌， No.12, pp.3949-3961.

NAT-fの動作



EN: External Node 外部ノード
IN: Internal Node 内部ノード

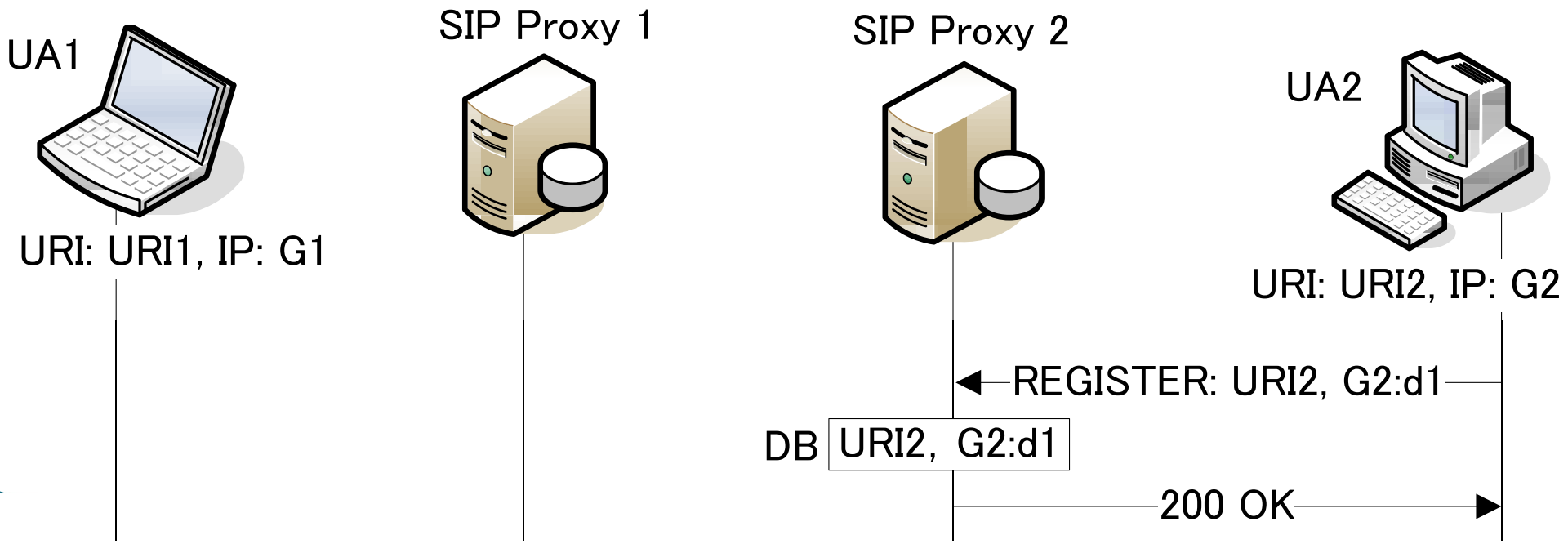
SIP(Session Initiation Protocol)

- IP電話や情報家電などで使用されている

- セッションを確立するためのプロトコル
  - 通信相手の呼び出し
  - 通信方式の交渉
  - セッションを確立すると、通信が開始される

SIPの基本シーケンス(1/2)

