

# 無線網路多媒體系統 Wireless Multimedia System

## Lecture 7: Network Mobility

吳曉光博士

<http://wmlab.csie.ncu.edu.tw/wms>



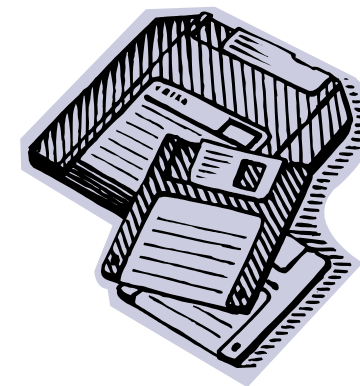
# Agenda

- ◆ All-IP System: Beyond 3G
- ◆ Evolutions of PCS
- ◆ ALL IP Challenges
  - Mobile IP/Cellular IP
  - QoS Provisions: Integrated Service / DiffServ
- ◆ Next Week (Mobile IP)



# Reading

- ◆ [Bhagwat96] Pravin Bhagwat, Charles Perkins, and Satish Tripathi, "Network Layer Layer Mobility: An Architecture and Survey

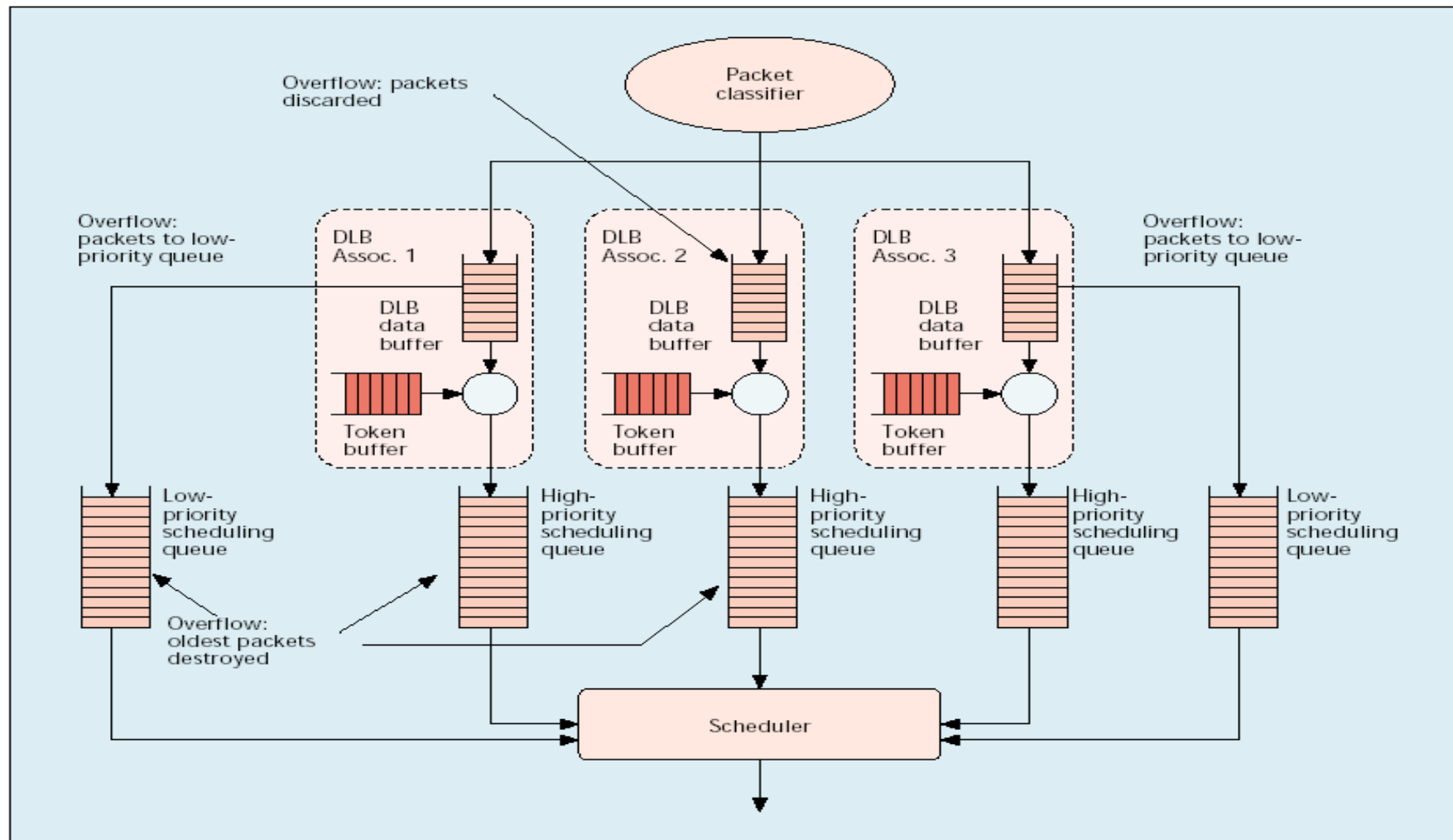


# All IP

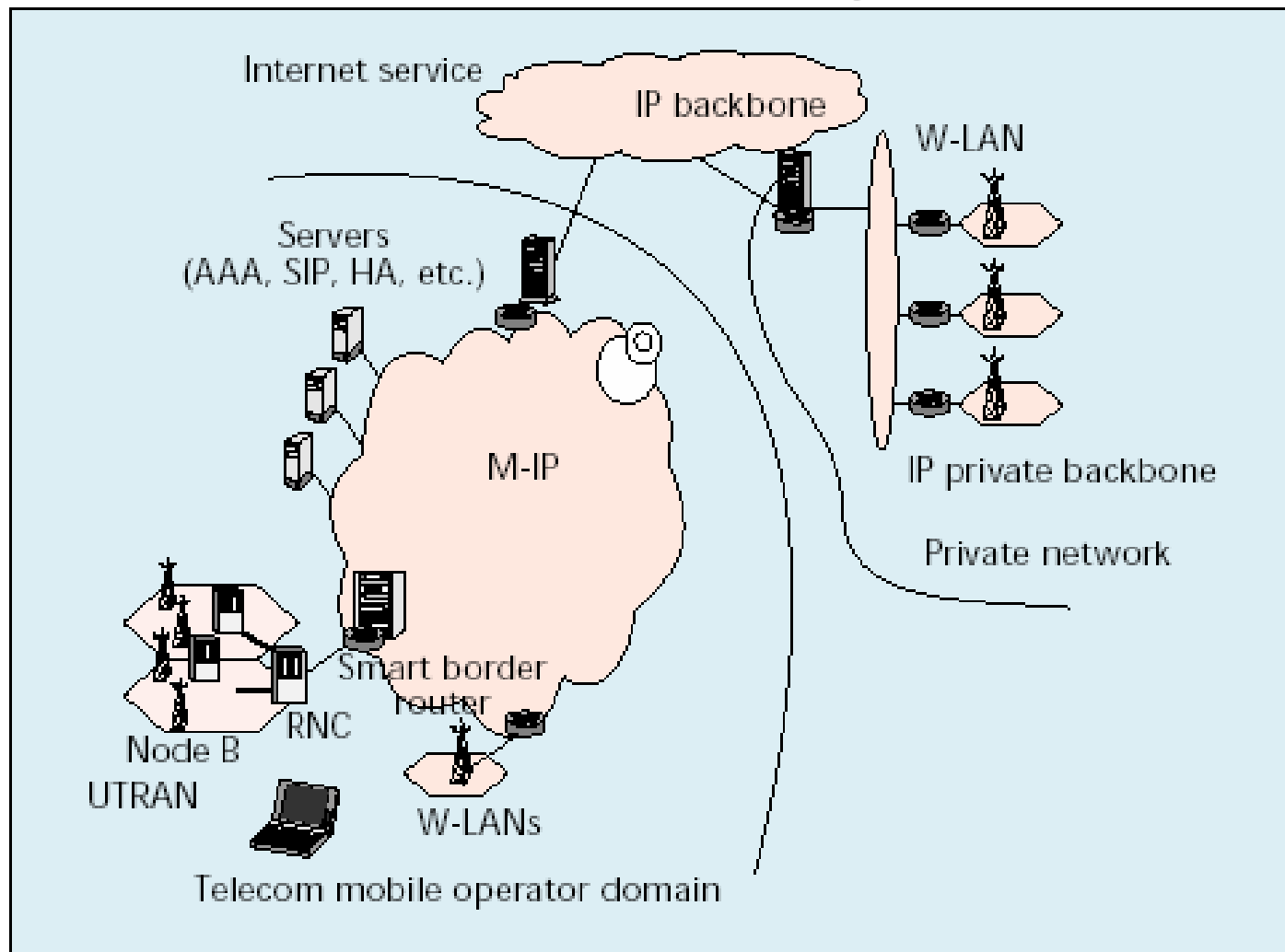


