

# 第 12 部

## 移動ノード



# 第 1 章

## はじめに

ラップトップ型やノート型パソコンに続き、ワークステーションの小型計量化も進んでいる。遠くない将来に、ノートパソコンと同じ大きさのワークステーションの登場が期待される。

小型のワークステーションの出現により、現在のように同じ部屋の中だけで固定して用いる環境から、持ち運んで、または移動しながら使用できる環境に変化する。

ワークステーションを単体で用いるのであれば、現状のものをそのまま小型化した可搬型ワークステーションを持って移動しても問題はない。しかし、ワークステーションがこのように広く使用されるようになった背景には、ネットワークを用いた分散処理環境への対応があることを忘れることはできない。つまり、ネットワークを介してあらゆる資源 (CPU、メモリ、データ等) を共有できるようになっているわけである。ネットワークがなければ、ワークステーションはただの速いパソコンにすぎない。

このように、ワークステーションはネットワークへ接続して使用することが一般的であるが、現在のネットワークは固定されたシステムにのみ対応していて、ホストが移動するということを考慮していない。このため、せっかく小型になった可搬型のワークステーションがケーブルに固定されてしまい、その機動性が活かされない。

現在、ワークステーションを接続するネットワークで広く採り入れられているプロトコルは、Internet で使用されている TCP/IP である。この TCP/IP プロトコルは、今のところ移動するノードに対応していない。これを解決するためにまったく新しいプロトコルを導入したのでは、現状のネットワークとの互換性が得られず、すぐに実用にはならない。

TCP/IP プロトコルが可搬ノードに対応できない最大の要因は経路制御にある。このプロトコルでは、データをいくつかのパケットと呼ばれるブロックに分解し、それぞれに宛先をつけて転送している。そのパケットを宛先まで転送する時の経路制御は、ホストを識別するための Internet アドレスをもとにして行なわれるため、可搬ノードが他のネットワークに移動するとパケットを転送する経路がわからなくなってしまう。

このことは、個人の住所と名前に対応させるとよく理解できる。つまり、手紙 (パケット) の宛先には住所 (IP アドレス) しか書いてないので、住人 (ホスト) が他へ移動してもその住所にしか届かないし、後から来た他人に届いてしまうかも知れないということである。現状では、この住所と名前の対応が固定されていて引越しすることは許されないため問題にはならない。しかし、ホストが移動するようになると、パケットを IP アドレ

スの場所へ転送しても正しく届かないことになる。

移動するノードを扱うためには、パケットの経路を決めるための情報とホストを特定するための情報が独立していなければならない。住所と名前の対がいつでも一定でないことと同様に、IP アドレスとホスト名の対応も固定しない方が自然である。

以上のように、Internet に可搬ノードを接続するためには、IP アドレスと経路制御の取り扱いについて変更を加えなければならない。これには二つの方式が考えられる。

一つは、移動することによって IP アドレスを変化させるもので、動的アドレス更新型と呼ぶ。この方式はコードレス電話に例えることができる。すなわち、移動すると他の親機に接続するので電話番号 (アドレス) が変化する。

もう一つは、IP アドレスをそのまま経路制御に使用しないことで可搬ノードに対応する、動的経路更新型である。この方式は自動車電話に例えられる。移動してもデータの通る経路が変わるだけで、電話番号は変化しない。

本論文では可搬ノードを含む Internet を構築する際の IP アドレスと経路制御の取り扱いについて考察する。そして、現在の Internet を変更せずに使用できる動的アドレス更新型の IP アドレス割り当て方式の実装について述べる。

IP アドレス割り当て方式ではノードをホスト名で識別するようにし、IP アドレスはパケットの経路制御にのみ使用する。可搬ノードの IP アドレスは接続したアドレス割り当てサーバから与えられ、その IP アドレスとホスト名の対応はネームサーバによって行なわれる。このため、ネットワークを使用するアプリケーションは、可搬ノードの IP アドレスが変化したことを認識する必要はない。また、途中のパケットは普通の IP パケットと何らかわるところがないので、現在の Internet にそのまま対応することができる。

この方式は、移動しながらネットワークを使用するといった自動車電話的な使い方をすることはできないが、既存の Internet プロトコルに変更を加える必要がない点や、他のノードに影響を与えないことなどを考えると、実用的な点において他の方式よりも優れている。

## 1.1 本論文の構成

本論文の構成は以下のようになっている。

2章では Internet について述べる。はじめに Internet の概要を説明し、つぎに可搬ノードを Internet に接続する時の問題点について考察する。

3章では、可搬ノードを Internet に接続する新しい方法について述べる。この新しい方式は、動的経路更新型と動的アドレス更新型の二つに分類することができる。それぞれの型から、いくつかの方式をとりあげ得失を考察し、設計する方式を選ぶ。

4章では IP アドレス割り当て方式の設計について述べる。はじめに IP アドレス割り当て方式の構成を説明し、それぞれの構成要素についてプロトコルを設計する。

5章では IP アドレス割り当て方式の実装について述べる。4章で述べた設計にしたがって、構成要素を実装する。この時の詳細について説明する。

6章では、実装した IP アドレス割り当て方式の評価について述べる。はじめに実験環境の説明をし、実際に使用した時の評価を述べる。次に、IP アドレス割り当て方式にどのような応用方法があるかを考察する。

7章では今後の課題について述べる。今後の課題には、IP アドレス割り当て方式の欠点の解決やセキュリティの問題等がある。

8章では、本論文のまとめを述べる。

## 第 2 章

# Internet

この章では、Internet の特徴について説明し、可搬ノードを接続する場合に問題となる点を考察する。

### 2.1 Internet の概要

ワークステーションがその能力を發揮するためには、ネットワークを用いた資源の共有が欠かせない。

初期の段階では、このネットワークは部屋単位、または建物、構内といった狭い範囲にある計算機を接続するために使用されていた。多くのネットワークはお互いに独立して、他のネットワークと情報を交換することはできなかった。

しかし、他の離れた場所と情報を交換するという要求により、この独立したネットワークがお互いに接続されるようになった。このようにネットワーク同士を接続することを、インターネットと呼ぶ。(図 2.1 参照)

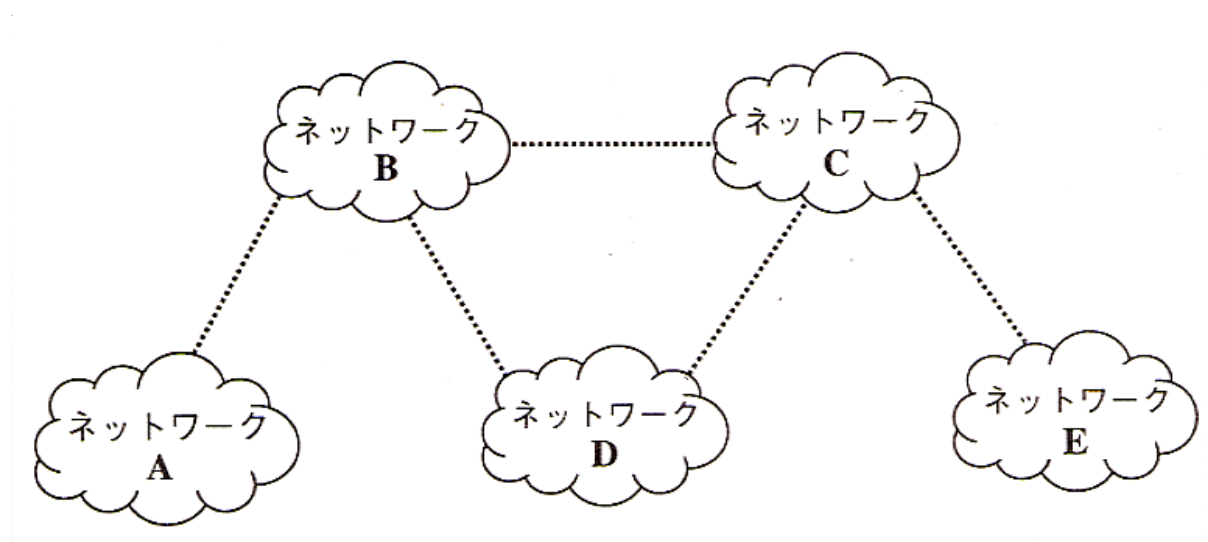


図 2.1: インターネット

このインターネットの研究がアメリカの DARPA (Defense Advanced Projects Research Agency : 国防総省高等研究計画局) の資金により行なわれ、TCP/IP (Transmission Control Protocol / Internet Protocol) [5] と呼ばれるプロトコルが開発された。この TCP/IP を使用したインターネットのことを、とくに Internet(大文字ではじまる) と呼ぶ。

TCP/IP ではデータの通信をパケット交換によって行なう。パケット交換とは、データをいくつかのブロック (パケット) に分解し、それぞれに宛先をつけて転送する方式である。この宛先を示すものが Internet アドレス (IP アドレス) である。

### 2.1.1 Internet アドレス

Internet アドレス (IP アドレス) はあるネットワークの特定のホストを識別するために使用される。このために、同じ IP アドレスを持った異なるノードは存在してはならない。

IP アドレスは 4 バイト長であり、TCP/IP がネットワーク同士の通信に使われることから、ネット部 (netid) とホスト部 (hostid) の二つの部分から構成されている。

ネット部はある特定のネットワークを示し、ホスト部がその中のホスト (ノード) を決定する。ネット部とホスト部の長さは、ネットワークの規模によって決められる。

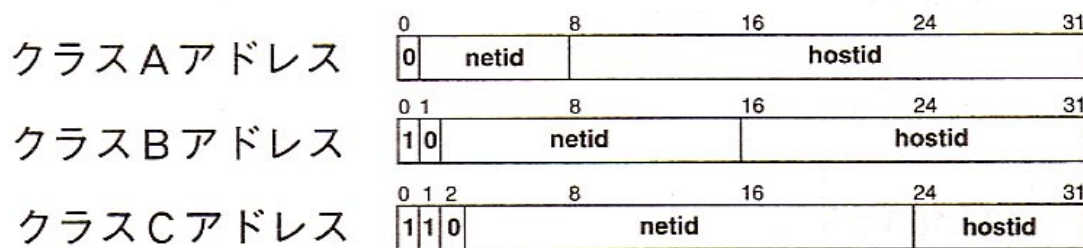


図 2.2: IP アドレスの構成

図 2.2 のように、IP アドレスはネットワークの規模に応じて三つのクラスにわかれている。多数のホストを持つ大規模ネットワークにはクラス A アドレスが与えられ、ネットワークの規模が小さくなる順にクラス B、クラス C アドレスが与えられる。

IP アドレスのホスト部を 0 に置き換えたものを特にネットワークアドレスという。ネットワークアドレスは、パケットの経路制御に使用される。経路制御については 2.1.2 節で詳しく説明する。

IP アドレスは 1 バイト毎に区切って、“.” でつなげた 4 個の 10 進数で表記することが一般的である。例えば 2 進数で示された IP アドレス

10000010 10011001 00011100 00000001

は

130.153.28.1

と表記される。

### 2.1.2 経路制御

すべてのネットワークをお互いに結合させることは、ネットワークの数の増加にともない困難となる。このため、直接つながっていないネットワークへパケットを転送する場合には、いくつかのネットワークを経由していかなければならない。例えば図 2.1 の場合、ネットワーク A とネットワーク E のノードが通信するには、ネットワーク B、ネットワーク C を経由しなければならない。このような、パケットの経路を選択する方法を経路制御 (routing) と呼ぶ。

Internet では、ゲートウェイ (gateway) と呼ばれるノードが経路制御を行なっている。ゲートウェイは図 2.3 のように複数のネットワークを接続するため、それぞれのネットワークに接続する回線毎に IP アドレスが付けられている。

