

インターネット応用特論

7. 文字通信: ウェブ、HTTP、HTML、
JAVA

太田昌孝

mohta@necom830.hpcl.titech.ac.jp

<ftp://ftp.hpcl.titech.ac.jp/appli7.ppt>

ところで、インターネットとは？

- 電子メールのことではない
 - 数年前には大真面目で主張されていた
- ウェブのことでもない
 - 現在は勘違いしている人が多い
- アプリケーションのことではない
- インターネットはIP(インターネットプロトコル)を用いて、インターネットの原理に基づいて接続された網である

アプリケーション？

- 面の皮（美人にみえるには重要だが）
 - iモード公式サイトの全収入はドコモの収入の4%程度
- ウェブはどこで発明されたか？
 - 米国にきまっている！？（ジュネーブでのINET会議でのスイス人の演説）
- ないところまるが、何かあればいい

放送

放送網

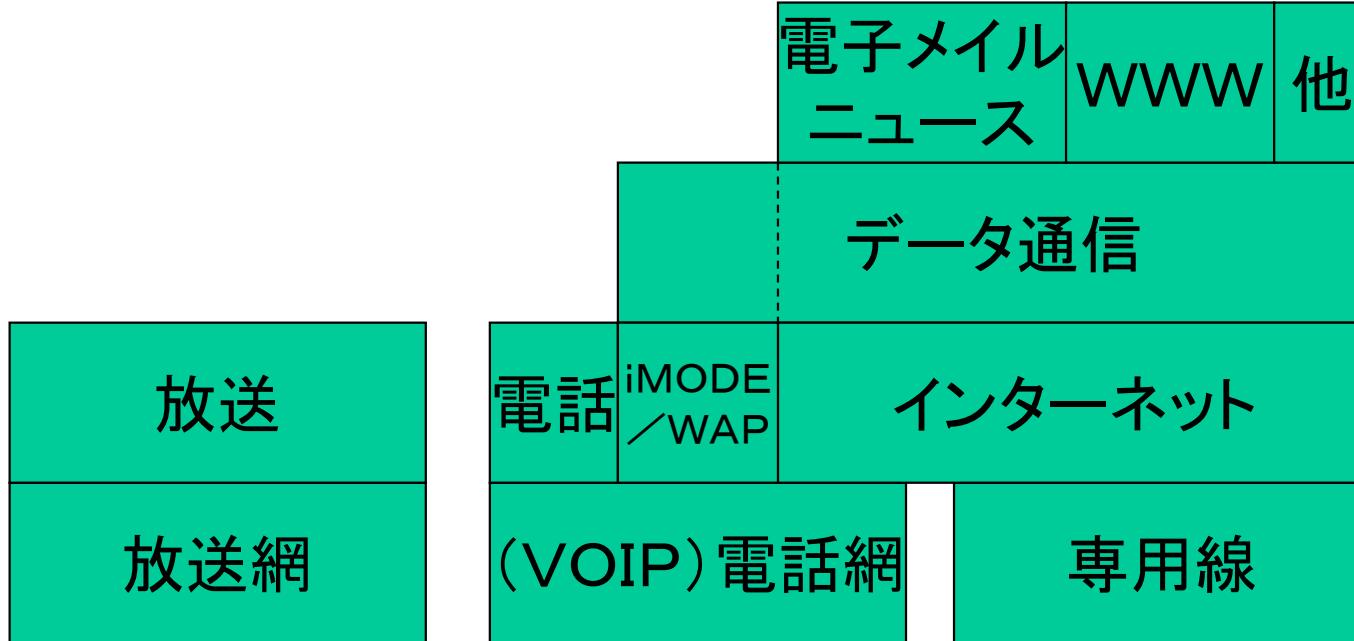
電話

電話網

データ通信

専用線

かつてのネットワーク



現在のネットワーク

放送	電話	電子メール ニュース	WWW	他			
ストリーミング		データ通信(バッチ)					
インターネット							
専用線(含無線)							