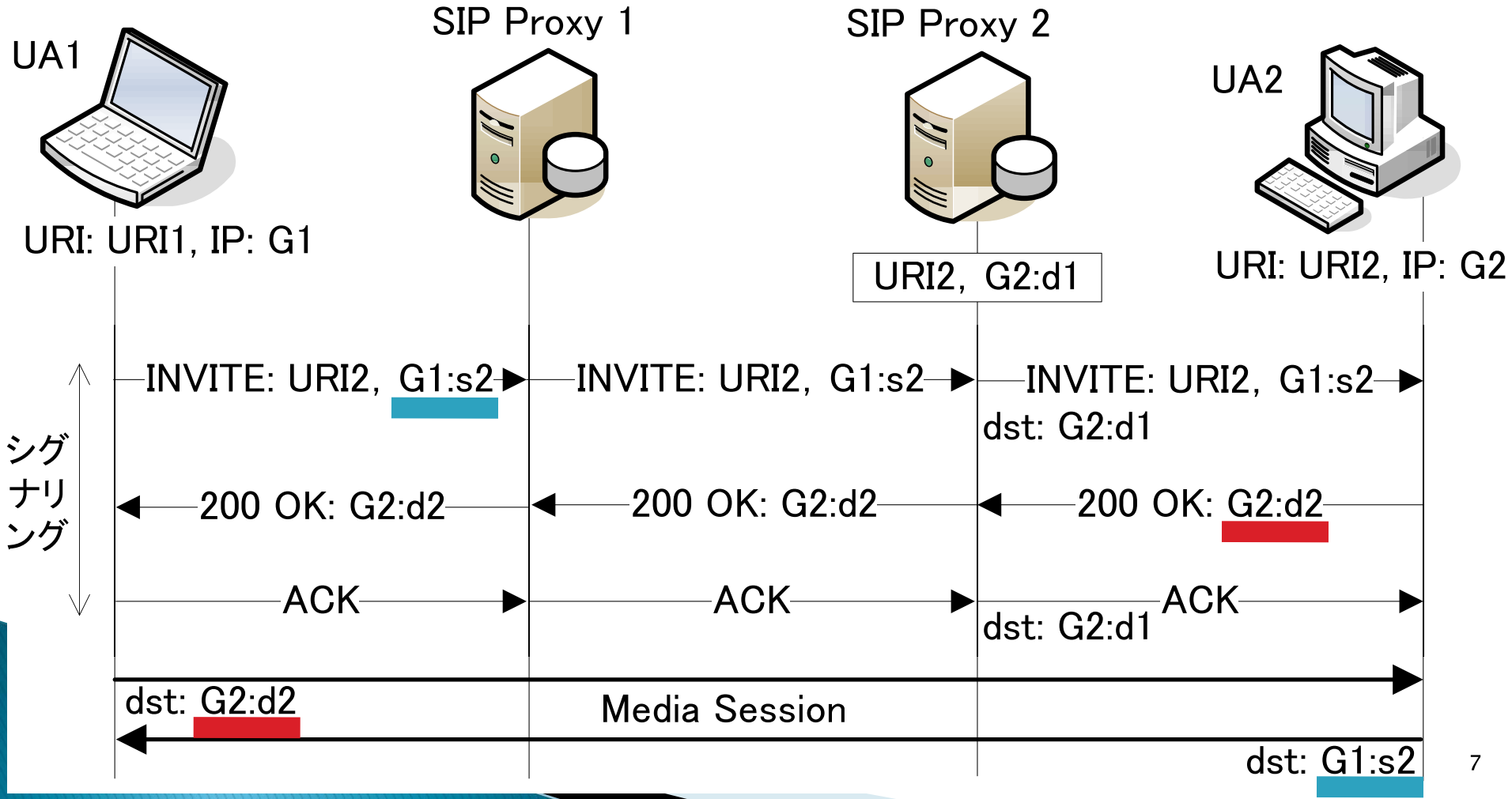
- 事前に端末情報を登録する
  - SIP Proxy 1はUA1が使用するサーバ
  - SIP Proxy 2はUA2が使用するサーバ



トランスポートアドレス: IPアドレスとポート番号の組
s1/d1: SIPメッセージ送受信時に使用するポート
s2/d2: 音声通信で使用するポート

SIPの基本シーケンス(2/2)