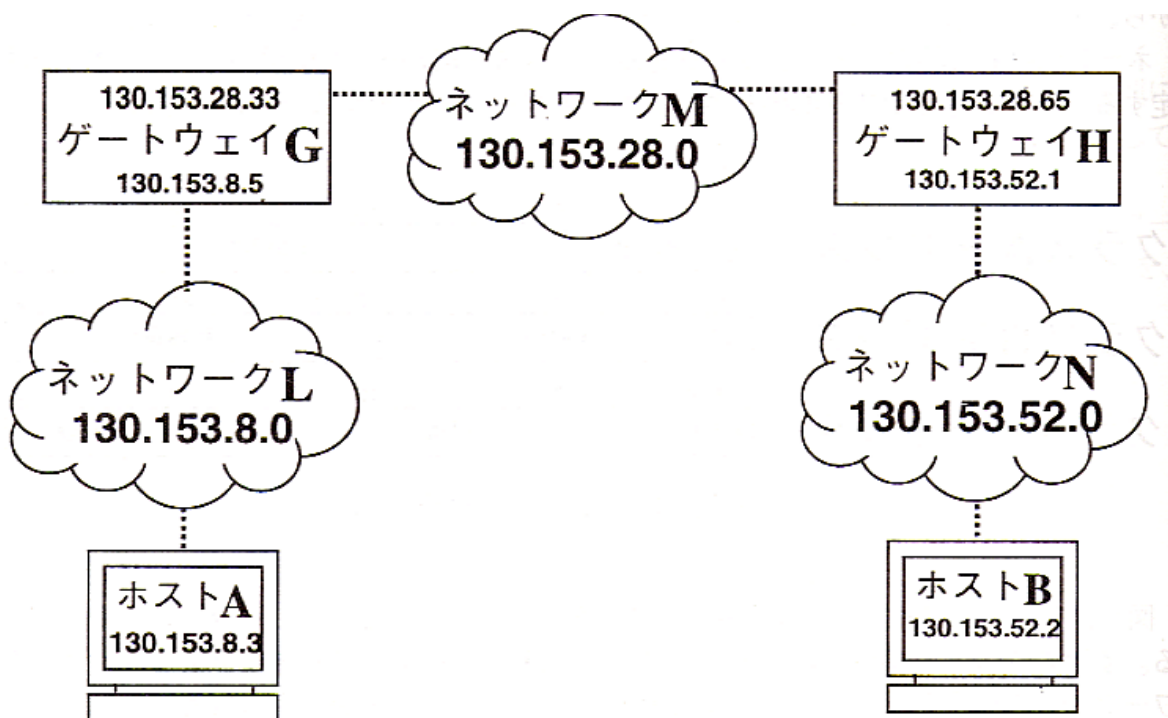


図 2.3: ネットワークを接続するゲートウェイ

ゲートウェイは、経路制御を行なうための情報が入ったテーブル (routing table) に基づいてパケットの経路を決定する。図 2.3 のネットワークでは各ゲートウェイのルーティングテーブルは表 2.1 のようになる。

中継するゲートウェイの欄に\*が付いているアドレスは、自分自身のアドレスである。これは、転送先のネットワークに直接接続しているため、他のゲートウェイを経由していく必要はなく、自分でパケットを直接転送できることを示している。

Internet での経路制御アルゴリズムは、転送先の IP アドレスのネット部によりパケッ



ゲートウェイ G		ゲートウェイ H	
転送先のネットワーク	中継するゲートウェイ	転送先のネットワーク	中継するゲートウェイ
130.153.8.0	*130.153.8.5	130.153.8.0	130.153.28.33
130.153.28.0	*130.153.28.33	130.153.28.0	*130.153.28.65
130.153.52.0	130.153.28.65	130.153.52.0	*130.153.52.1

表 2.1: ゲートウェイのルーティングテーブル

トの転送経路を決定するものである。これは、すべてのホストに対する経路をルーティングテーブルに格納することは、ホストの増加により現実的でなくなるためである。

経路制御の例として、図 2.3 の構成においてネットワーク L に接続しているホスト A からネットワーク N に接続しているホスト B にパケットを転送する場合を示す。

1. ホスト A は、ホスト B のネットワークアドレスから自分と異なるネットワークに送らなければならないことを見つけ、自分で直接送れないので、パケットをゲートウェイ G に転送する。
2. ゲートウェイ G は、ホスト B のネットワークアドレスでルーティングテーブルを参照し、ゲートウェイ H に転送すれば良いことがわかるので、パケットをゲートウェイ H に送る。
3. ゲートウェイ H は、ルーティングテーブルを参照するとホスト B がネットワーク N に接続していることが得られるので、ホスト B にパケットを直接転送する。

### 2.1.3 名前空間

Internet ではドメインネームシステム (DNS: Domain Name System) [18][19] により、ネットワークの名前空間 (ネームスペース) を管理している。DNS におけるネームスペースは図 2.4 のような木構造をしている。

図 2.4 の木の各ノードをドメインと呼び、ドメインの名前はそのノードに到達するまでのパスにあるラベルを “.” で接続したものである。

あるドメインのホストをルートからのパスで示すことによりそのホストが接続しているドメインを識別することができる。図 2.4 の例では、ホスト uecgw は uecgw.cs.uec.ac.jp と表される。

ホスト名や IP アドレス等の情報への問い合わせに答えるものがネームサーバである。例えば、ホスト名をキーにしてネームサーバに問い合わせることで対応する IP アドレスを得ることができる。

DNS を UNIX の BSD 版に実装したものが BIND (Berkeley Internet Name Domain) [101] [102] である。BIND では、ネームサーバは named [103] というプログラムで実現され、ネームサーバに問い合わせを行なう時は、レゾルバ (resolver) ライブラリ [103] を使用する。

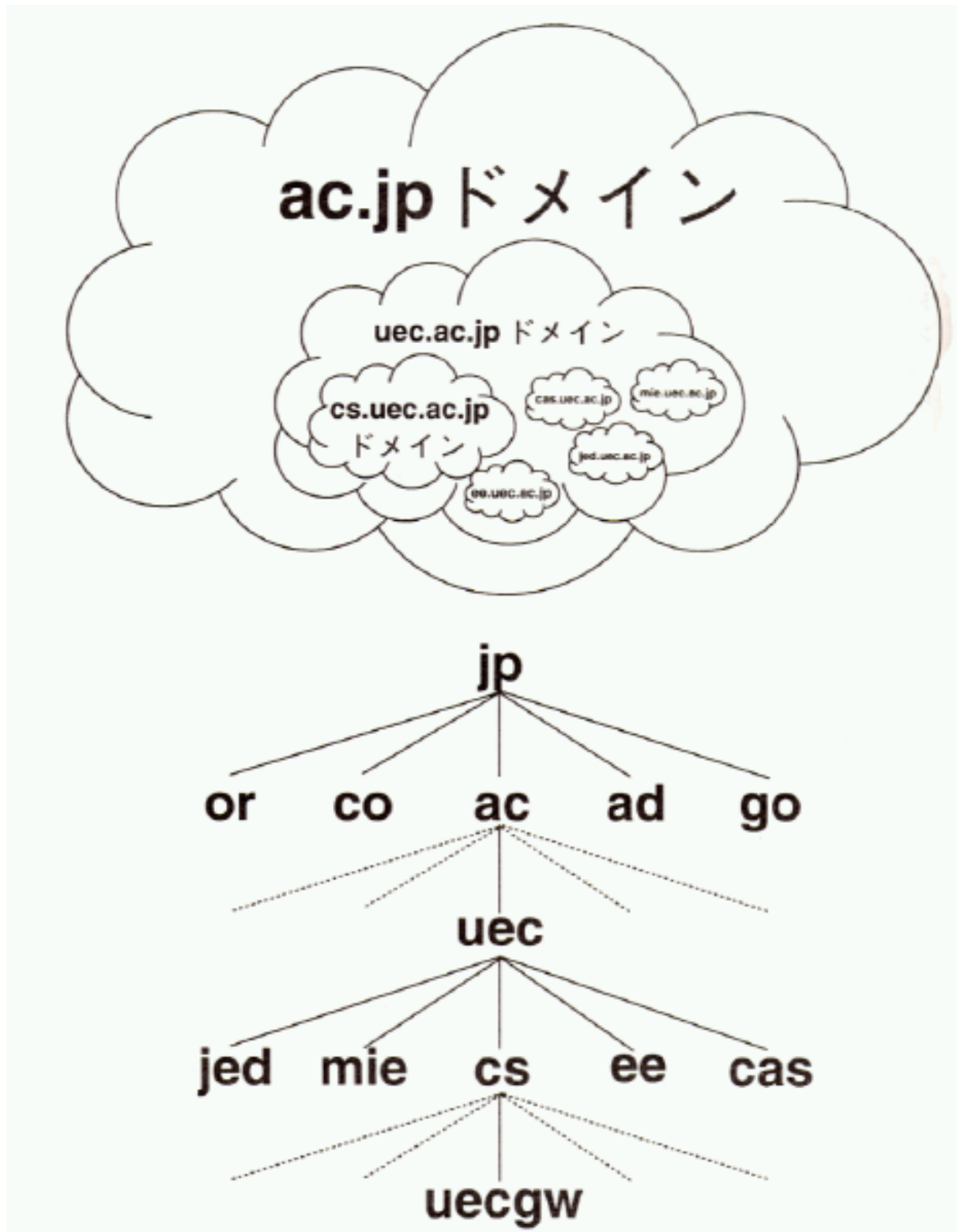


図 2.4: ドメインネームシステムの木構造

一つのデータベースによって管理されるドメインの集合をゾーンと呼ぶ。ゾーンの中には後で述べるプライマリマスタネームサーバが一つあり、ゾーンのデータを管理している。その他のネームサーバはいくつあっても構わない。

このゾーンはドメインと同様に階層構造持っている。自分のゾーンにはない情報についての問い合わせを受けたネームサーバは、自分が知っている他のゾーンのネームサーバに問い合わせを依頼する。どのゾーンのネームサーバもルートにあるネームサーバへ到達することができ、またルートにあるネームサーバからたどることができるようになっているので、この問い合わせに答えることができる。

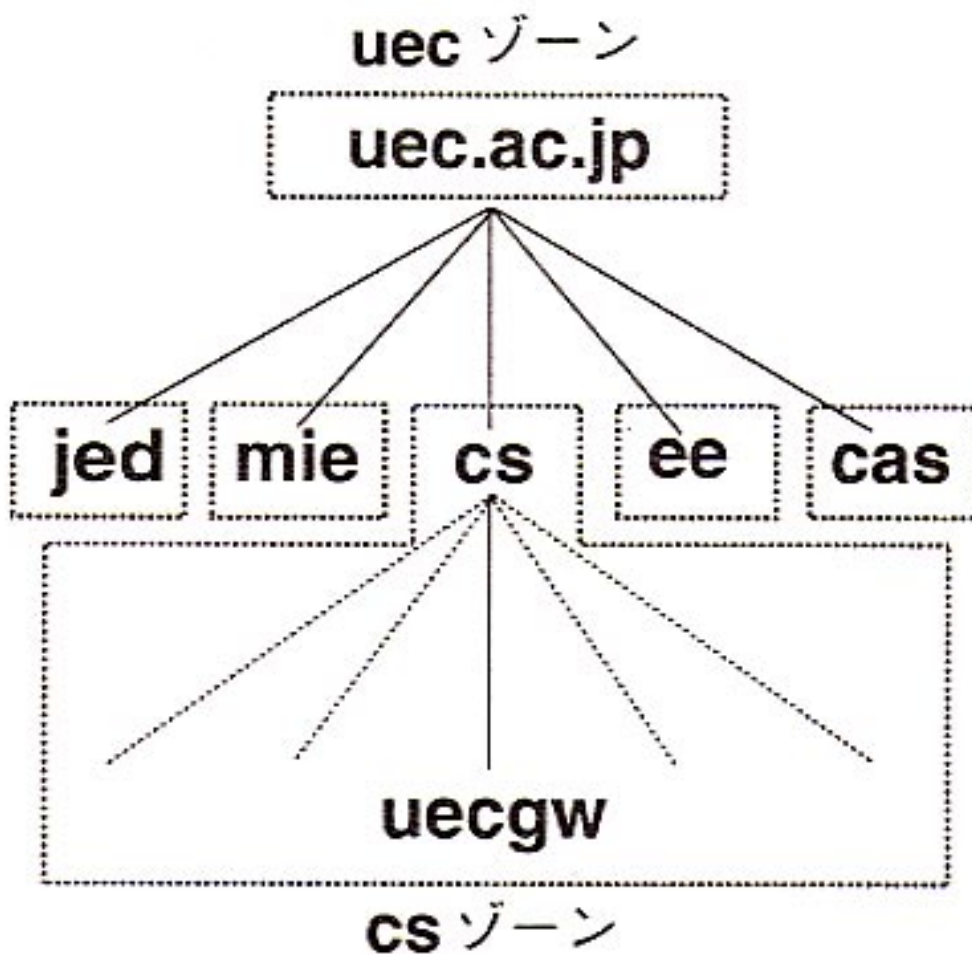


図 2.5: ゾーンの階層構造

図 2.5 の例では、cs ゾーンのネームサーバは uec ゾーンのネームサーバのアドレスを持っており、uec ゾーンのネームサーバは cs, jed, mie, ee, cas ゾーンのネームサーバのアドレスの他に、さらに上位層のネームサーバのアドレスを持っている。