## Something to happen?

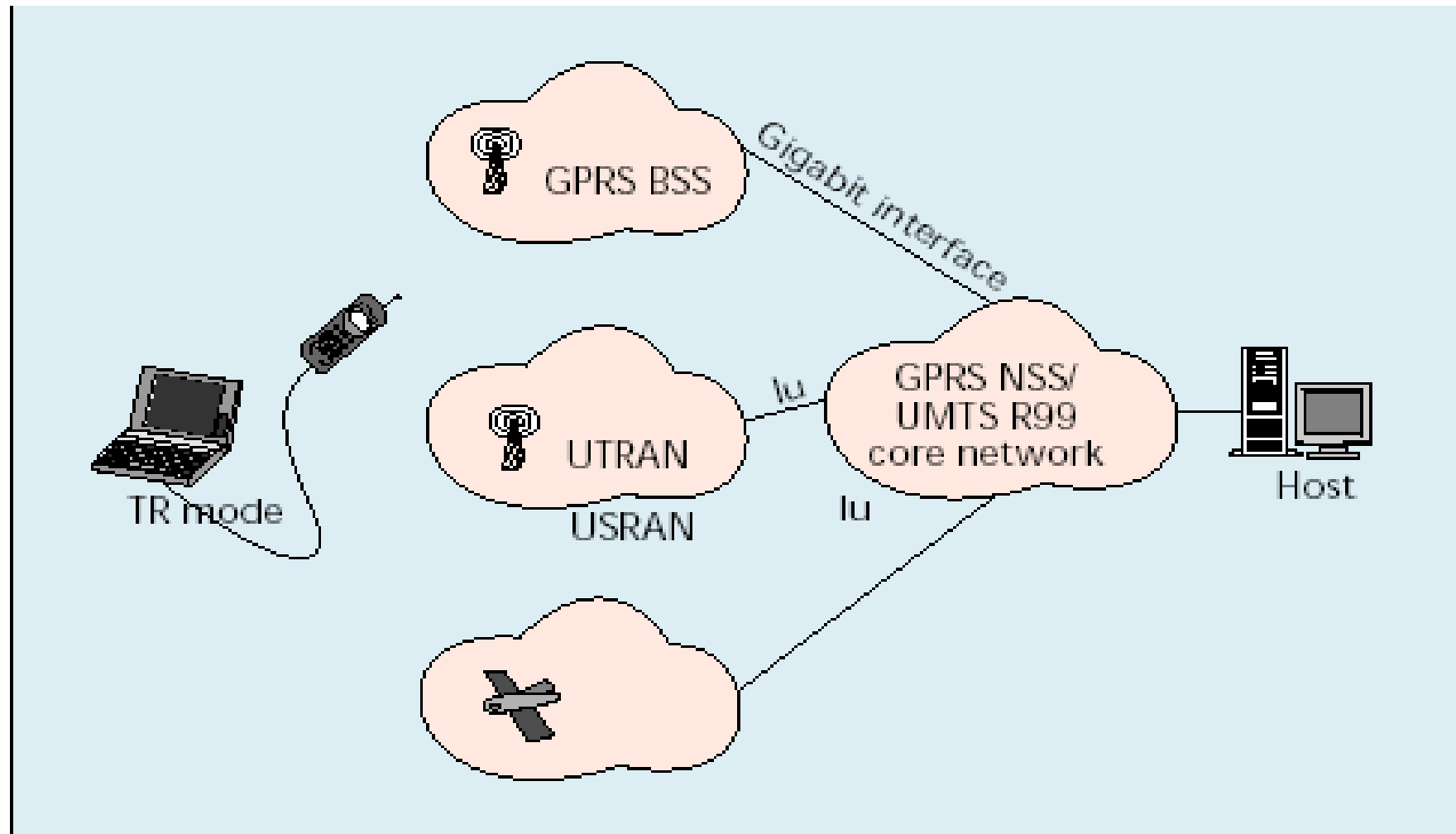
# MT Scheduler



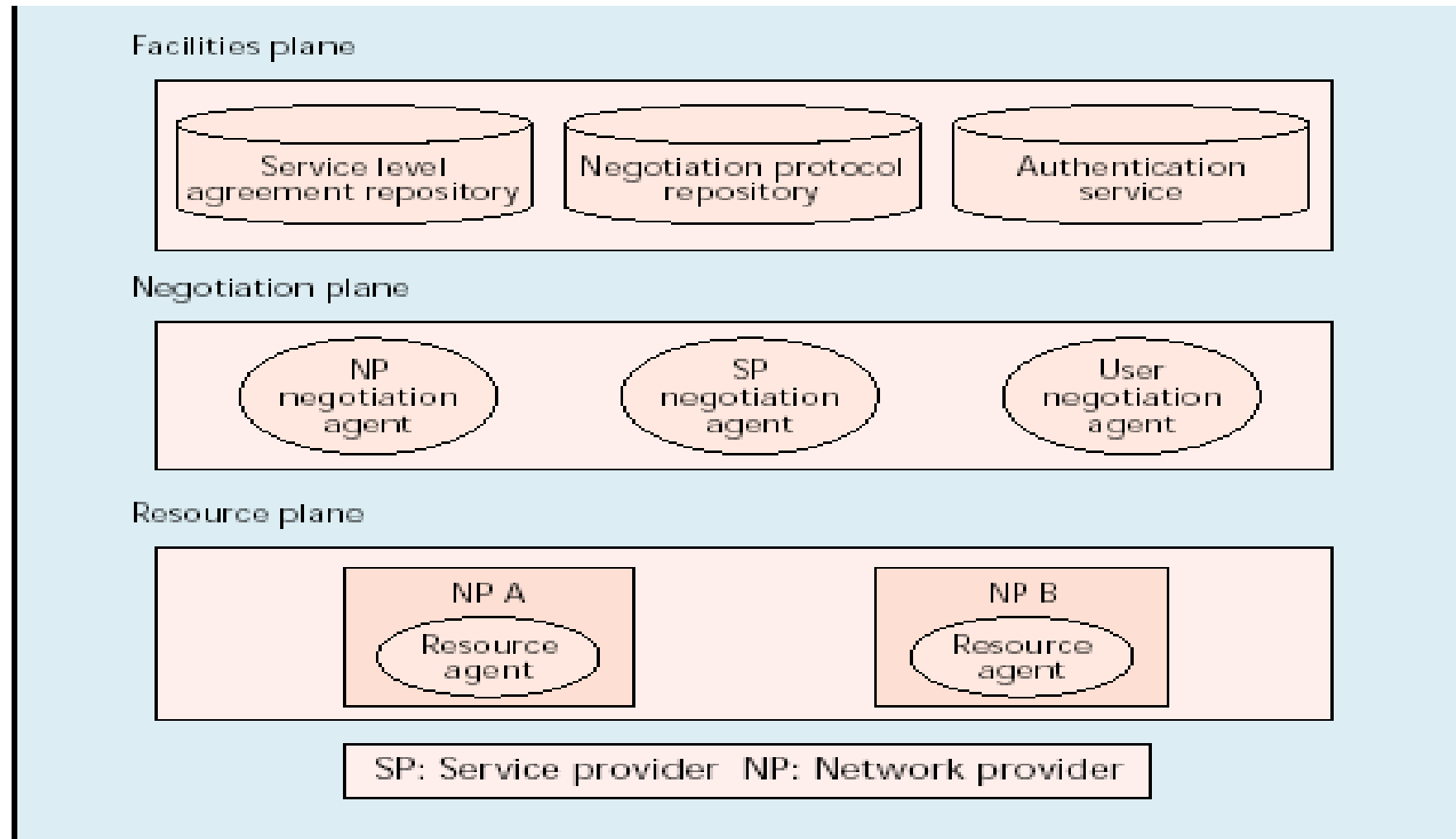
# A IP reference Architecture for Wireless Mobile System