今後のネットワーク

ウェブの歴史

- そもそもはテレテキストシステム
 - NAPLPS(北米)、CAPTAIN(日本)は失敗
 - MINITELはフランス近辺で大流行
 - 容易な情報発信、端末無料化などが効いた
- MINITELのインターネット版がウェブ
- HTML、URI、HTTP、JAVA、、、
- 大成功の秘訣は
 - インターネットにあり

HTML (Hypertext Markup Language、RFC1866)

- ハイパーテキスト
 - ハイパーリンク(URI)の埋め込まれたテキスト
 - マルチメディア業界では30年前からある概念
- マークアップ
 - 書式をととのえる指示
 - 字サゲ等を“<”と“>”で囲んだ文字で指示
 - 元々は欧米中心だが、だいぶ改善された
 - 縦書き、ネストした双方向性の扱いなど

URI (Uniform Resource Identifier、RFC2396)

- URN(Uniform Resource Name)
 - 資源を識別する
 - 参考文献の書き方の一種でしかない
- URL(Uniform Resource Locator)
 - 資源の利用方法がわかる
 - ・インターネット上の資源は、サーバのアドレスも
- クリックやサーフができるのは
 - インターネット上で即時利用可のURLだから

URIの例

ftp://ftp. is. co. za/rfc/rfc1808.txt

 -- ftp scheme for File Transfer Protocol services

gopher://spinaltap.micro.umn.edu/00/Weather/California/Los%20Angels

 -- gopher scheme for Gopher and Gopher+ Protocol services

http://www.math.uio.no/faq/compression-faq/part1.html

 -- http scheme for Hypertext Transfer Protocol services

mailto:mduerst@ifi.unizh.ch

 -- mailto scheme for electronic mail addresses

news:comp.infosystems.www.servers.unix

 -- news scheme for USENET news groups and articles

telnet://melvyl.ucop.edu/

 -- telnet scheme for interactive services via the TELNET Protocol

URIの書式

- 最も一般的には
 - <scheme>:<scheme-specific-part>
- より詳しくは
 - <scheme>://<authority><path>?<query>
- 役に立つ<authority>は<server>
 - server = [[userinfo "@"] hostport]
 - hostport = host [":" port]
 - host = hostname | IPv4address

インターネットでのオブジェクト処理

- URLで指すリソース=オブジェクト
- URLの冒頭のscheme(とpathの末尾)でクラスを識別
- pathはserver内でオブジェクトを識別
- URLの後部のqueryでメソドを識別
 - デフォルトは「アクセス」
- queryは、パラメータも渡せる

TCPとコマンド

- TCP上でASCII文字でコマンドを出し応答を受け取る
- 行末はCRとLFで区別
- データは同じTCPコネクションで送ってもいい(SMTP)し、別立てでも(FTP)でもいい
 - 同じで送る場合、区切りが必須

FTP (RFC959)

- File Transfer Protocol
- インターネットでファイルをやりとりするプロトコル
- ポート番号21をコマンドに利用
- 多用なファイル形式をサポート

HTTP (Hypertext Transfer Protocol、RFC2616)

- FTPやSMTP同様ASCIIベースでコマンドに対する応答を返す
- OPTIONS
 - オプションが何か問い合わせ
- GET
 - URLの内容をヘッダとともに取得
- HEAD
 - URLの内容のヘッダだけ取得

HTTP (2)

- POST
 - URLにデータを送りつける
- PUT
 - URLの内容を送るデータで置き換える
- DELETE
 - URLの指す資源を削除

HTTP (3)

- TRACE
 - プロキシチェーンのリストを取得
 - インターネットとは無関係
- CONNECT
 - 予約
 - ・ 将来プロキシーからトンネルへの切り替えに

HTTPリプライの意味(1)