BIND におけるネームサーバは次の 4 種類に分類できる。

### Primary Master Server

このサーバはネームサーバが管理するゾーンに対してオーソリティ(権威)を持つサーバで、起動する時にファイルからデータを読み込む。

### Secondary Master Server

セカンダリマスタサーバは、プライマリマスタサーバからオーソリティを委託される。このサーバは起動時にプライマリマスタサーバからそのゾーンのデータを受け取り管理する。また、プライマリマスタサーバのデータを一定時間毎に検査し、データが更新されていないか調べる。

### Caching-Only Server

キャッシングオンリサーバはデータをキャッシュするだけのサーバである。このサーバはどのゾーンに対してもオーソリティを持たないので、問い合わせを受けた時にはオーソリティを持つマスタサーバに聞き直す。

### Forwarding Server

フォワーディングサーバは、すべての問い合わせを指定されたサーバに転送するサーバである。このサーバの仕事は、他のネットワークにアクセスできないホストの、そのネットワークに対する問い合わせを中継することである。

すべてのネームサーバでは情報のキャッシュが行なわれる。これは、同じ問い合わせを繰り返し、ネットワークの負荷を上げることを避けるためである。キャッシュされた情報は一定時間経過すると削除される。この時間は、各データ毎に指定された生存時間(ttl: time to live)によって決定される。

## 2.1.4 Internet のまとめ

Internet の特徴をまとめると次のようになる。

- TCP/IP プロトコルを用いたパケット交換網である。
- ゲートウェイを介して複数のネットワークを接続している。
- IP アドレスはノードを識別するために用いられ、固定である。
- ネットワークアドレスは経路制御のために用いられる。
- ホスト名はルートからのパスを“.”で連結したものである。
- ネームサーバに問い合わせることでホスト名とアドレスの対応を検索する。

## 2.2 可搬ノードを接続する時の問題点

2.1節で説明したことから、なぜ現在の Internet が可搬ノードに対応していないかを述べる。

可搬ノードのネットワークアドレスは、普段マシンを接続しているネットワークを表している。この IP アドレスを持つ可搬ノードが他のネットワークへ移動しても IP アドレスは変化しない。このために、可搬ノードのネットワークアドレスと、移動先のネットワークのアドレスに矛盾が生じる。この矛盾により、可搬ノードは他のマシンへパケットを送ることはできても、受けとることができなくなる。なぜならば、他からこの可搬ノードに向けて送信されたパケットは、その IP アドレスのネット部が指し示す、普段可搬ノードが接続しているネットワークへ届いてしまうためである。当然そこにはマシンはないので、パケットは失われてしまう。

ここで IP アドレスが固定でないとしてみる。つまり可搬ノードの移動した先で、そのネットワークに合う IP アドレスを割り当てられたとする。この方法だと、確かにパケットの送受信は間違いなく実行できる。しかし、他のホストには可搬ノードの新しい IP アドレスがわからないので、他のホストからそのマシンと通信することはできなくなってしまう。

以上のような理由により、現在の Internet は可搬ノードに対応することができない。解決方法については次の章で議論する。

## 第 3 章

# 可搬ノードに対応した Internet の構築

2.2節で述べたように、現在の Internet では移動するノードをネットワークにうまく接続できない。これを解決するためには、ノードを識別する情報と、経路を制御する情報を別々に持たなければならない。これより、現在の Internet に対応する方法には、次の二つの方針が考えられる。

- IP アドレスのネット部を経路制御に使用しない。
- IP アドレスをノードの識別に使用しない。

はじめの方式を動的経路更新型とし、二つめの方式を動的アドレス更新型とする。それぞれについて、以下の節で詳しく述べる。

### 3.1 動的経路更新型

動的経路更新型の経路制御では、ネットワークアドレスを経路制御に使用せず、IP パケットの経路を動的に変化させ、可搬ノードに対応する。

このために新しい経路制御アルゴリズムを開発する必要がある。この経路制御アルゴリズムは、移動した可搬ノードの位置をデータベースに登録し、そのデータベースにしたがってパケットを転送することが基本になっている。

このデータベースをどのノードが持つかによって三つに分類できる。一つは可搬ノードが指定するホストが持つ方法であり、もう一つはネットワーク内のゲートウェイが持つ方法、最後は各ホストが持つ方法である。特定のホストが持つ方法として Point-to-Point 方式を取り上げ、ゲートウェイが持つ方式として専用ネットワーク方式を説明し、各ホストが持つ方式として仮想ネットワーク方式について述べる。

また、ネットワーク層 [5] では経路制御を行わず、データリンク層 [5] で行なう方式として、自動車電話方式についても簡単に述べる。

#### 3.1.1 Point-to-Point 方式

可搬ノードに特定のホストを対応させて移動を可能にする方法が、Point-to-Point 方式 [104] である。この特定のホストをホームホストと呼ぶ。ホームホストは移動しないノードで、可搬ノードの IP アドレスのネット部が示すネットワークに接続している。

可搬ノードは普段ホームホストがあるネットワークに接続している。この可搬ノードが移動して他のネットワークにある中継ノードと接続すると、(図 3.1) に示すようにパケットはホームホストを経由して転送される。

このために、可搬ノードは他のネットワークに接続すると、ホームホストに対して自分が接続した中継ノードのアドレスを送信する。ホームホストは受けとったアドレスを登録し、以後受けとった可搬ノードに対するパケットをその中継ノードに転送する。

他のノードが可搬ノードにパケットを送ると、そのパケットは始めに可搬ノードの IP アドレスから決まるネットワークに送られる。そのネットワークには IP アドレスの示す可搬ノードは接続していないので、ホームホストがそれを代わりに受けとり可搬ノードに転送する。

可搬ノードの IP アドレスは固定であるので、通常の経路制御ではホームホストから直接パケットを送ることはできない。これは可搬ノードの移動した先のネットワークアドレスと、可搬ノードのネットワークアドレスが異なるためである。このために、ホームホストから可搬ノードへのパケットの転送は、ソースルーティング [5] を使用して行なわれる。

ソースルーティングとは、パケットの中にあらかじめ通過すべき経路を記述しておく方法であり、途中のゲートウェイにおける経路制御を無効にすることができる。これにより、可搬ノードの矛盾した IP アドレスにもパケットを転送することができる。

可搬ノードから送信するパケットは、ホームホストを経由することなく直接送ることができる。これは、受信ノードのネットワークアドレスに対する経路制御情報をゲートウェイが持っているためである。

この方式は、ホームホストにおけるパケットの転送と、ホームホストに中継ノードのアドレスを登録する機構を実装することで実現できる。このアルゴリズムをうまく設計することにより、移動しながら、つまりパケットの経路が変化しながらでも通信を続けることができる。移動して中継ノードが変わった時には、新しい中継ノードのアドレスをホームホストに登録すれば良い。

また、途中のネットワークにあるマシンではソースルーティングに対応していれば良く、新しいプロトコルのサポートなどは必要ない。

欠点としては、可搬ノードへのパケットがすべてホームホストを通過するために、無駄なトラヒックが発生することである。例えば、普段日本で使用している可搬ノードをアメリカに持って行って使用したとすると、アメリカにあるノードからその可搬ノードに向けられたパケットは太平洋を 1 往復することになる。これは、限られた帯域しか持たない通信回線にとっては問題である。