# Integration Scenario

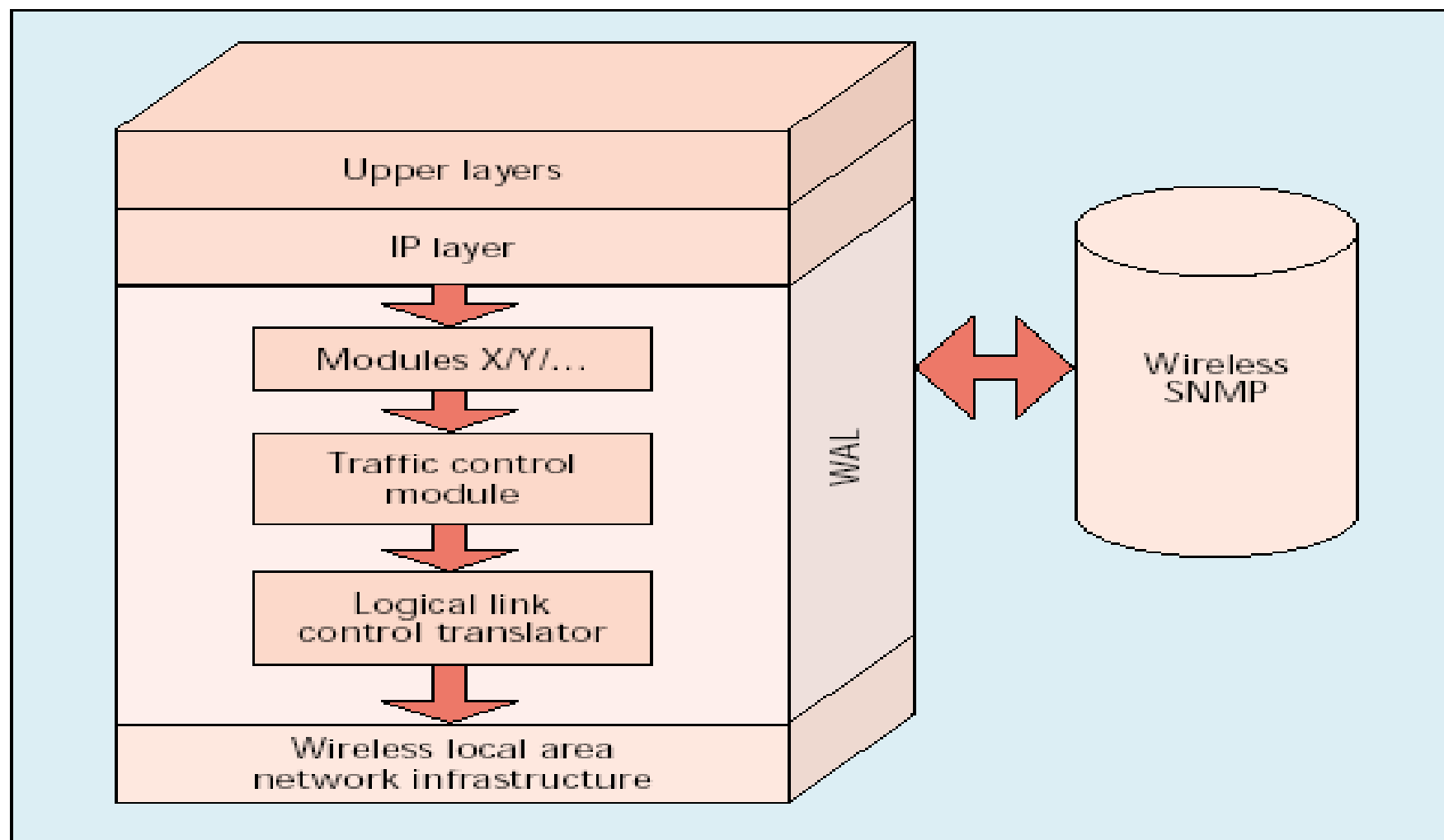