- トランスポートアドレスを交換し，セッションを確立する

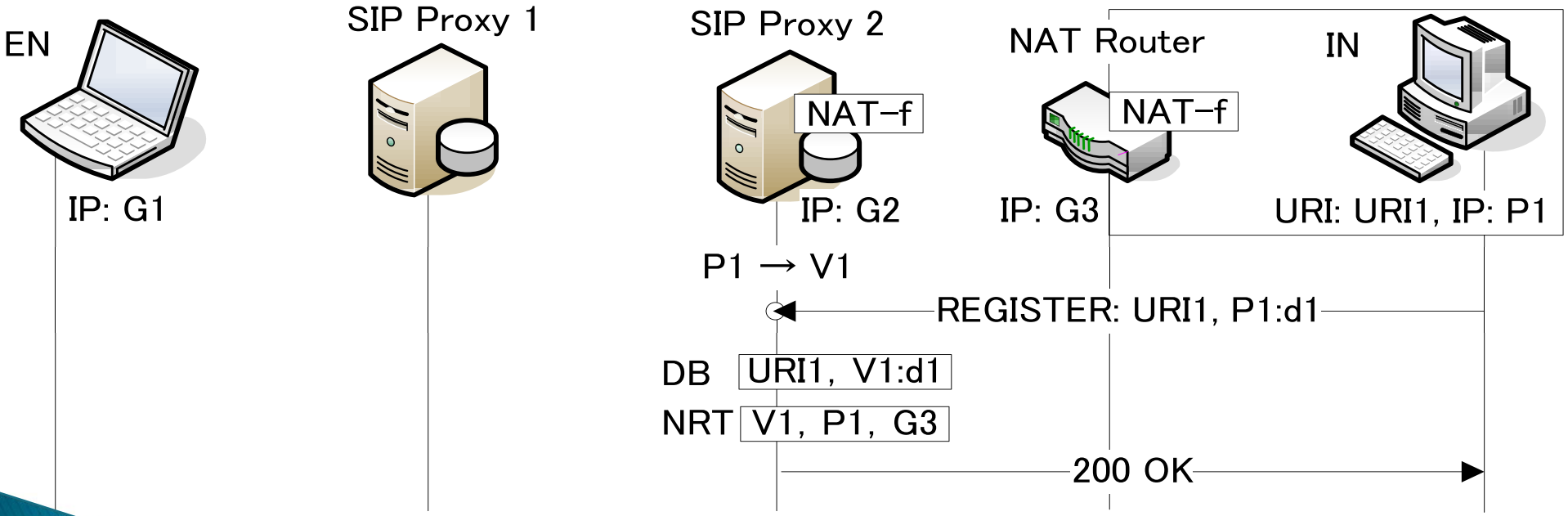


SIPにおけるNAT越え問題

- 通常のNAT越え問題
  - NATの外部から内部に向けてシグナリングを開始できない
- IPペイロード内（SIPメッセージ内）にIPアドレスが埋め込まれる
  - SIPメッセージがNATを通過するとアドレスの不整合が生じる
    - IPヘッダ内のIPアドレス → NATで変換される
    - IPペイロード内のIPアドレス → NATで変換されない

提案方式(1/3)

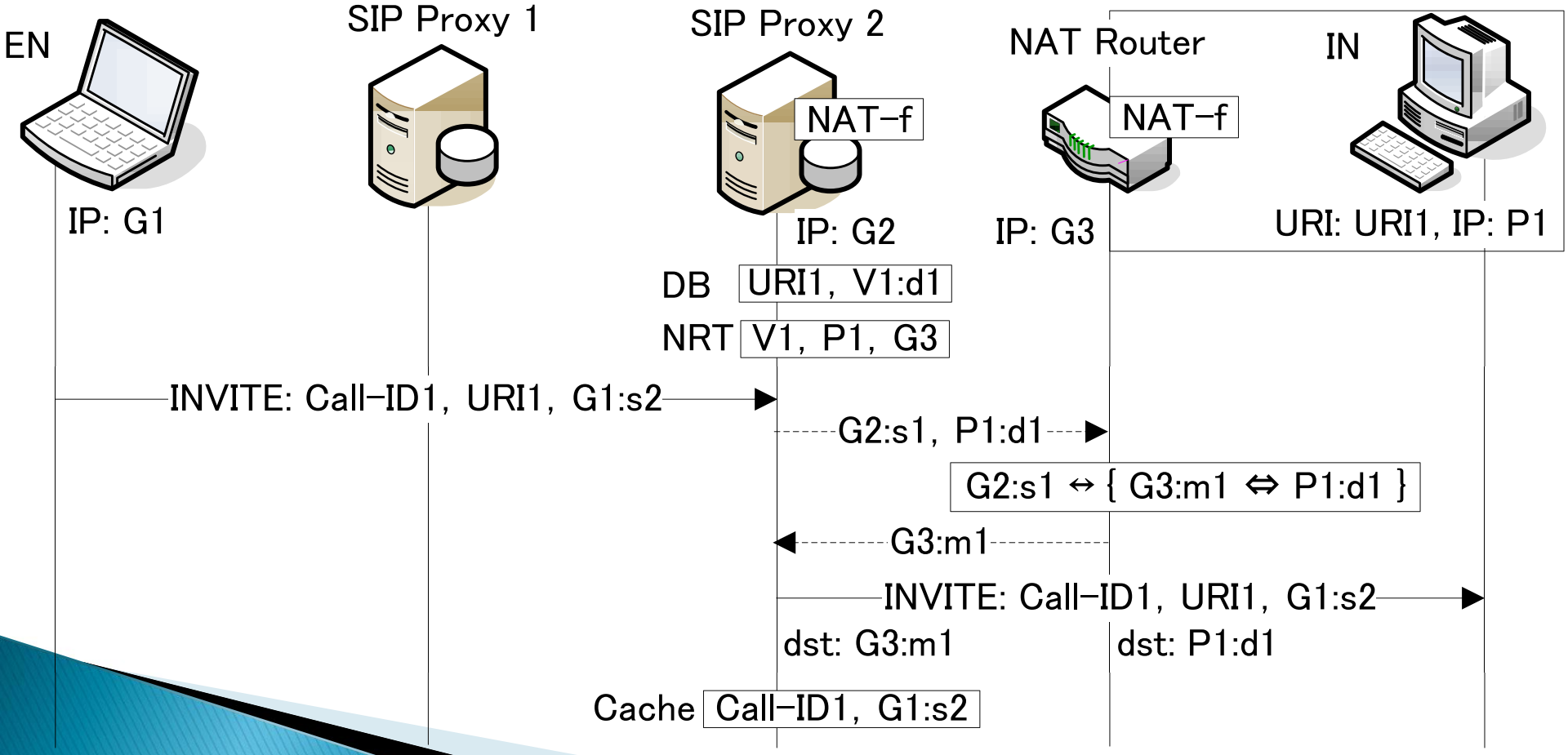
- 端末情報の登録
  - DBには仮想アドレスが登録される
  - アドレス書き換えと同時に，NRT生成