### 3.1.2 専用ネットワーク方式

専用ネットワーク方式は、図 3.2 に示すように移動通信用に専用のネットワークを構成する方法である。

このネットワークは、アドレス的に一つのネットワークを構成するようになっていて、

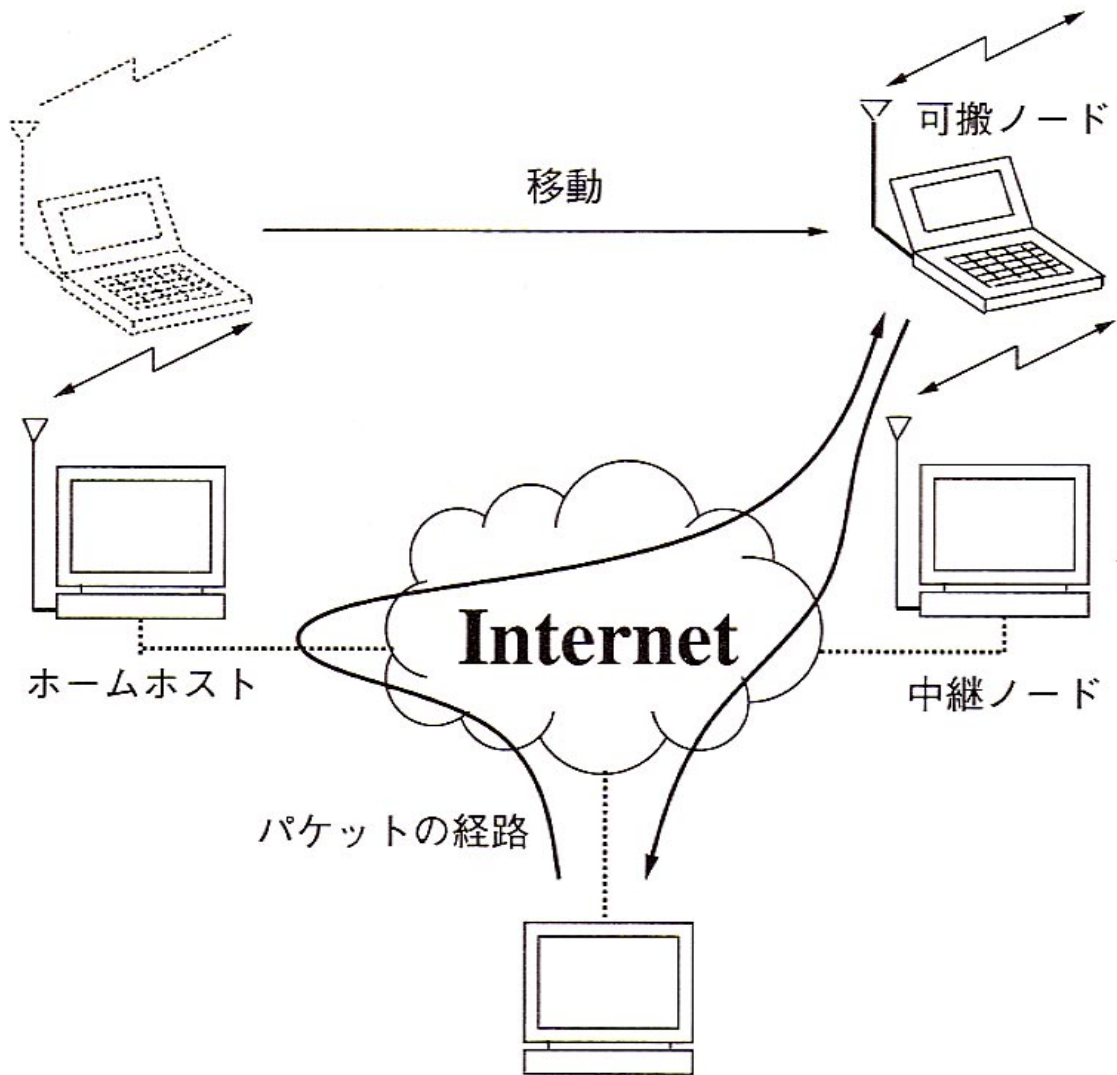


図 3.1: パケットを中継するホームホスト



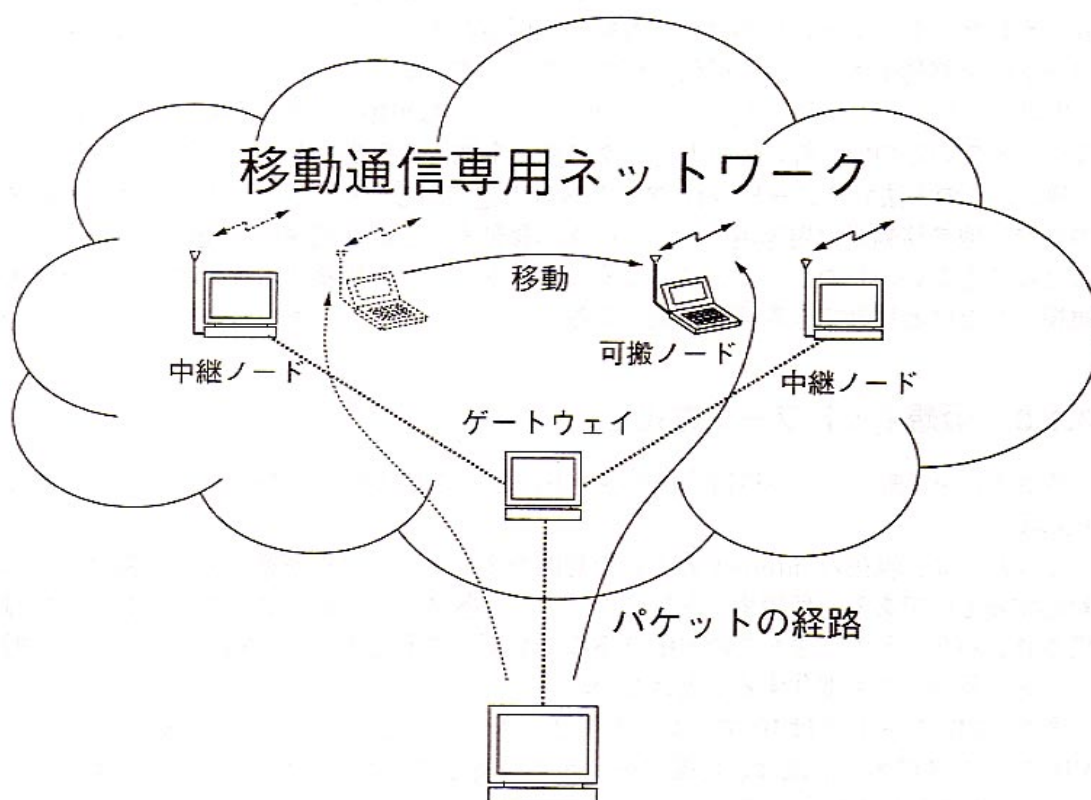


図 3.2: 専用ネットワーク方式

その中を移動する可搬ノードの IP アドレスは固定である。可搬ノードへのパケットは、このネットワークのゲートウェイを経由して転送される。したがって、他のネットワークのホストは、可搬ノードのネットワークアドレスから求められるゲートウェイに転送すれば、可搬ノードの位置について考えることなくパケットを送ることができる。

移動通信ネットワークの中では、可搬ノードに対応した経路制御を行わなければならない。このために、ネットワーク内の中継ノードは自分に接続している可搬ノードの情報をお互いに交換し、どこに誰が接続しているというデータベースを構築する。ゲートウェイはこのデータベースをもとにしてネットワーク内の経路制御を行なう。

可搬ノードが通信中に移動し、他の中継局と接続することになっても、その情報がすみやかにゲートウェイに伝われば、接続が切断されることはない。つまり、このネットワーク内を移動中はいつでも通信を行なうことができる。

専用ネットワーク方式では、外のネットワークからは可搬ノードが移動することは見えなくなっているので、外のネットワークに変更を加える必要がない点で優れている。

逆に、この方法ではノードの IP アドレスは固定であるために、ネットワークアドレスの異なる他の移動通信専用ネットワークへ移動すると経路制御の問題により使用することはできない。したがって、このようなネットワークを構築する場合には、かなり大規模にしないと移動できる範囲が狭くなる。

### 3.1.3 仮想ネットワーク方式

各ホストが可搬ノードの移動情報を持つ方法として仮想ネットワーク方式 [105] があげられる。

この方式は、現在の Internet の上に仮想的なネットワークを構築して、可搬ノードに対応するものである。仮想ネットワークの上では各ノードは仮想 IP(VIP) アドレスで識別され、パケットの転送もこの VIP アドレスに対して行なわれる。VIP アドレスは可搬ノードが移動しても変化することはない。

また、VIP アドレスは IP アドレスと同様にネット部とホスト部から構成されている。VIP アドレスのネット部は、可搬ノードが普段属しているネットワーク(ホームネットワーク)のアドレスになっている。

仮想ネットワークの下になった物理ネット(Internet)では、IP アドレスを用いて実際のパケットを転送する。この IP アドレスのホスト部は、VIP アドレスのホスト部と同じ値を持つ。可搬ノードが移動すると、ノードの IP アドレスのネットワーク部は、移動先のネットワークアドレスを持つように変えられる。この VIP アドレスと IP アドレスの変換情報は、可搬ノードがホームネットワークにパケットを転送することで、ホームネットワークのホストに登録される。

VIP アドレスと IP アドレスの対応づけは各ノードが自分の持つ情報によって行なう。この情報は自分のネットワークを監視して、通過するパケットのアドレスをキャッシュすることによって得ることができる。

可搬ノードにパケットを送る時には、この VIP アドレスから IP アドレスの変換が必

要になる。IP アドレスがわからない時は、IP アドレスを VIP アドレスと同じものにして送信する。この IP アドレスは、その VIP アドレスと IP アドレスの変換情報を持っているノードを通過する時に、正しい IP アドレスに変換される。最悪でも、自分のホームネットワークではこの変換が行なえるので、パケットは、正しく転送される。

また、一度パケットが通過すると、変換情報がキャッシュされるため、次からは無駄な経路を避けることができる。図 3.3 の例は、ネットワーク A には可搬ノードが移動したことが知らされていないために、可搬ノードへのパケットをホームネットワークに転送するが、ホームネットワークから可搬ノードへのパケットが通過するとこの変換情報をキャッシュするため次からは直接転送するようになることを示している。

この方式でも、可搬ノードは移動しながらネットワークを使用することが可能である。可搬ノードが移動先から変換情報をホームネットワークに送ることで、パケットの送られるべき IP アドレスが得られるためである。

仮想ネットワーク方式のパケットは途中では IP アドレスを持ったパケットであるため、パケットが通過するすべてのホストで VIP を実装する必要はない。ただし、VIP が実装されていないと変換情報をキャッシュできないので、Point-to-Point 方式と同様に無駄な経路が発生する。

また、古いキャッシュのデータが残っていたために VIP アドレスと IP アドレスの変換を間違っに行なうことがある。図 3.4 の例では、ネットワーク B が古いキャッシュデータを持ったままだったので、パケットはネットワーク C に送られてしまう。これを解決するには、キャッシュの有効時間や、変換情報の伝搬などについて検討しなければならない。

### 3.1.4 自動車電話方式

自動車電話方式はパケットを転送する媒体として自動車電話(移動電話)を使用する方式である。可搬ノードが移動する際のパケットの経路変更は、すべて自動車電話側で行なう。このため、既存の Internet に何の変更を加えることもなく可搬ノードに対応できる。

欠点としては無駄な経路が生じることと、料金が高いことである。

可搬ノードが自動車電話を用いて接続するネットワークはいつも同じものであるので、移動先の近くにあるノードへパケットを転送する時にも遠まわりをしなければならない。

料金の問題は、デジタル方式になり課金体系がパケット単位にならなければコストがかかり過ぎて使いものにならない。しかし非常の場合などのバックアップとしての使用方法が考えられる。

この方式では、自動車電話ネットワークが整備されていないところでは使用できないという欠点もある。

## 3.2 動的アドレス更新型

つぎに、移動することにより IP アドレスを変化させ、パケットの経路制御を行なう動的アドレス更新型について述べる。

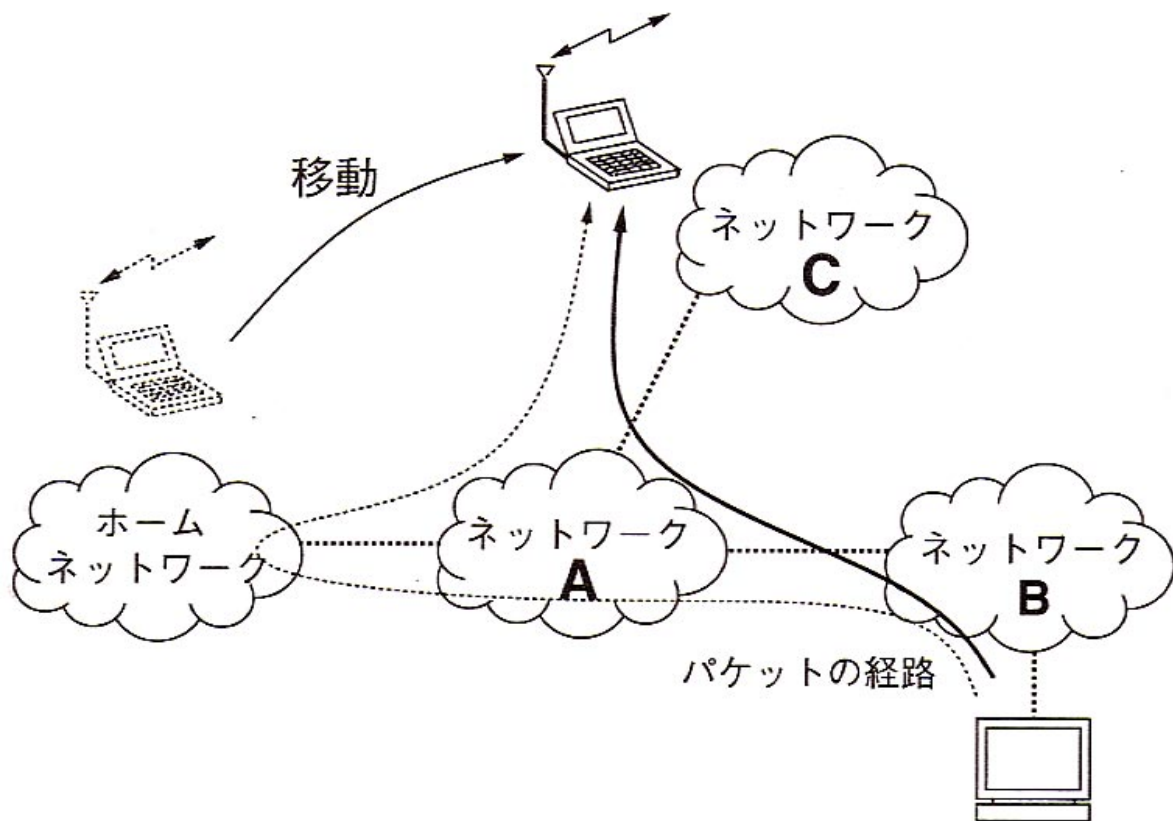


図 3.3: 仮想ネットワーク方式における変換情報のキャッシュによる経路の変化

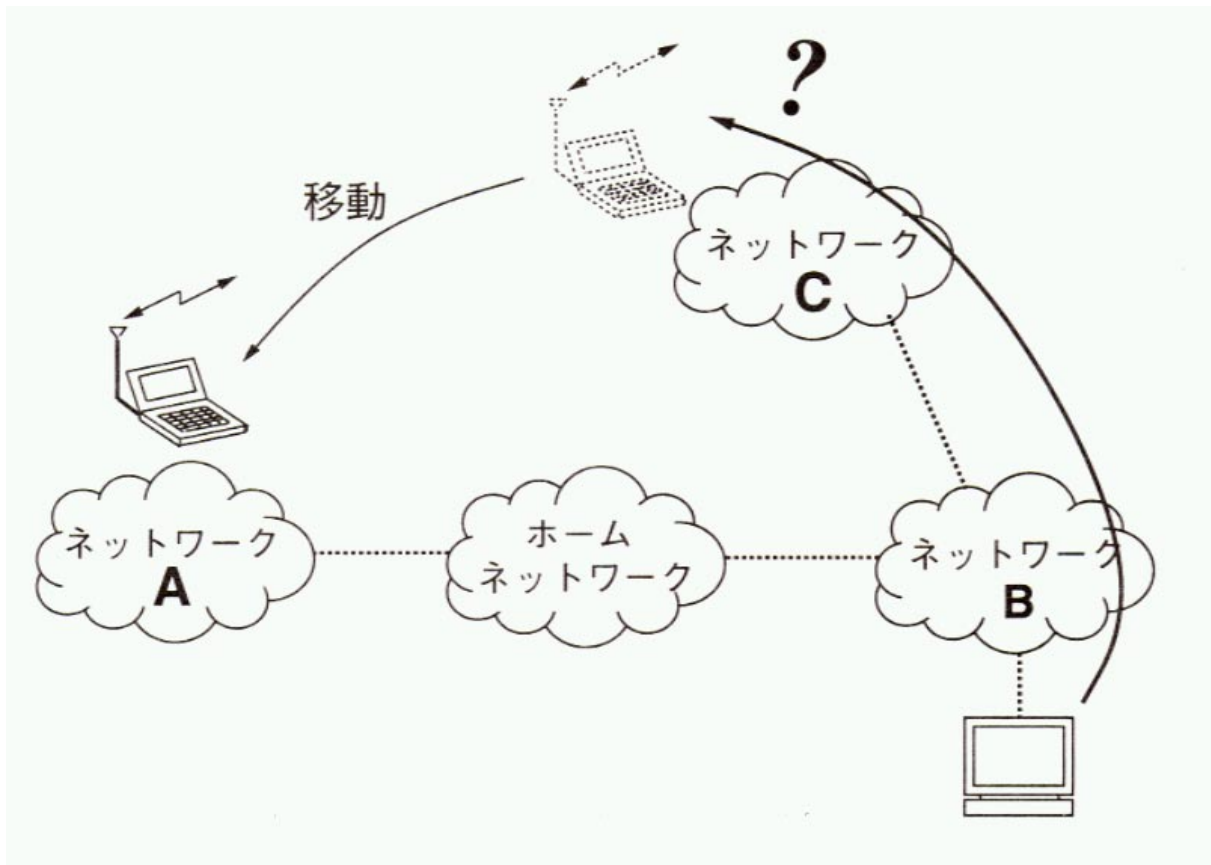


図 3.4: 古いキャッシュ情報による誤った経路制御

動的アドレス更新型では、割り当てられた可搬ノードの IP アドレスのネット部と、移動先のネットワークのアドレスの整合性がとれている。このため、パケットの経路制御は Internet の通常の方法である、アドレスのネット部を用いたものを使用できる。