- 100番台
 - 中間報告
- 200番台
 - 成功
- 300番台
 - リダイレクト
- 400番台
 - クライアントエラー

リプライの意味(2)

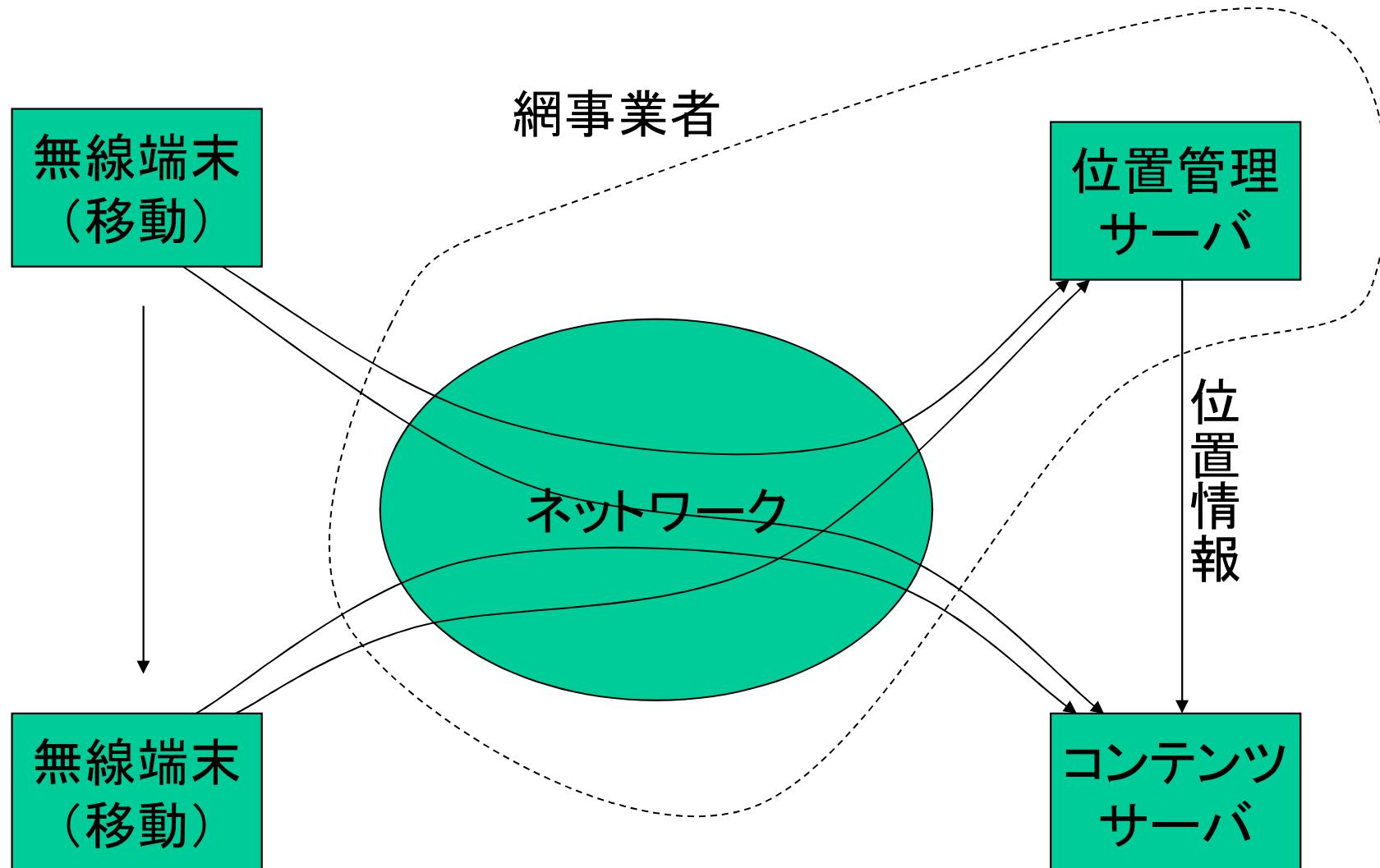
- 500番台
 - サーバーエラー

インターネットとコンテンツ規制