NRT: Name Relation Table 名前関連テーブル

提案方式(2/3)

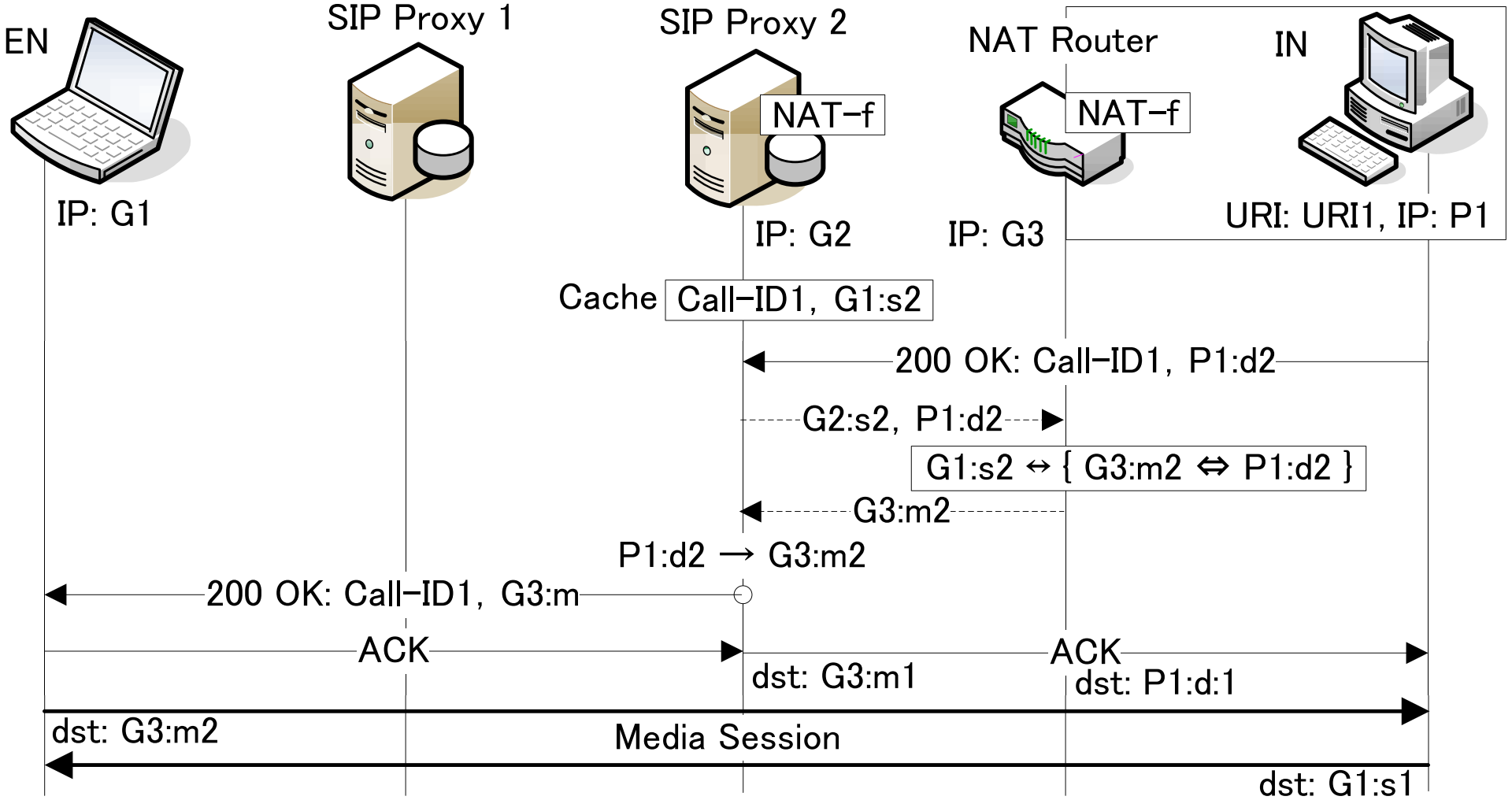
- SIPプロキシ2とIN間のNAT越え



Call-ID: 一連のシグナリングで共通するID

提案方式(3/3)