この方式の中から、IP アドレス割り当て方式をとりあげ説明する。

### 3.2.1 IP アドレス割り当て方式

IP アドレス割り当て方式は移動先のネットワークから IP アドレスを割り当ててもらおう方式である。割り当てられた IP アドレスのネット部はそのネットワークのアドレスと整合性がとれているので、経路制御方法を変更する必要がない。

IP アドレスが変化してしまうために、可搬ノードを識別する情報が新たに必要となる。この識別情報には、可搬ノードのホスト名を使用することにする。また、ホスト名と IP アドレスの対応づけはネームサーバで行なうことにする。

あるホストが可搬ノードにパケットを送信する時には、ネームサーバに可搬ノードのホスト名を使って問い合わせ、IP アドレスを取得する。このために、可搬ノードのホスト名と割り当てられた IP アドレスの対を、ネームサーバに登録する必要がある。

この方式では、ネットワークの経路制御情報にしたがって直接パケットが転送されるため、Point-to-Point 方式のような無駄な経路は発生しない。

IP アドレス割り当て方式の欠点は、IP アドレスの変化を伴う移動をしながらの使用はできないことである。これは、ネットワークを使用するアプリケーションでは、一度 IP アドレスをネームサーバから取得した後は再びネームサーバに問い合わせることはしないためである。

この方式については 4 章で詳しく述べる。

## 3.3 設計方式の選択

この章で述べた方式の中から、次のような方針に合うものを選択し実装する。

- Internet プロトコルをなるべく変更しない。
- 実験環境が実現できる。
- 実装が容易。
- 実用的。

この章で述べた方式についてそれぞれ検討してみると以下のようなになる。

### Point-to-Point 方式

Internet プロトコルを変更しない点と、実験環境の構築、実装については問題はないがパケットの経路に無駄がある点で実用的でないと思われる。

### 専用ネットワーク方式

この方式は現状の Internet プロトコルを変更する必要がない点は良いが、実験環境を実現するという点で問題がある。つまり、実際に複数のマシンを使用して動的な経路の更新を行なってみて、動作を確認する必要があるがこのような実験環境を整備することは時間とコストの問題で容易にはできない。

また、この専用ネットワークを構築できなければ実用にはならないことも問題である。

### 仮想ネットワーク方式

はじめから新しいネットワークを構築する時には良いが、Internet のプロトコルを大幅に変更しなければならないため、現状のネットワークに適応させるにはむりがある。

途中にあるネットワークでは実装されていなくても良いので実験環境は容易に実現できそうであるが、変換情報のキャッシュが行なわれていないと無駄なトラフィックが発生してしまう。

キャッシュの伝搬についても考察が必要ですぐに実装できる状況にはない。

### IP アドレス割り当て方式

現在の Internet プロトコルに変更を加えることなく実現でき、実装も IP アドレスの割り当て部とネームサーバへの登録部だけである。また実験環境も、最小限 IP アドレス割り当てノードと可搬ノードだけで済む。他のホストに変更を加える必要がないので、現在のネットワークに接続して使用する時に障害とならず実用の面でも優れている。

以上より、IP アドレス割り当て方式を設計し実装する。

## 第 4 章

# IP アドレス割り当て方式の設計

この章では動的アドレス更新型の IP アドレス割り当て方式の設計について述べる。はじめにこの方式の構成要素を説明し、次に接続、切断等のプロトコルについて述べる。

### 4.1 IP アドレス割り当て方式の構成

IP アドレス割り当て方式は、次の 3 つの部分から構成される。

- 可搬ノード

可搬ノードは、持ち運んで使用されるワークステーションである。この可搬ノードはホスト名を持っているが、IP アドレスは割り当てられていない。このため、ネットワークに接続する時にはアドレス割り当てサーバから IP アドレスを割り当ててもらう必要がある。可搬ノードとアドレス割り当てサーバ (ゲートウェイ) の間は point-to-point で接続される。

- アドレス割り当てサーバ

アドレス割り当てサーバは、可搬ノードからの要求により IP アドレスの割り当てを行ない、可搬ノードをネットワークに接続する役割を持っている。このために、アドレス割り当てサーバは割り当て用にいくつかの IP アドレスを確保してある。可搬ノードに割り当てた IP アドレスとホスト名の対応をネームサーバに登録することもアドレス割り当てサーバで行なわれる。

- ネームサーバ

ネームサーバは、可搬ノードの IP アドレスとホスト名を対応づけるために使用される。クライアントはネームサーバに問い合わせることで可搬ノードの IP アドレスを得ることができる。

以下の節で IP アドレスの割り当てと解放、ネームサーバへの登録と解放について述べる。



## 4.2 IP アドレスの割り当て

アドレス割り当てサーバは可搬ノードからの要求を受けて、IP アドレスを供給する。この時の可搬ノードに対する IP アドレスの割り当て手順は図 4.1 のようになる。

図 4.1 の例は、ドメイン `cs.uec.ac.jp` にある可搬ノード `mobile1` が、他のネットワーク (ネットワークアドレス: `192.14.64.0`) に移動した時の様子である。

この時、アドレス割り当てサーバは確保してある IP アドレスの中から `192.14.64.89` を可搬ノードに割り当てることにする。この手順は次のようになる。

1. 可搬ノードはアドレス割り当てサーバに対して、自分のホスト名 `mobile1.cs.uec.ac.jp` とネームサーバのアドレス `130.153.28.161` を含む、アドレス要求を出す。
2. アドレス要求を受けとったアドレス割り当てサーバは、空いている IP アドレス `192.14.64.89` を割り当て、可搬ノードにアドレス割り当て応答を返す。
3. アドレス割り当てサーバは IP アドレスとホスト名の組を持つ、登録要求をネームサーバに送る。
4. 登録要求を受けとったネームサーバは、可搬ノードのホスト名と IP アドレスの組をデータベースに登録する。

可搬ノードとアドレス割り当てサーバは、割り当てた IP アドレスによってネットワークの設定を行わなければならない。ネットワークの設定には、インタフェースアドレスの設定と経路制御情報の設定がある。

可搬ノードではすべてのパケットをアドレス割り当てサーバに送るようにし、アドレス割り当てサーバでは割り当てた IP アドレスへのパケットだけを可搬ノードに転送する。

## 4.3 IP アドレスの解放

次に IP アドレス解放手順について述べる。

IP アドレスの解放は、可搬ノードからの切断要求を受信するか、一定期間可搬ノードからのパケットを受信しなくなった時に行なう。なぜなら、可搬ノードは電源が突然落ちたり、電波が届かなくなる等の原因により、自分で IP アドレスを解放できなくなる可能性があるためである。

IP アドレス解放の手順を図 4.2 に示す。

図 4.2 では、可搬ノードがいつの間にかいなくなってしまった場合である。この時の IP アドレス解放の手順を示す。

1. アドレス割り当てサーバは可搬ノードからのパケットを監視し、一定時間パケットを受信しない時には可搬ノードに対してエコー要求を出す。

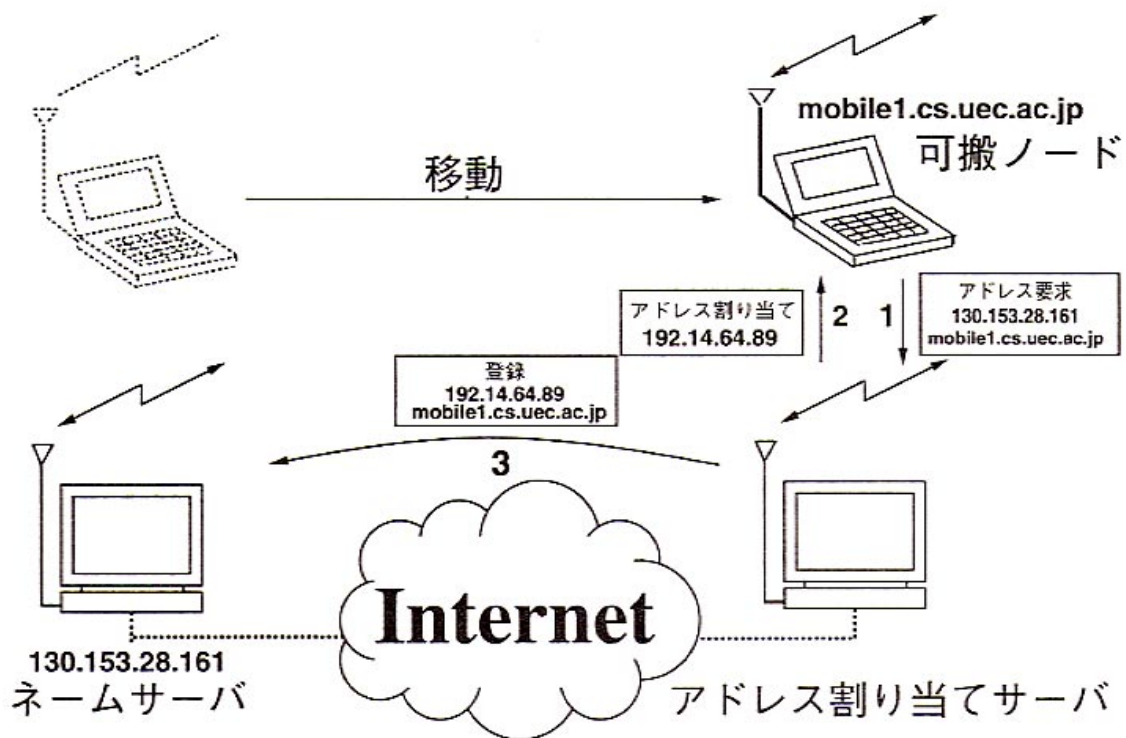


図 4.1: IP アドレス登録プロトコル

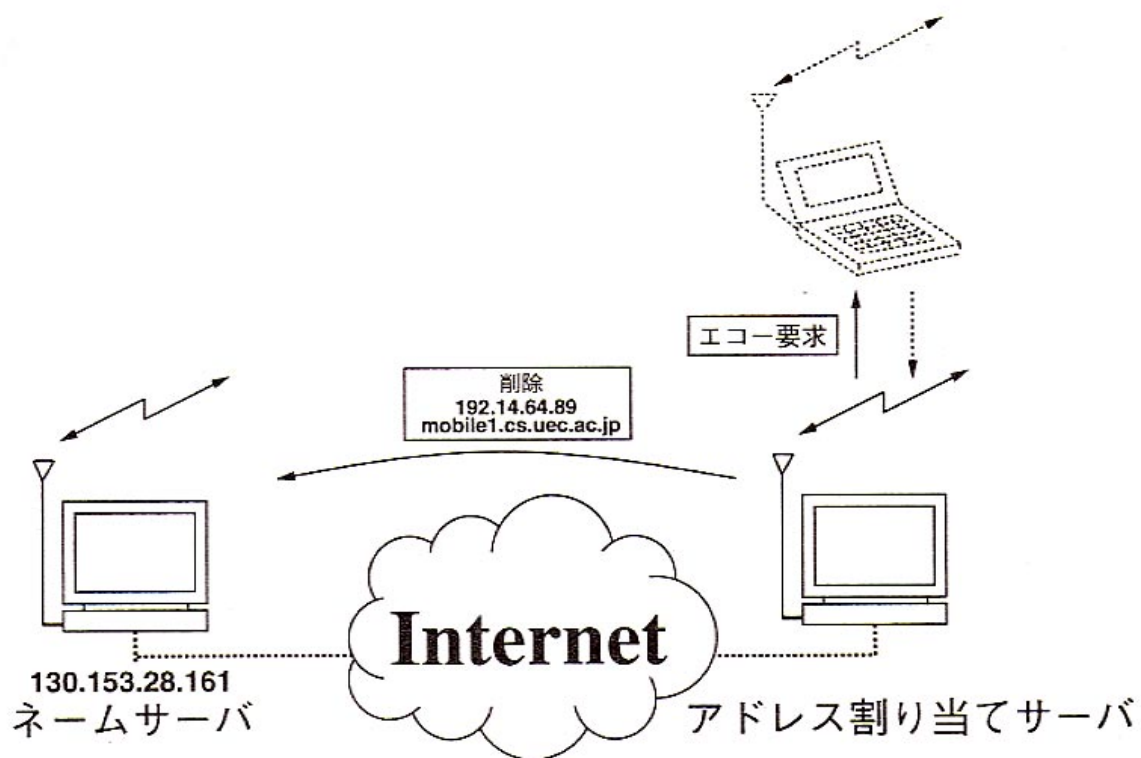


図 4.2: IP アドレスの解放

2. さらに一定時間待って可搬ノードからの反応がない場合には可搬ノードの接続を解放し IP アドレスを解放する。
3. アドレス割り当てサーバは可搬ノードがいなくなったことをネームサーバに通知する。
4. ネームサーバはデータベースからホスト名と IP アドレスの対を削除する。