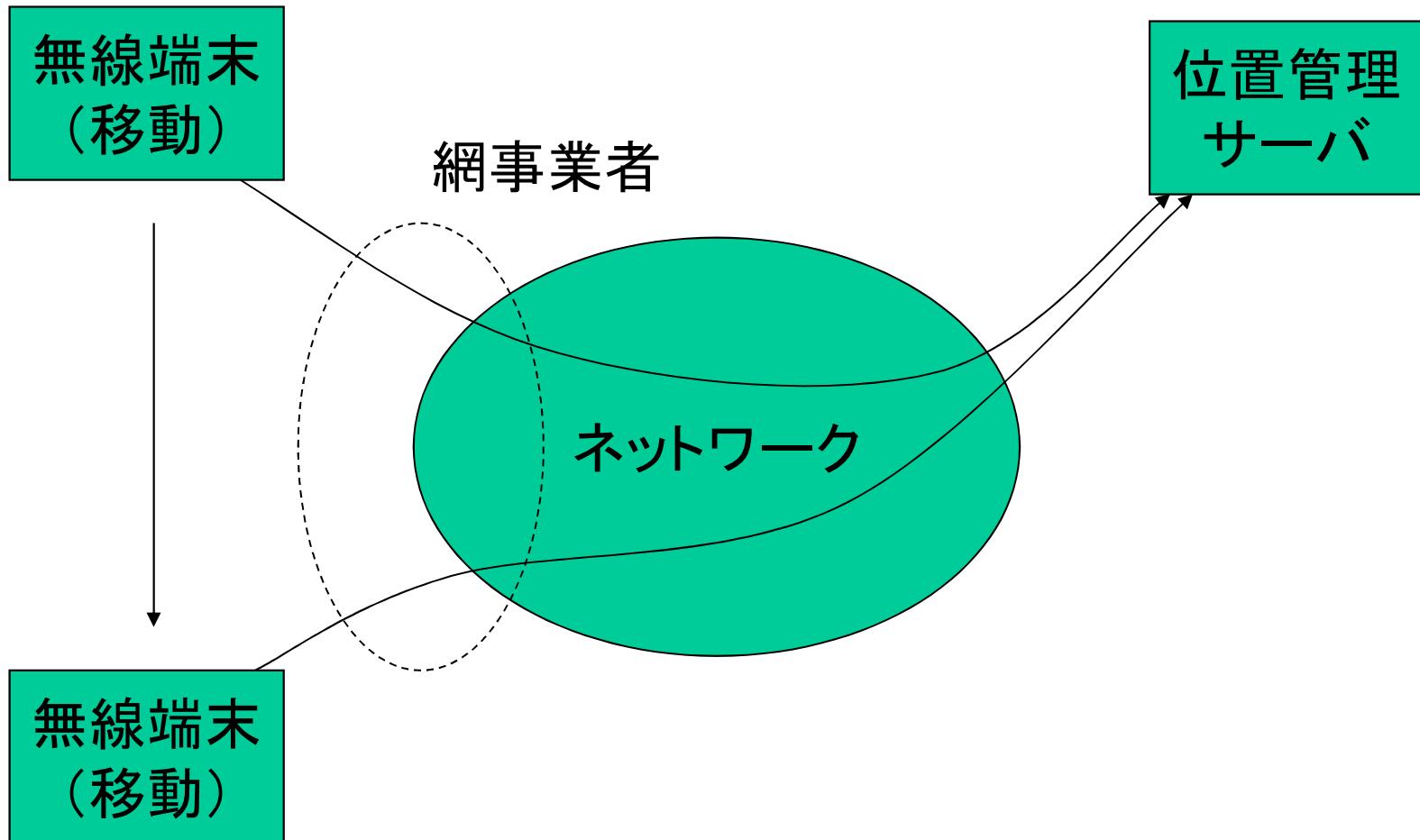
- ダイアルアップインターネットでは
 - サーバはネット事業者が集中して運営
 - ネット事業者は規制可能
- 常時接続インターネットでは
 - 誰もがサーバを運営可能(完全な分散)
 - ピアツーピアモデル
 - 規制すべき業者が不在