- ENとIN間のNAT越え



むすび

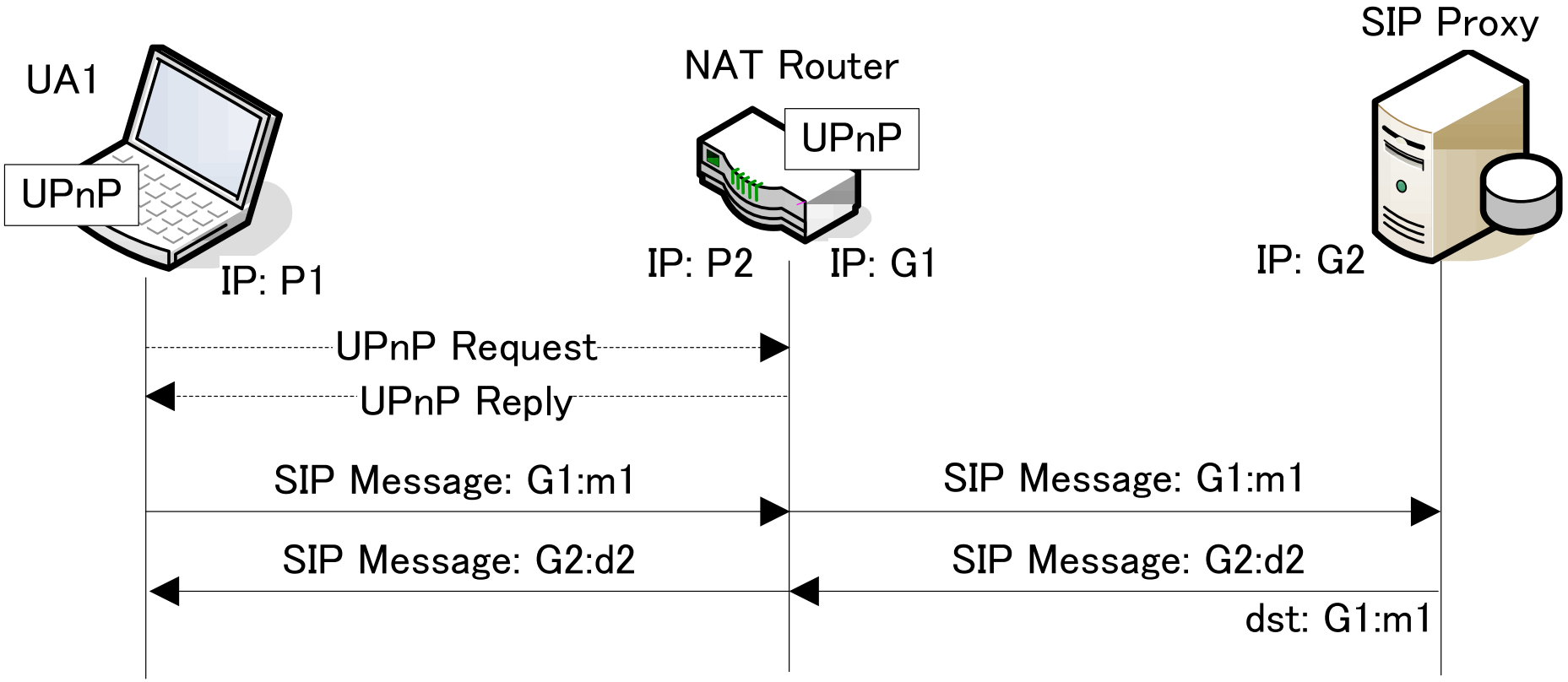
- NAT-fを利用したSIPのNAT越え通信の方法について検討した
- 今後は，実装と動作確認を行う．