可搬ノードは自発的に IP アドレスを解放することもできる。この時は、アドレス割り当てサーバに対して切断要求を送る。アドレス割り当てサーバが切断要求を受けとった時には済みやかに接続を解放する。

アドレス割り当てサーバは、回収した IP アドレスを一定時間他の可搬ノードに割り当てないようにしなければならない。これは、それまで使用していた可搬ノードに対するパケットがしばらくの間到着する可能性があるためである。

このために IP アドレスサーバでは、一つの回線に対して複数の IP アドレスを用意しておくことが望ましい。

## 4.4 可搬ノードとの通信

可搬ノードと通信する時には図 4.3 のように、可搬ノードのホスト名 `mobile1.cs.uec.ac.jp` を使ってネームサーバから IP アドレス `192.14.64.89` を得る必要がある。

通信を行なうアプリケーションは以後この IP アドレスを使用してパケットの転送を行なう。ネームサーバに問い合わせる時の可搬ノードのホスト名は、可搬ノードの情報があるドメインを特定する必要があるため長いホスト名を使用しなければならない。

## 4.5 ネームサーバ

ネームサーバには DNS を UNIX の BSD 版に実装した BIND を使用する。BIND のネームサーバには、2.1.3 節で述べたように他のネームサーバに対して行なった問い合わせの結果をキャッシュしておく機能がある。このキャッシュの情報は一定時間で消滅する。しかし、キャッシュの情報が消滅するまでの間に、可搬ノードが移動して他のネットワークに接続すると、仮想ネットワークのところで述べたようなキャッシュによる間違っただ変換が行なわれる。(図 3.4 参照)

これを回避するためには、キャッシュしているホストを把握してそこに対して変更の通知をするか、キャッシュを無効にするしかない。

どのホストがデータをキャッシュしているかを記録しておくことは現実的ではない。これは、Internet に接続しているホストの数を考えれば明らかである。したがって、ここでは可搬ノードの情報についてはキャッシュしないという対応をとることにする。

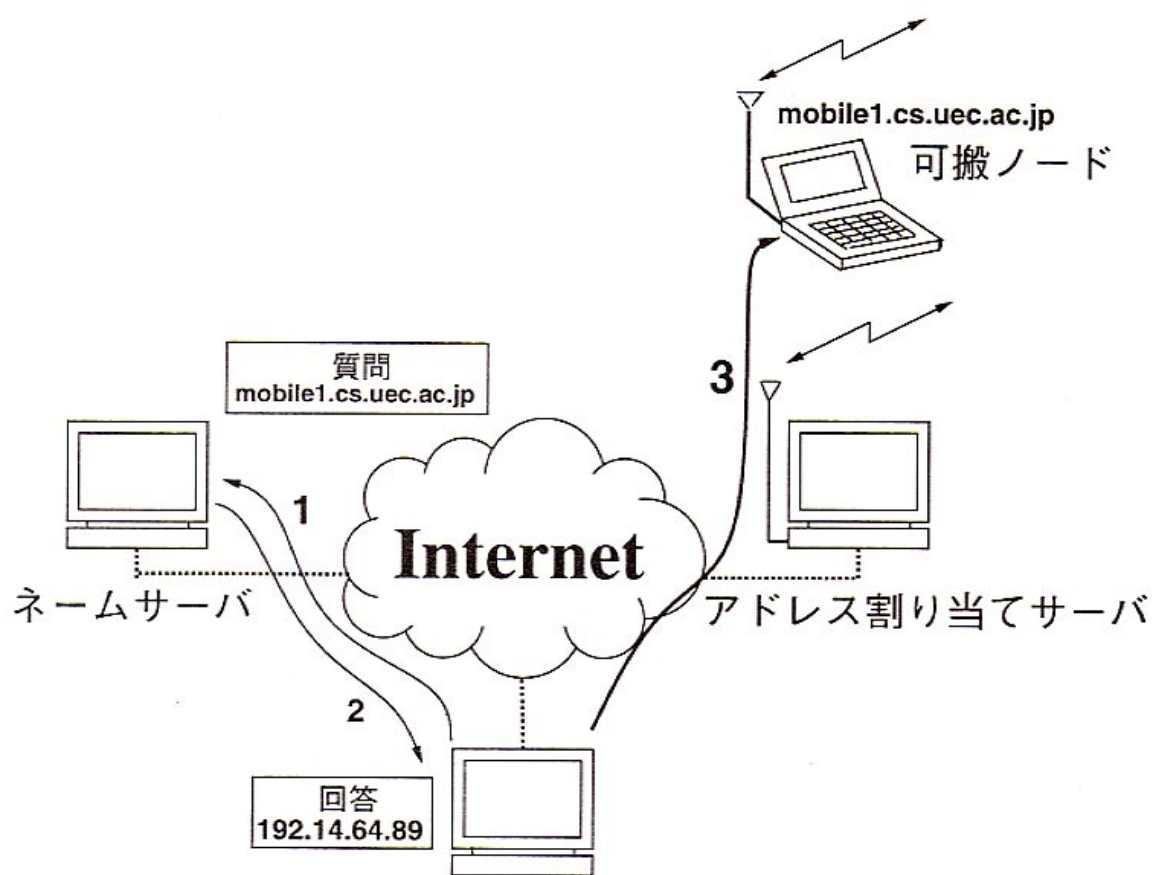


図 4.3: 可搬ノードの IP アドレスの取得

## 4.6 セキュリティ

可搬ノードが IP アドレスを受けとる時と、ネームサーバに IP アドレスとホスト名を登録する時には認証機構がなければならない。これは、他のホストのふりをしてネットワークに侵入することや、ネームサーバのデータベースを破壊することを防ぐためである。

この認証機構においては、盗聴を防ぐためにデータを暗号化する必要がある。しかし、暗号化だけではデータを録音し再び転送することにより情報の混乱を引き起こされる可能性がある。

高度な認証機構を導入することは難しいので、今回は可搬ノードとアドレス割り当てサーバの間に認証機構を導入することに留まった。これには、UNIX システムのパスワードと同じ機構を使用している。

## 第 5 章

# IP アドレス割り当て方式の実装

この章では、4章で述べたプロトコルの設計にしたがった実装について述べる。

### 5.1 アドレス割り当てサーバ

アドレス割り当てサーバには次の三つの機能が必要である。

- 可搬ノードと Internet の接続。
- IP アドレスの割り当て。
- ネームサーバへの登録と削除。

以上のような機能を持ったサーバプログラム `slserver` を実装した。それぞれの機能について以下で述べる。

#### 5.1.1 可搬ノードと Internet の接続

アドレス割り当てサーバと可搬ノードの間は point-to-point で接続する。この間のパケットの転送には SLIP[9] で接続する。SLIP は IP パケットをそのまま送り特別なヘッダはつけない。

SLIP では、END(8 進数で 0300) と ESC(8 進数で 0333) は特別なデータバイトとして取り扱われる。END は IP パケットの終了を示し、ESC は次のバイトともに 1 バイトのデータを示す。パケットを転送する時に END を見つけると ESC と 8 進数で 0334 の 2 バイトのコードに変換し、ESC を見つけると ESC と 8 進数で 0335 の 2 バイトのコードに変換される。

IP アドレスを割り当てる時には、可搬ノードには IP アドレスがない。このために通常の IP パケットを送ることはできない。そこで、SLIP に RAW データ転送プロトコルを追加した。パケットの形式を図 5.1 に示す。

RAW パケットと SLIP のパケットを区別するために、RAW パケットの先頭には ESC RAWDATA の 2 バイトが追加される。ESC は 8 進数で 0333、RAWDATA は 0345 である。

各フィールドの内容は次の通りである。

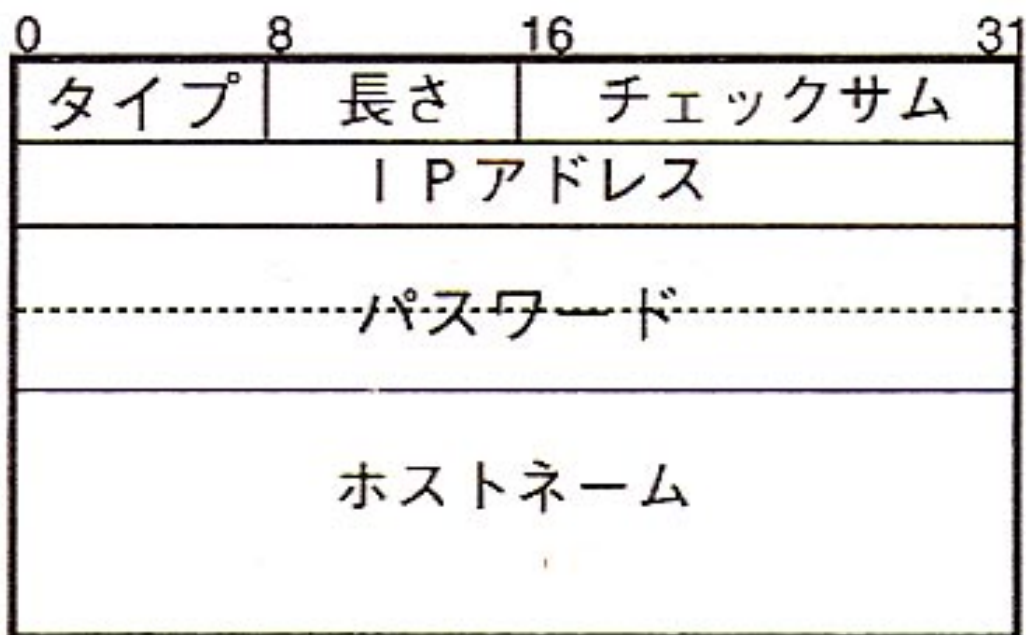


図 5.1: SLIP の RAW データパケット



## タイプ

タイプフィールドは RAW パケットの種類を示す。RAW パケットには表 5.1 に示すようなタイプがある。

8 進 名前	意味
0000 REQ_ASSIGN_ADDR	アドレス割り当て要求
0100 ACK_ASSIGN_ADDR	アドレス割り当て応答
0010 REQ_ECHO	エコー要求
0110 ACK_ECHO	エコー応答
0300 DISCONNECT	切断
0200 NACK_NO_ADDR	否定応答-割り当てられるアドレスがない
0210 NACK_CHECKSUM	否定応答-チェックサムがあわない。
0220 NACK_PASSWD	否定応答-パスワードが異なる。

表 5.1: RAW パケットのタイプ

タイプが要求であるパケットを受信した時には、対応する応答パケットを返す。エコー要求はエコー応答を返すように指示するもので、可搬ノードとの接続が保持されているかどうかを調べるために使用される。切断は可搬ノード、アドレス割り当てサーバの双方から送信することができる。

## 長さ

長さフィールドは RAW パケットの全長をバイト数で示したものである。このフィールドの長さは 8 ビットであるので、RAW パケットの最長は 255 バイトである。

## チェックサム

チェックサムフィールドは、RAW パケットの中のチェックサムフィールドを除く部分の、バイト単位の合計値である。

## IP アドレス

IP アドレスフィールドはパケットのタイプによってアドレスの意味が異なる。アドレス割り当て要求タイプの時には可搬ノードの情報を登録するネームサーバの IP アドレスを示し、アドレス割り当て応答タイプの時には割り当てられた IP アドレスを示している。その他のタイプの時には意味を持たない。