位置依存サービス

- 端末の位置に応じたコンテンツを供給
- IP層では無理
 - IPヘッダーには位置情報は入らない
 - アプリケーション層で対応
- 既存サービスと使い勝手は同一に
 - コンテンツサーバでの位置情報の取得は自動化したい



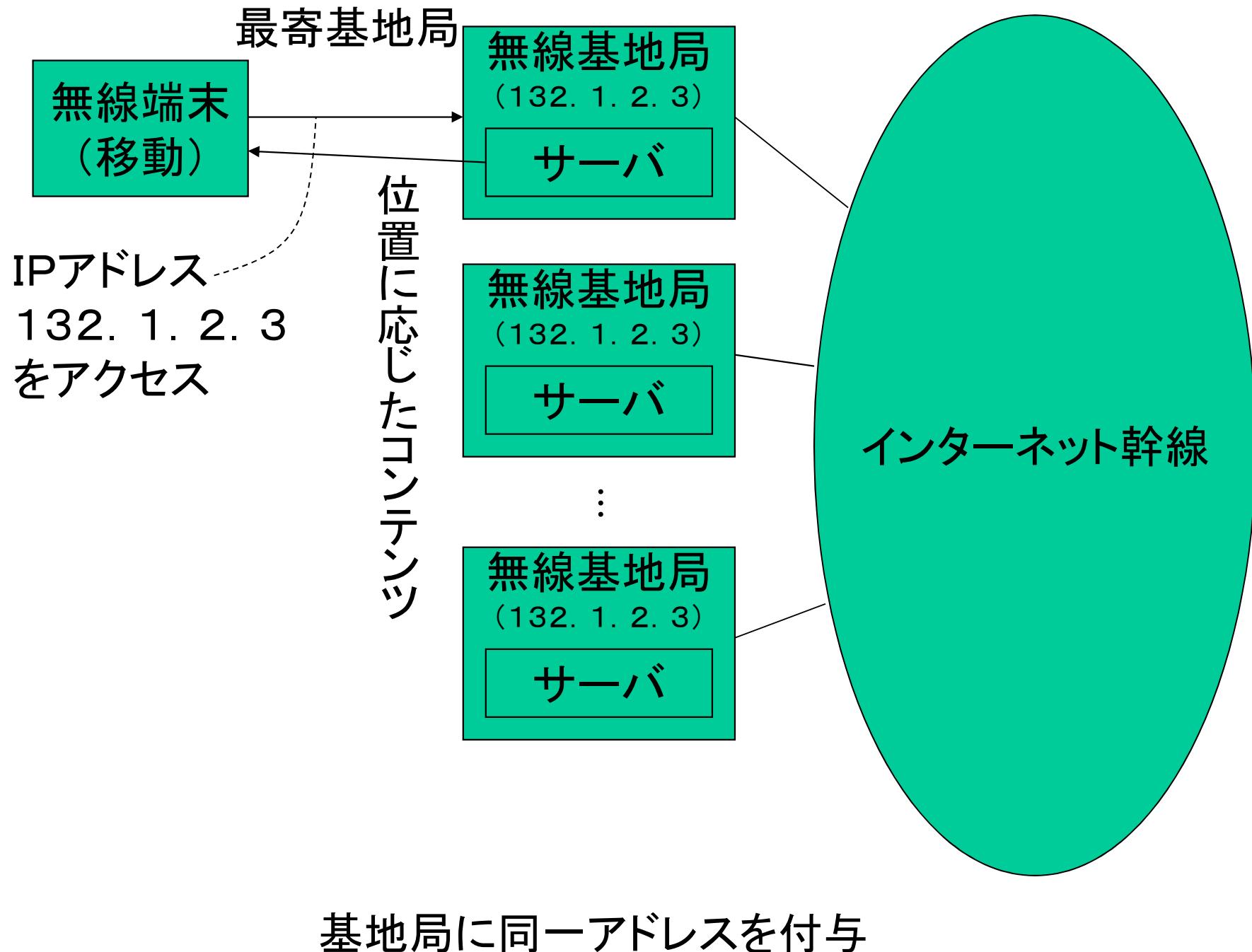
携帯電話網での移動体と位置依存情報