補足資料

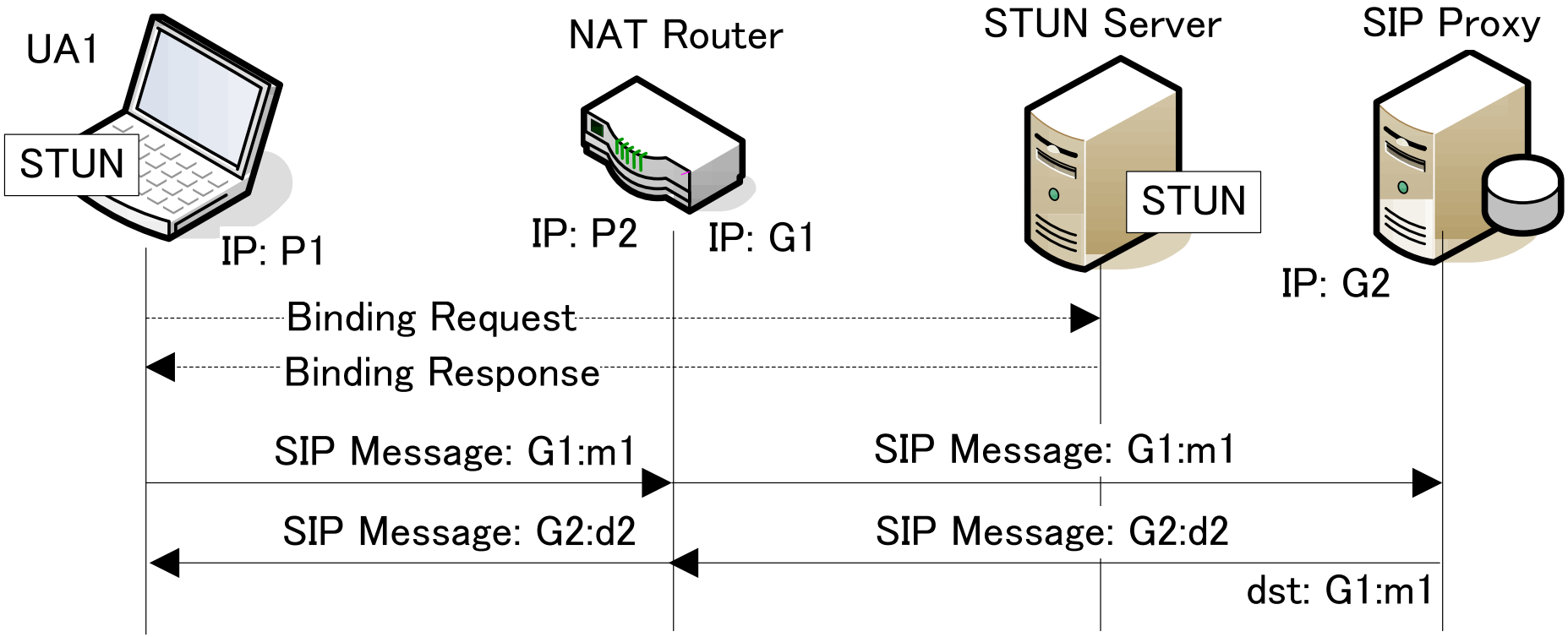
既存技術

- アドレス埋め込み型：UPnP、STUN
  - SIPメッセージを送信する際に，予めNAT越え問題が解決済みのIPアドレス・ポート番号を埋め込んでおく方式
- アドレス書き換え型：ALG
  - アドレス書き換え型は，NATがSIPメッセージ中のIPアドレス・ポート番号の書き換える方式
- サーバ中継型：TURN
  - サーバ中継型は，グローバルネットワークに設置されたサーバを中継し，通信を行うことで，NAT越えを実現する方式