## パスワード

パスワードフィールドはアドレス割り当て要求タイプのパケットの時にのみ使用され、可搬ノードの認証を行なうために用いられる。ここには、UNIX システムで用

```
count = 0;
while(1) {
    一定時間待つ;
    if (可搬ノードからのパケットがある)
        count = 0;
    else {
        count++;
        if (count > 2) {
            接続を切断し IP アドレスを回収する;
            break;
        }
        else
            エコー要求タイプの RAW パケットを送る;
    }
}
```

表 5.2: 可搬ノードとの回線テストアルゴリズム

いられている crypt 関数 [103] により暗号化された文字列が入っている。crypt にはホスト名の頭 2 文字とパスワードを与えている。

### ホスト名

ホスト名フィールドには、可搬ノードのホスト名がはいる。このホスト名は、可搬ノードが普段属しているドメインを示すために長いホスト名でなければならない。ホスト名の最後には 0x00 が付け加えられている。

つぎに、アドレス割り当てサーバは可搬ノードとの接続が保持されていることを確認しなければならない。これは、可搬ノードとの回線が突然切断されてしまう可能性があるためである。このために、アドレス割り当てサーバは可搬ノードとの回線を一定時間毎に調べる。このアルゴリズムを表 5.2 に示す。

今回の実装では 5 分毎に調べるようにしてある。

### 5.1.2 IP アドレスの割り当て

アドレス割り当てサーバでは IP アドレスを割り当てる時には、もっとも最近使用されなかったアドレスを割り当てる。これは以前使われた他の可搬ノードへのパケットが受けとられることを避けるためである。

割り当てられる IP アドレスの一覧はファイル/etc/portable に次のようなフォーマットで格納する。

```
192.14.64.87
192.14.64.88
192.14.64.89    mobile1.cs.uec.ac.jp
192.14.64.90
```

第二カラムにホスト名を書くことによって、その IP アドレスは示されたホストに優先的に割り当てられる。ただし、使用できるアドレスが他にない時には他のホストに対しても割り当てられる。また、すでに使用されている場合には明示されたホストであっても他の IP アドレスが割り当てられる。

### 5.1.3 ネームサーバへの登録と削除

ネームサーバのデータベースに登録する時には、ソケット (socket)[103] を使用してネームサーバ更新デーモン (常駐プログラム) `ns-update` と通信する。この時使われるポート番号は 8932 である。

登録と削除は次のようなコマンドで行なわれる。

```
register hostname ipaddress
delete hostname ipaddress
```

`register` は登録コマンドで `delete` は削除コマンドである。どちらもホスト名と IP アドレスが引数として必要である。

## 5.2 可搬ノード

可搬ノードに必要な機能は、アドレス割り当てサーバとの接続である。5.1.1節で述べたように可搬ノードとアドレス割り当てサーバの間は `point-to-point` で接続する。このために用いる SLIP と、RAW パケットについては 5.1.1節で述べた通りである。

アドレス割り当てサーバと接続する機能を持つプログラム `slclient` を実装した。このプログラムを実行すると、アドレス割り当てサーバに IP アドレスを要求し、その次に割り当てられたアドレスでネットワークの設定を行なう。

## 5.3 ネームサーバ

ネームサーバは、可搬ノードの IP アドレスを知らせるために、可搬ノードと同じドメイン (`cs.uec.ac.jp`) のデータベースを持っていなければならない。

ネームサーバのデータベースを動的に変化させるためにはネームサーバ本体に変更を加える必要がある。しかし、今回の実装ではネームサーバに手を入れず、データファイルを変更し、ネームサーバに読み込ませるデーモン `ns-update` を作製した。

ネームサーバには、ホスト名から IP アドレスを引くタイプと、IP アドレスからホスト名を引くタイプがある。通常は二つのタイプの動作を一つのネームサーバが行なうが、IP アドレスの示すネットワークとホスト名の示すドメインが異なるので二つに分かれる。

アドレスからホスト名を引く時のネームサーバがどのマシンにあるかをアドレス割り当てサーバは知っていなければならない。逆に引くネームサーバのアドレスは、可搬ノードから伝えられる。

## 5.4 可搬ノードへのパケットの転送

可搬ノードに割り当てられる IP アドレスは独立したネットワーク、またはサブネットにせず、アドレス割り当てサーバがあるネットワークと同じアドレスを用いる。このために、可搬ノードが割り当てられる IP アドレスのネット部はアドレス割り当てサーバのあるネットワークと同じでなければならない。

このようにするのは、可搬ノードに対する経路情報が広まることを防ぐためである。可搬ノードがない時でも経路情報が他のマシンにあるのはおかしい。

したがって他のネットワークから可搬ノードにパケットを転送する時には、可搬ノードの IP アドレスのネット部が示すネットワークに送れば良い。

アドレス割り当てサーバは自分の接続しているネットワークを監視して、可搬ノード向けのパケットを取り出し転送する必要がある。この動作は、アドレス割り当てサーバ上で proxyarp[5] を動かすことで実現できる。

## 第 6 章

### 評価

#### 6.1 環境の構築

5章で述べた実装にしたがって、実験環境を構築した。構成要素を以下に示す。

##### 可搬ノード

可搬ノードにはパーソナルコンピュータとワークステーションを使用した。

パソコンには PC-9801VX(80286 10MHz) を用い、TCP/IP プロトコルをサポートするために KA9Q[106] パッケージを使用した。このパッケージ上で使用できる、アプリケーションには telnet[103] と ftp[103] がある。

またワークステーションには SPARC LT(SunOS 4.0.3) を使用した。この宿主には 5.1.1節で述べた拡張 SLIP を実装した。

可搬ノードは自分の IP アドレスが変化するために、`/etc/hosts` ファイルに自分のアドレスを書きおくことができない。つまり、可搬ノードにおいても自分の IP アドレスを得るためにネームサーバにアクセスする必要がある。

SunOS ではネームサーバに問い合わせる機構を NIS[103] を用いて実現している。クライアントからの問い合わせを受けとった時に、自分の NIS データベースにそのデータがない場合、NIS サーバはネームサーバに問い合わせ、答を得る。

このために可搬ノードから NIS サーバにアクセスできる環境を実現しなければならない。しかし、NIS のプロトコルはイーサネットのブロードキャストを使用するために、可搬ノードが他のホストの NIS サーバに接続することはできない。このために NIS サーバも可搬ノードで動かさなければならない。

実際に NIS を動かす時には、次のような順序で行なう。

```
ypserv          NIS サーバの起動。
ypbind -ypset   NIS クライアントをサーバを指定できるモードで起動。
ypset 127.0.0.1 NIS サーバを指定する。
```

NIS のサーバを指定する時には自分の localhost を IP アドレスで指定しなければならない。これは、まだ ypbind がサーバに接続できていないので、ホスト名を示しても IP アドレスに変換理解できないためである。

### アドレス割り当てサーバ

アドレス割り当てサーバには SUN3/60(SunOS 4.1) を使用し、5.1.1節で述べた拡張 SLIP を実装した。

### ネームサーバ

今回の実験環境では、ネームサーバをアドレス割り当てサーバと同じホストで動作させた。これは可搬ノードが移動していない状況である。

ネームサーバには BIND のネームサーバをインストールした。また、ネームサーバ更新デーモンも同じホストに実装した。

### 通信デバイス

可搬ノードとアドレス割り当てサーバの間は、9600bps 全二重ワイヤレスモデム HD-9600[107] を使用して接続した。このモデムは、データの圧縮やフロー制御は一切行わずケーブルで直接接続することと同じ環境を提供する。また微弱電波を用いているので免許の必要はない。

## 6.2 評価

IP アドレス割り当て方式を実際に使用し、評価を行なった。

可搬ノードを Internet に接続する時には、まず可搬ノードがアドレス要求パケットをアドレス割り当てサーバに送信する。これは、可搬ノードを立ちあげるときに slclient を実行することで行なわれる。

接続にかかる時間を測定したところ、平均 160 ミリ秒であった。この時間は 16.7 ミリ秒単位で測定したものである。測定した時の RAW パケットのサイズは 40 バイトであったので、9600bps の回線で転送するためには 33 ミリ秒程度かかる。アドレスを割り当てる時にはこの RAW パケットが往復するので、約 66 ミリ秒かかる。測定された結果より、アドレスを割り当てる処理は 100 ミリ秒以下で終了していることがわかる。

アドレス割り当てサーバがネームサーバに登録する時にかかる時間を測定したところ、平均 250 ミリ秒であった。これは、10 ミリ秒単位での測定結果である。この測定を行なった時には、ネームサーバとアドレス割り当てサーバが同じホスト上にあったので、測定毎の時間の変化は少なかったが、ネットワークを介して登録を行なう場合にはネットワークの負荷により変化が大きくなる。また、ネームサーバのデータベースのサイズが大きくなると、書き換えに時間がかかる。

可搬ノードとアドレス割り当てサーバの間の通信速度を、ファイル転送プログラム ftp を用いて測定したところ、0.5k バイト毎秒程度の値が得られた。無線モデムの通信の速度は 9600bps なので、単純に計算すると最高で 1.2k バイト毎秒は出ることになる。

実際の IP パケットには、TCP/IP プロトコルで使用されるヘッダが追加されているので、モデムの最高速度ではデータを転送できない。SLIP を用いた回線の場合、一つのパケットで送れる長さは 1006 バイトで、ヘッダは最低でも 44 バイトの長さがある。したがって、パケットのヘッダによる損失は 4% にすぎず影響は小さい。転送効率の低下は、他の部分の影響によるものと考えられる。

次に、ネットワークファイルシステム (NFS) を用いて他のホストのディスクを可搬ノードに mount[103] する実験を行なった。NFS を用いてディスクをマウントする場合、マウントされるホストの /etc/exports ファイルに、アクセスを許可するホスト名を記述しなければならないが、アクセスされるホストがネームサーバを引くように設定してあれば、IP アドレスが変化する可搬ノードに対応することができる。

NFS マウントしたファイルシステムは普通に扱えるように見えたが、大きなサイズのファイルをアクセスすると、途中でタイムアウトを起こしてしまう。このために、NFS の設定を変えてタイムアウトの時間が長くなるようにしたが、うまく対応することができなかった。これの原因は、IP アドレス割り当て方式の問題ではなく、単に回線が遅いためである。

可搬ノードでもネームサーバを引くように設定したため、ホスト名を扱うアプリケーションを実行すると、ネームサーバに問い合わせにいき時間がかかる。遅い回線を使用する時には、/etc/hosts または NIS のデータベースになるべく多くの情報を入れて、可搬ノードからネームサーバをアクセスしないようにするべきである。

IP アドレス割り当て方式には特に問題はなかったが回線の遅さによる不具合が多くあった。もう少し速い無線デバイスが要求される。

## 6.3 応用方法

IP アドレス割り当て方式は実用的であると前に述べたが、この節では実際にどのような使用方法が有効であるかを考察する。IP アドレス割り当て方式は、前にも述べたように移動しながら使用することができないので、その点を考慮しなければならない。

ここでは、データ通信回線に無線と電話回線を使用した方法について述べる。

### 建物内の移動 (無線で接続)

可搬ノードを持って建物の内部を移動する使用方法である。可搬ノードとアドレス割り当てサーバの間は無線を用いて接続する。

図 6.1 に示すように、各部屋にはアドレス割り当てサーバとアンテナを設置しておく。アドレス割り当てサーバが無い部屋は、アンテナだけを設置し、近くのアドレス割り当てサーバに接続する。また、各アドレス割り当てサーバは、有線のネットワークで接続されている。