# Resource Managements

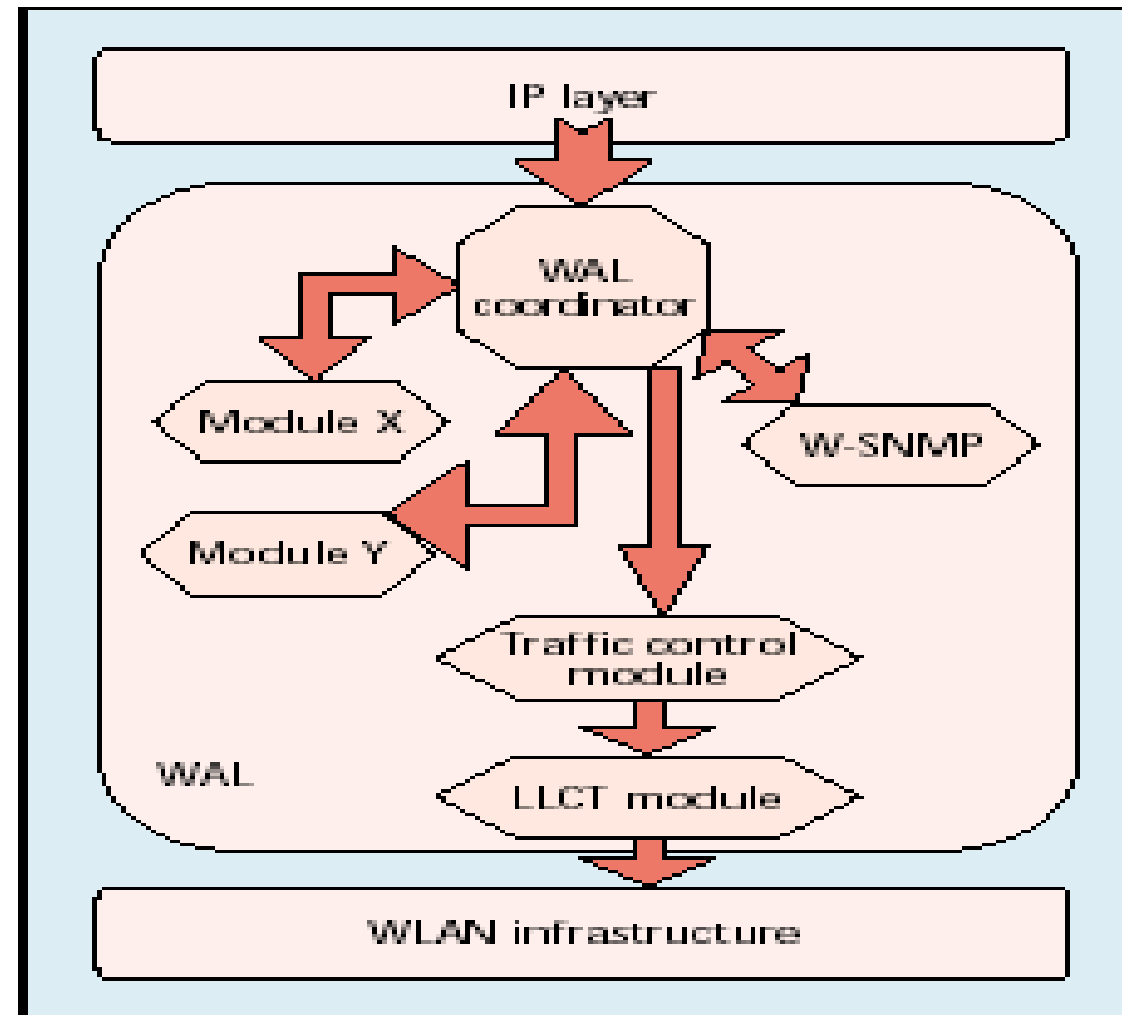