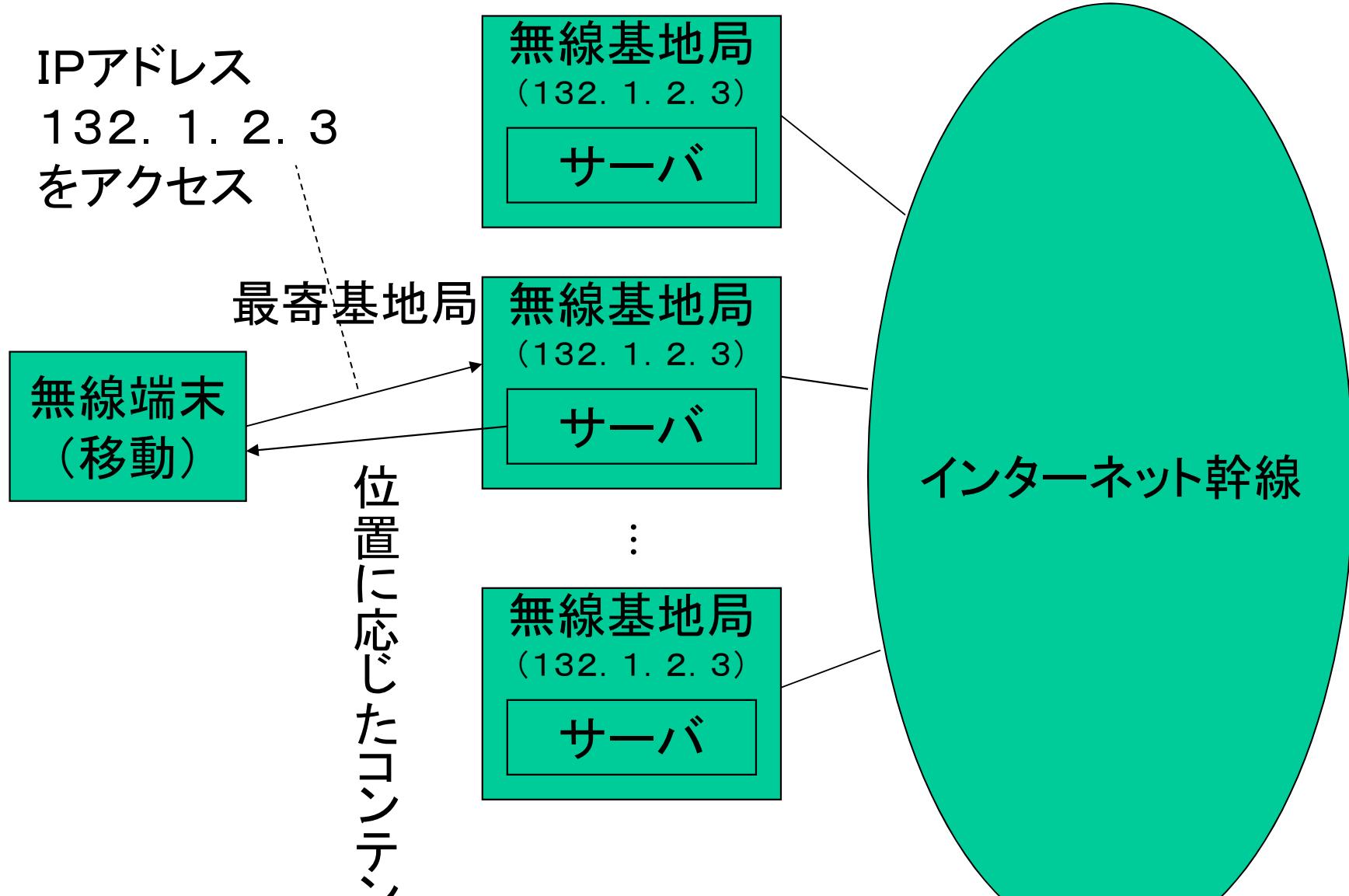
インターネットでの移動体

エニキャストによる 分散型位置依存サービス

- 異なる位置の複数の無線基地局に同じIPアドレス(エニキャストアドレス)を付与
- 無線端末は、同じIPアドレス(URL)で、最寄の無線基地局上をアクセス
- 無線基地局は位置に依存した情報を提供
 - あるいは、位置情報を付加したURLを付加して他のURLに転送



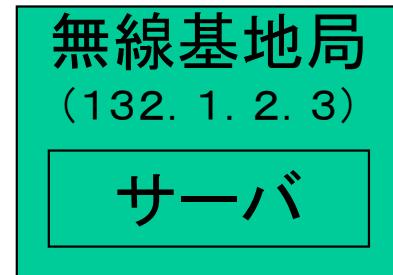
IPアドレス
132. 1. 2. 3
をアクセス



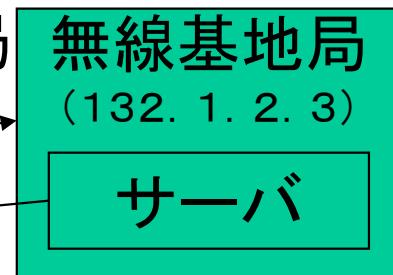
基地局に同一アドレスを付与 <移動後>

1) URL:
<http://map.here/>
をアクセス
(map.hereのIP
アドレスは
132.1.2.3)

無線端末
(移動)