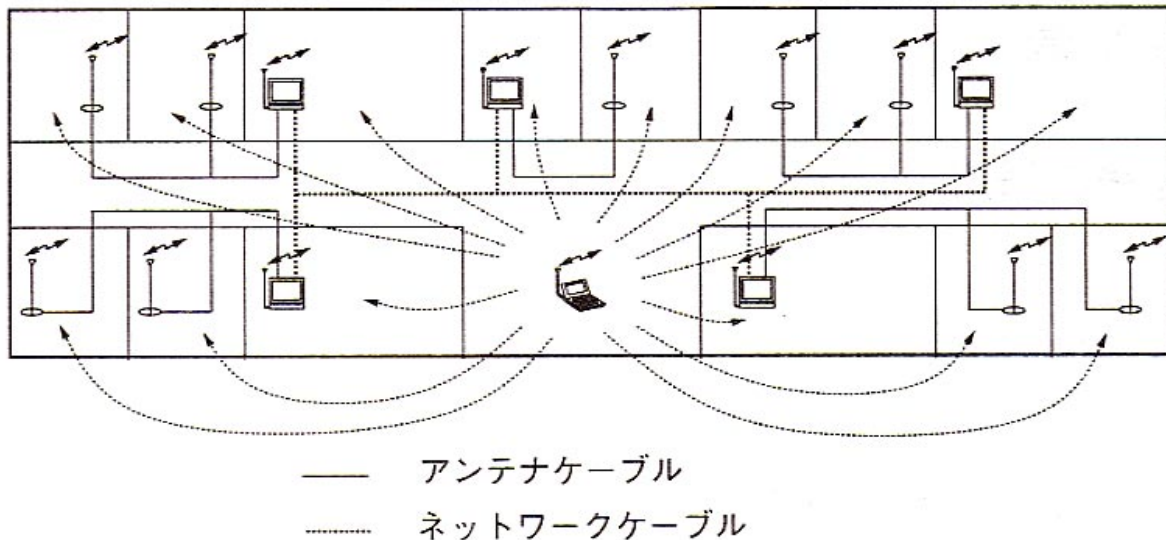


図 6.1: 部屋間を移動する可搬ノード

可搬ノードを他の部屋に移動させる時は、一度接続を切り IP アドレスを解放し、移動した部屋で再度 IP アドレスの割当を受ける。再度割り当てられた IP アドレスは、前のアドレスと異なることもある。

電波を使用しているので移動が簡単に行なえ、また、電波の届く範囲では移動しながら使用することも可能である。

#### 出先で使用 (電話回線で接続)

各地区に可搬ノード接続センターを作り、可搬ノードとセンターとは電話回線で接続する。各可搬ノード接続センターは Internet と接続しているので、他のネットワークのノードと通信することが可能である。地方に移動した時でも近くの可搬ノード接続センターに接続することで通信コストを最小にすることができる。また ISDN のような高速デジタル回線を使用することによって、固定ノードと同様の環境で可搬ノードを使用することができる。



## 第 7 章

### 今後の課題

この章では今後の課題について述べる。

#### 7.1 セッションの継続

IP アドレス割り当て方式では、移動することによって IP アドレスが変化する。このため、現在のネットワークアプリケーションでは、移動することによってセッションが切断されてしまう。この問題は IP アドレス割り当て方式を使用した場合、ネットワーク層では解決できない。

移動によるセッションの切断を避けるためには、アプリケーションで対応するかネットワークへのアクセス方法を変更する以外にない。

アプリケーションでの対応は、セッションを維持する機能をアプリケーション自身で持つ方法である。アプリケーションは可搬ノードへのパケットの転送がエラーになった時には、再度ネームサーバに問い合わせして IP アドレスを得る。これで、可搬ノードの IP アドレスが変化した場合でも通信は保たれる。

アプリケーションがネームサーバに問い合わせても IP アドレスが得られない場合もある。これは、可搬ノードが移動中でどのネットワークにも接続していないことを示している。この時には、エラーで終了せず、可搬ノードがどこかのネットワークに接続するまで待つようにする。セッションの再開は、可搬ノードがネットワークに接続した時に、このアプリケーションに通知することで行なうことができる。

ネットワークへのアクセス方法の変更は次のような対応である。現在の UNIX システムでは通信を行なうためにソケットが用いられている。このソケットを使用した通信では、通信相手の指定を IP アドレスで行なっているが、それをホスト名に変えてしまう。これにより、IP アドレスの変化はアプリケーションからは見えなくなる。しかし、この対応ではネットワークインタフェースに膨大な変更を必要とし、また、すべてのアプリケーションの変更を伴うので現実的ではない。

以上より、IP アドレス割り当て方式においてセッションを維持するためには、各アプリケーションで対応することが最良である。

## 7.2 ホスト名とネットワークの矛盾

IP アドレス割り当て方式ではホスト名をノードの識別に用いるため、ホスト名の指すドメインと実際に存在するネットワークが異なる。このためホスト名のドメインを直接扱うアプリケーションでは不都合が起こる。

例えば、メールのアドレスはホスト名で指定するが、可搬ノードが移動している時には遠回りをしてしまうか、届かない可能性もある。

メールの問題はネームサーバを用いることで解決できる。これは、現在のメール配送システム (sendmail)[103] にはネームサーバに対応したものがあり、それを使用することでメールの遠回りを避けることができるためである。

このように、ホスト名の問題はアプリケーションがネームサーバに問い合わせることで解決する。

## 7.3 イーサネットへの拡張

今回の IP アドレス割り当て方式の実装では、可搬ノードとアドレス割り当てサーバを point-to-point で接続するが、イーサネット [5] のようなブロードキャスト型のネットワークを用いた接続も考えられる。

イーサネットに、ディスクレスのホストのような自分の IP アドレスを記憶できないノードを接続する時には、そのノードの IP アドレスを他のノードに教えてもらう必要がある。このためのプロトコルを RARP (Reverse Address Resolution Protocol)[5] という。RARP はホストのイーサネットアドレスを鍵にして IP アドレスを引く方式で、イーサネットアドレスと IP アドレスの対応はあらかじめ登録されていて固定である。

このプロトコルを IP アドレス割り当て方式に対応させると、可搬ノードをイーサネットへ接続することが可能になる。RARP を変更する点は、イーサネットアドレスと IP アドレスの対応を可変にすることと、可搬ノードのホスト名を受けとりネームサーバに登録する機構を付加することである。

## 7.4 セキュリティ

今回の IP アドレス割り当て方式の実装では、IP アドレスを割り当てる時にノードの認証を行なう。しかし、実際にはネームサーバに登録する時の認証システムも必要であり、またデータの暗号化も重要である。

ネームサーバに登録する時の認証システムはネームサーバ更新デーモンと交信を開始する時にパスワードを入力することで解決できる。このパスワードは可搬ノードから入力されたパスワードではなく、各アドレス割り当てサーバが持つものである。

データの暗号化はさらに重要である。それは、可搬ノードとの通信に無線を使用する場合があり、盗聴される危険があるからである。

また暗号化だけでは、データを記録して同じデータを再度送信することによる妨害が行なえるが、これにはタイムスタンプやシーケンス番号を用いた対応が考えられる。

もともと、現在の Internet はセキュリティがあまく、他のノードになりすますことが簡単にできる。これを解決するために、MIT の Athena プロジェクトでは kerberos[108] と呼ばれる認証システムを開発している。このシステムを可搬ノードの接続時における認証に対応させることも考えられる。

## 7.5 移動中のネットワークへの接続

本論文では、動的アドレス更新型である IP アドレス割り当て方式の実装について述べた。この方式では移動しながら可搬ノードを使用することができないと述べた。これに対応するためには、アプリケーションもしくはネットワークインタフェースを変更しなければならない。7.1節で述べたようにセッションを回復できるなら、断続的に Internet に接続することは可能である。

しかし接続したまま移動するためには3章で述べた動的経路更新型の経路制御アルゴリズムを用いる必要がある。ここでは、動的経路更新型の中から専用ネットワーク方式をとりあげて考察する。この方式は、他のネットワークに変更を加える必要がない点で他の方式に比べて優れている。

大規模に構築しないと移動できる範囲が狭くなると述べた。しかし、一つのネットワークを複数の専用ネットワークに分割し、その間を既存のネットワークで補間するという方法により移動できる範囲を広げることができる。

例えば、東京と大阪にそれぞれ専用ネットワークを構築し、その間は有線のネットワークで接続する。こうすることで可搬ノードは、東京と大阪の間を移動中には使用できないが、東京でも大阪でも同じ IP アドレスで使用することが可能となる。

このようにすると専用ネットワークへのゲートウェイが複数でき、どのゲートウェイに転送すれば良いかを考えなければいけないように見える。しかし、この場合には図 7.1 のように自分に最も近いゲートウェイにパケットを転送すれば良く、その後はゲートウェイ間で転送される。これは可搬ノードがその時に接続している専用ネットワークの情報を、ゲートウェイの間で交換するためである。

専用ネットワークないでは、今までの経路制御アルゴリズムを使用することはできない。これには新しい経路制御アルゴリズム OSPF (Open Shortest Path First) [109] の使用が検討される。OSPF は、複数のルートの中から最短になるものを選択するので、ノードが移動した時のパケット経路の変化にうまく対応できる。これを可搬ノードに応用するためには、可搬ノードの移動の情報が素早く伝達されるようなデータベースの更新アルゴリズムの追加が必要である。

しかし、OSPF のようなデータベースを用いた経路制御の方式は可搬ノードの数が増えると破綻をまねくおそれがあるので、さらに良いアルゴリズムの開発が必要である。

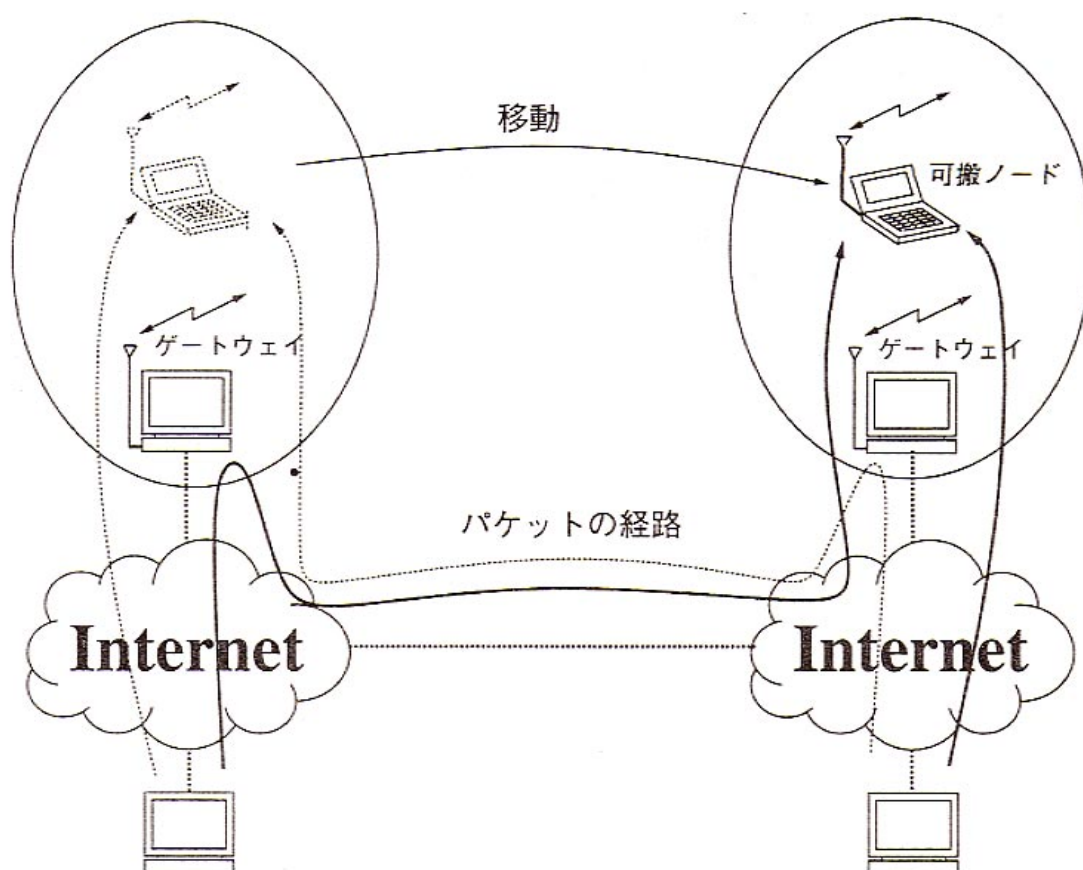


図 7.1: 複数に分割された専用ネットワーク

## 第 8 章

### まとめ

本論文では、可搬ノードを Internet に接続する時の問題点について考察した。この問題は可搬ノードの識別とパケットの経路制御情報は互いに独立でなければならないが、現状の Internet ではそのようになっていないというものである。

この問題点を解決する方法として、パケットの経路制御の方法を改良する動的経路更新型と、ホストの識別情報を変える動的アドレス更新型の二つの方式について述べた。また実際に動的アドレス更新型の IP アドレス割り当て方式を取りあげ実装した。

ここで実装した IP アドレス割り当て方式は、Internet プロトコルに変更を加えず、すでにあるネームサーバの機能を利用して可搬ノードに対応する方法である。

このため容易に実装することができ、現在の環境に手を加える必要がなく他のノードに対しても影響を与えないので実用的である。

移動しながら使用できないという欠点はあるものの、用途を選択すれば有効に利用できることが明らかになった。