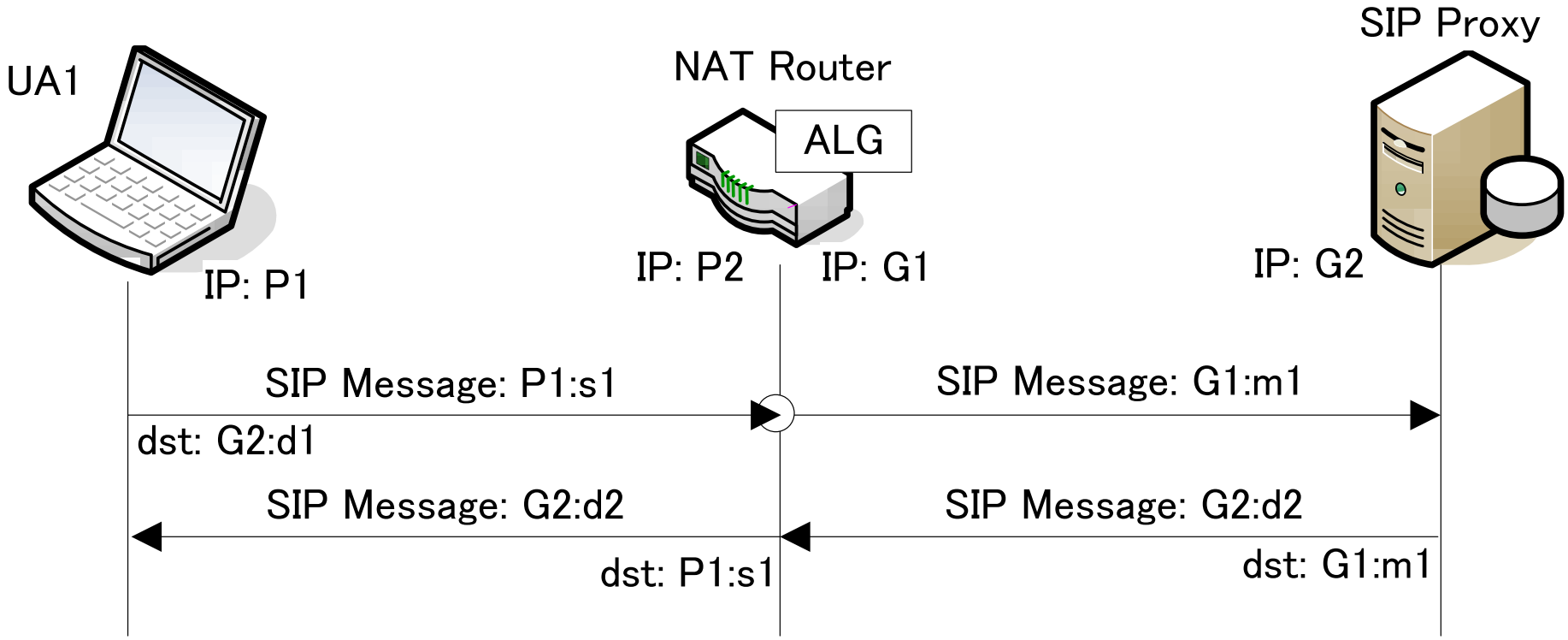
UPnP: Universal Plug and Play



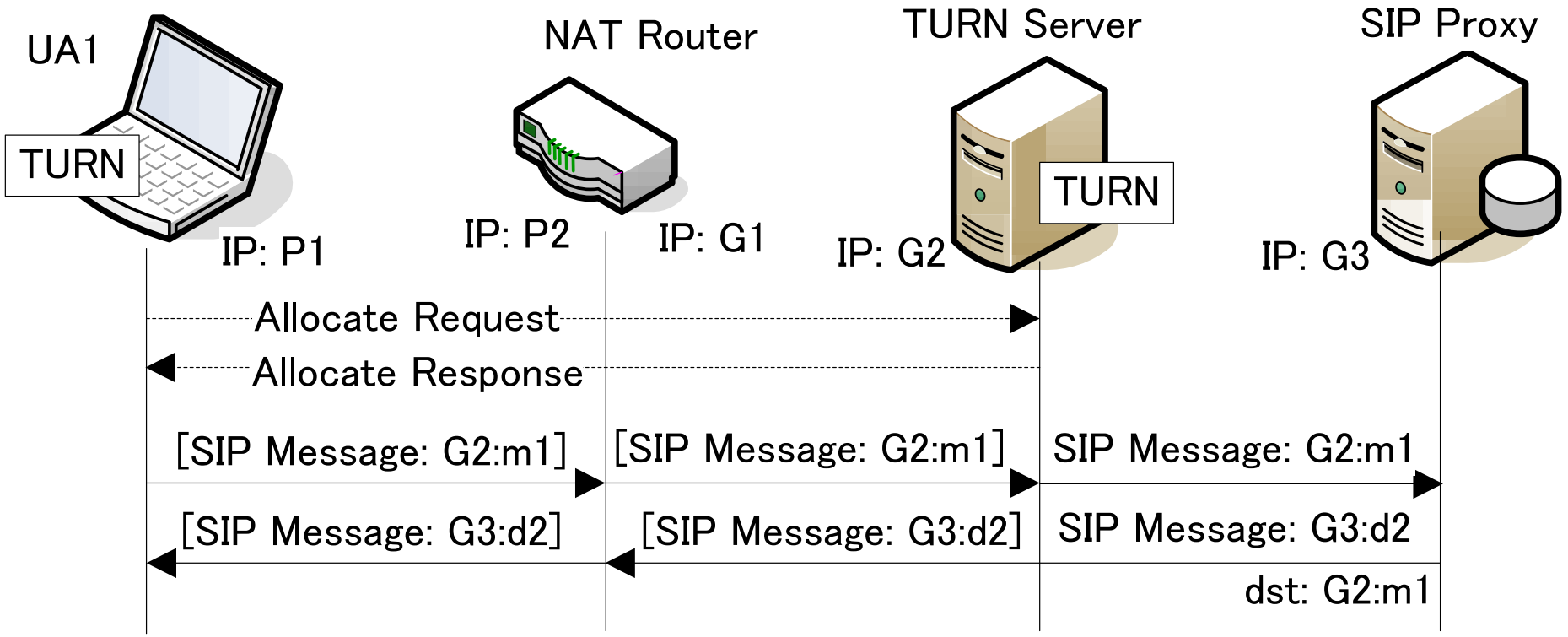
STUN: Simple Traversal of UDP through Network Address Translators