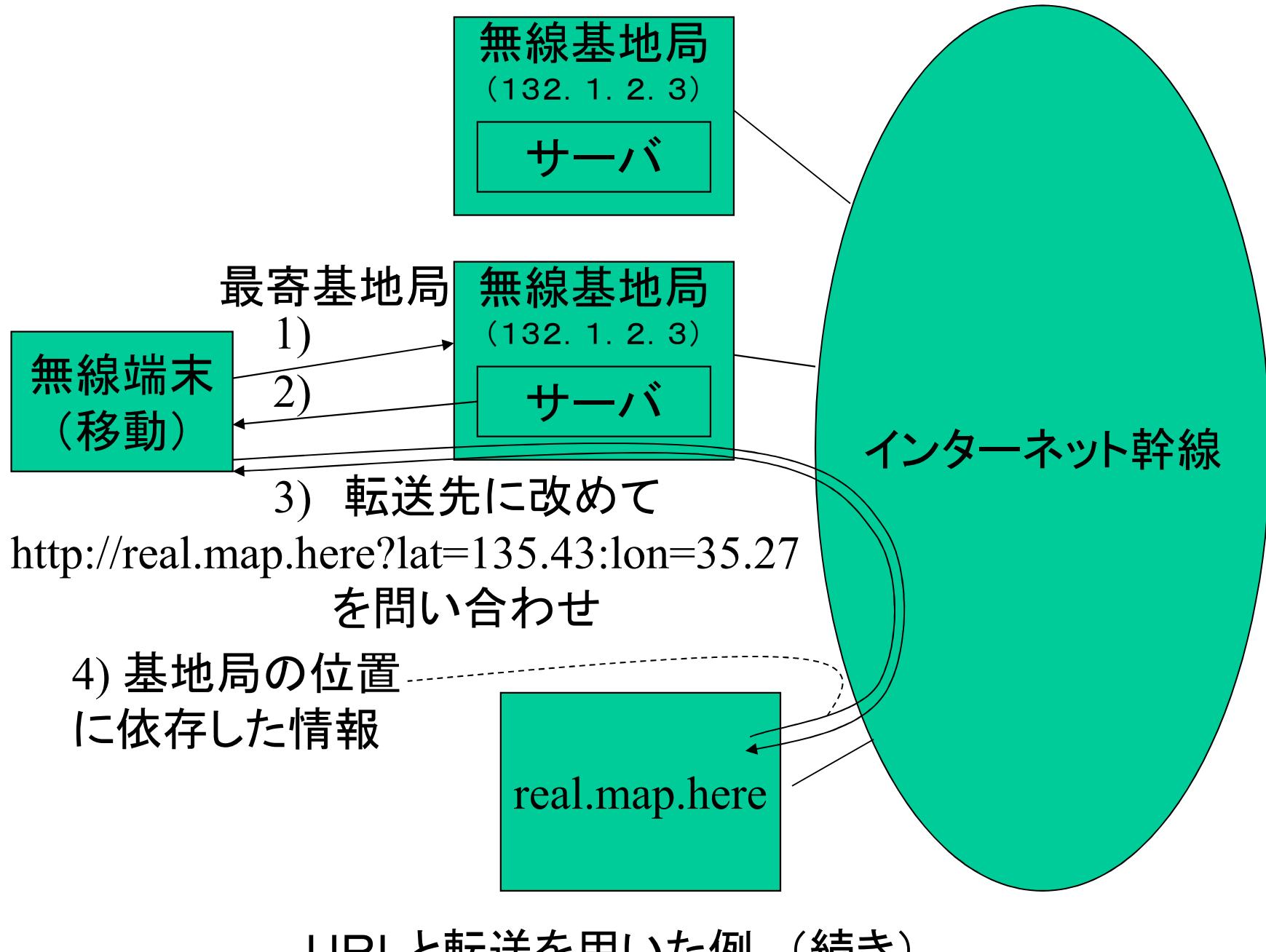
最寄基地局



2) URL:
<http://real.map.here?lat=135.43:lon=35.27>
に転送



URLと転送を用いた例



エニキャスト方式の利点

- 完全分散処理
- プライバシー管理問題がない
 - 場所情報を受け取るのはユーザ

JAVA

- HTMLに埋め込み
- クライアント側で解釈実行
- クライアント側で画面操作すると
 - JAVAアプレットが起動
 - クライアント側で各種動作をする

JAVAアプレット

- クライアントがサーバのプロトコルを知らなくても
 - ブラウザがありJAVAアプレットを動かせば
 - サーバと通信可能
- 一見まともだが
 - 実はすべて人間まかせ
 - ブラウザの内容が理解・信用できるとは限らず

JAVA APIの標準化

- プロトコルがわからなくとも
 - JAVA APIを標準化しておけば
 - ネットワークからJAVAコードを送り込んで相手を操作できる
- プロトコル決めるよりAPI決めるのが楽?
 - プロトコルを決めておけばJAVAなど不要
 - 相手のJAVAプログラムを無条件で実行?
 - 要求がプロトコルレベルでわからないと危険

まとめ

- ウェブは現在は全盛のアプリ
 - 電子メール・ニュース時代からスムーズに移行
 - クライアント・サーバモデル
 - アプリレベルで遊ぶには手ごろ
- コンテンツはHTML、転送はHTTP
 - URLがオブジェクトID
- JAVAは計算機屋の領域