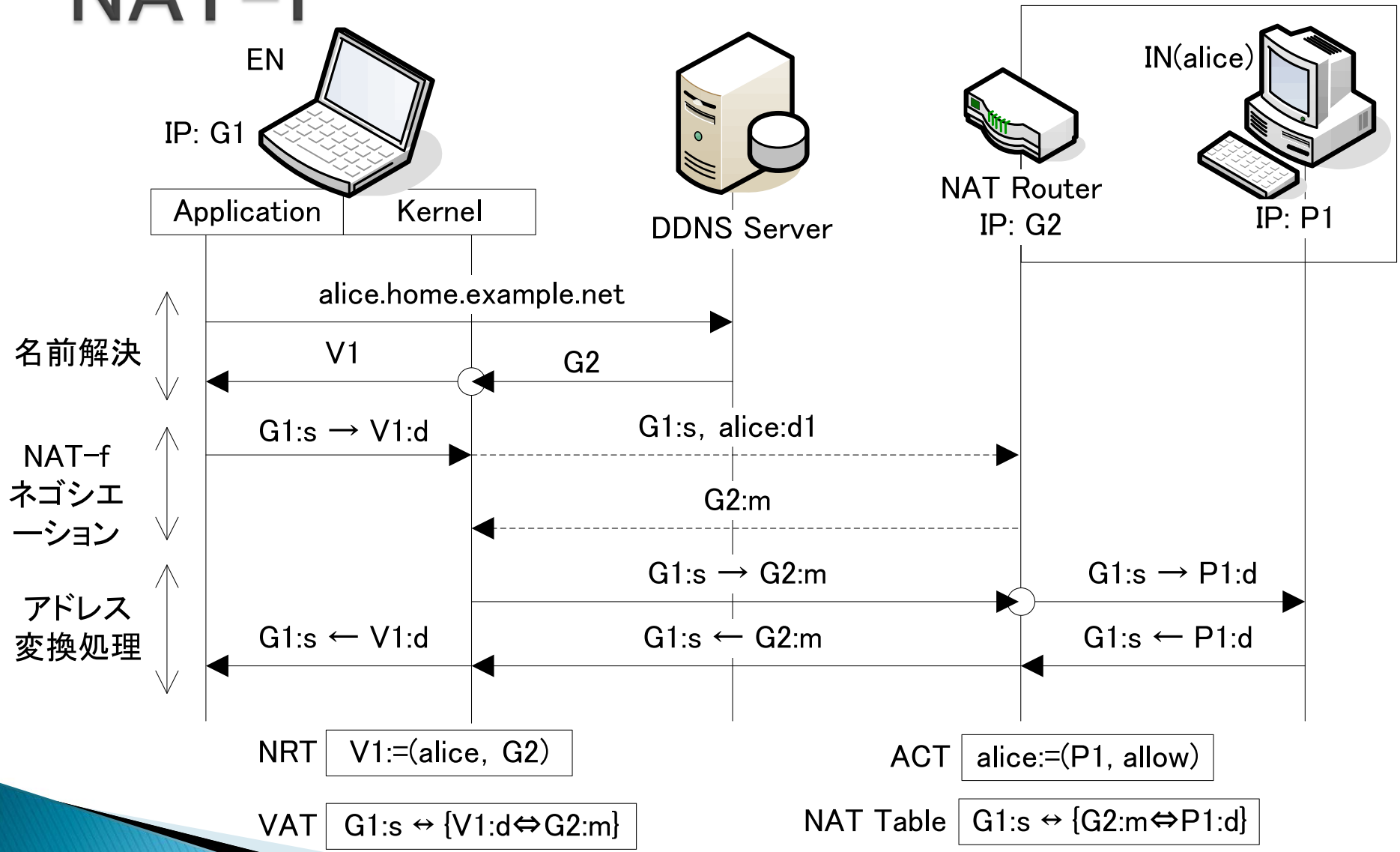
ALG: Application Level Gateway



TURN: Traversal Using Relay NAT



NAT-f


EN
IP: G1
Application
Kernel
DDNS Server
NAT Router
IP: G2
IN(alice)
IP: P1
alice.home.example.net
名前解決
V1
G2
G1:s → V1:d
G1:s, alice:d1
NAT-f
ネゴシエ
ーション
G2:m
G1:s → G2:m
G1:s → P1:d
アドレス
変換処理
G1:s ← V1:d
G1:s ← G2:m
G1:s ← P1:d
NRT
V1:=(alice, G2)
VAT
G1:s ↔ {V1:d⇔G2:m}
ACT
alice:=(P1, allow)
NAT Table
G1:s ↔ {G2:m⇔P1:d}