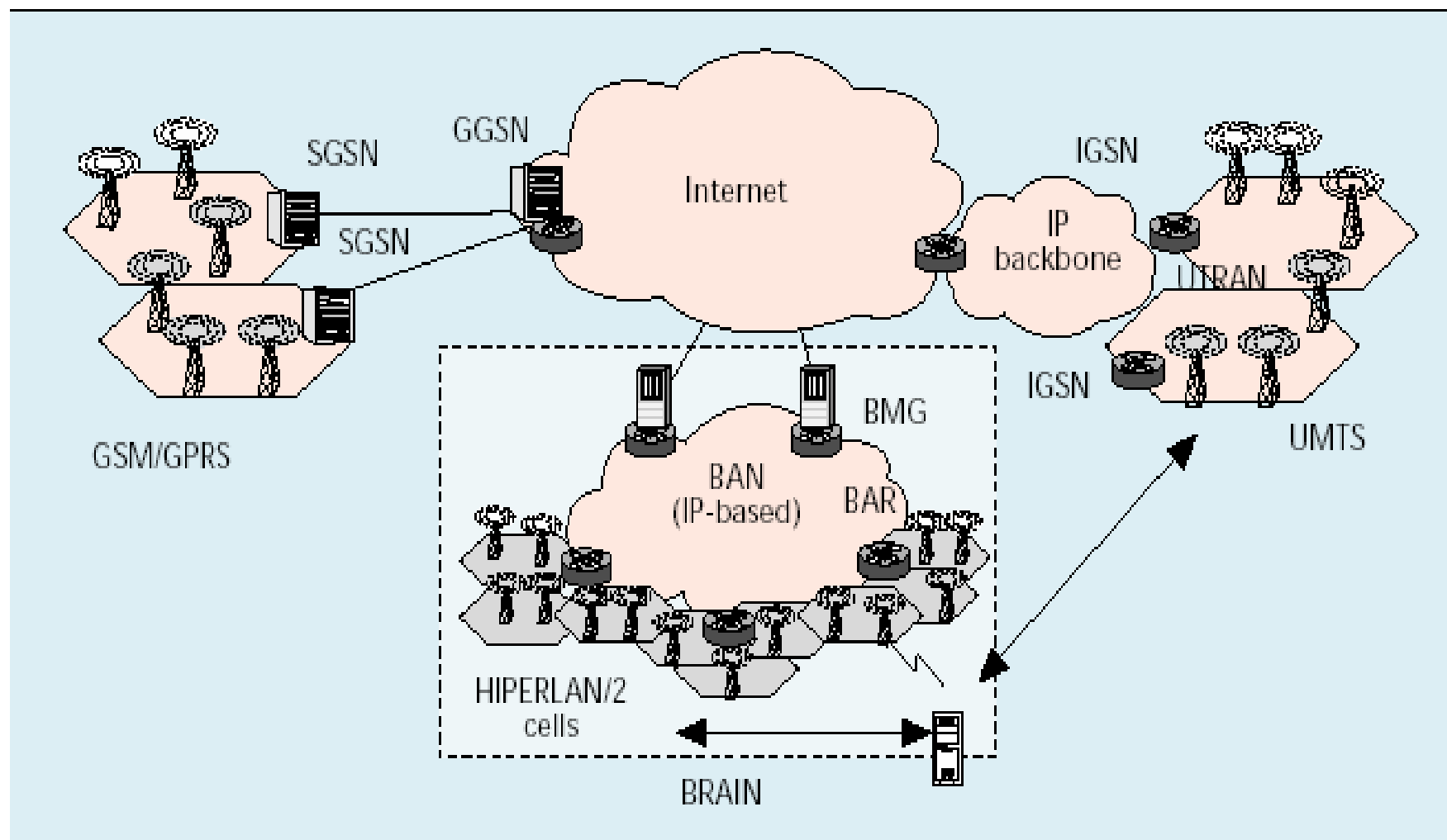
# WAL



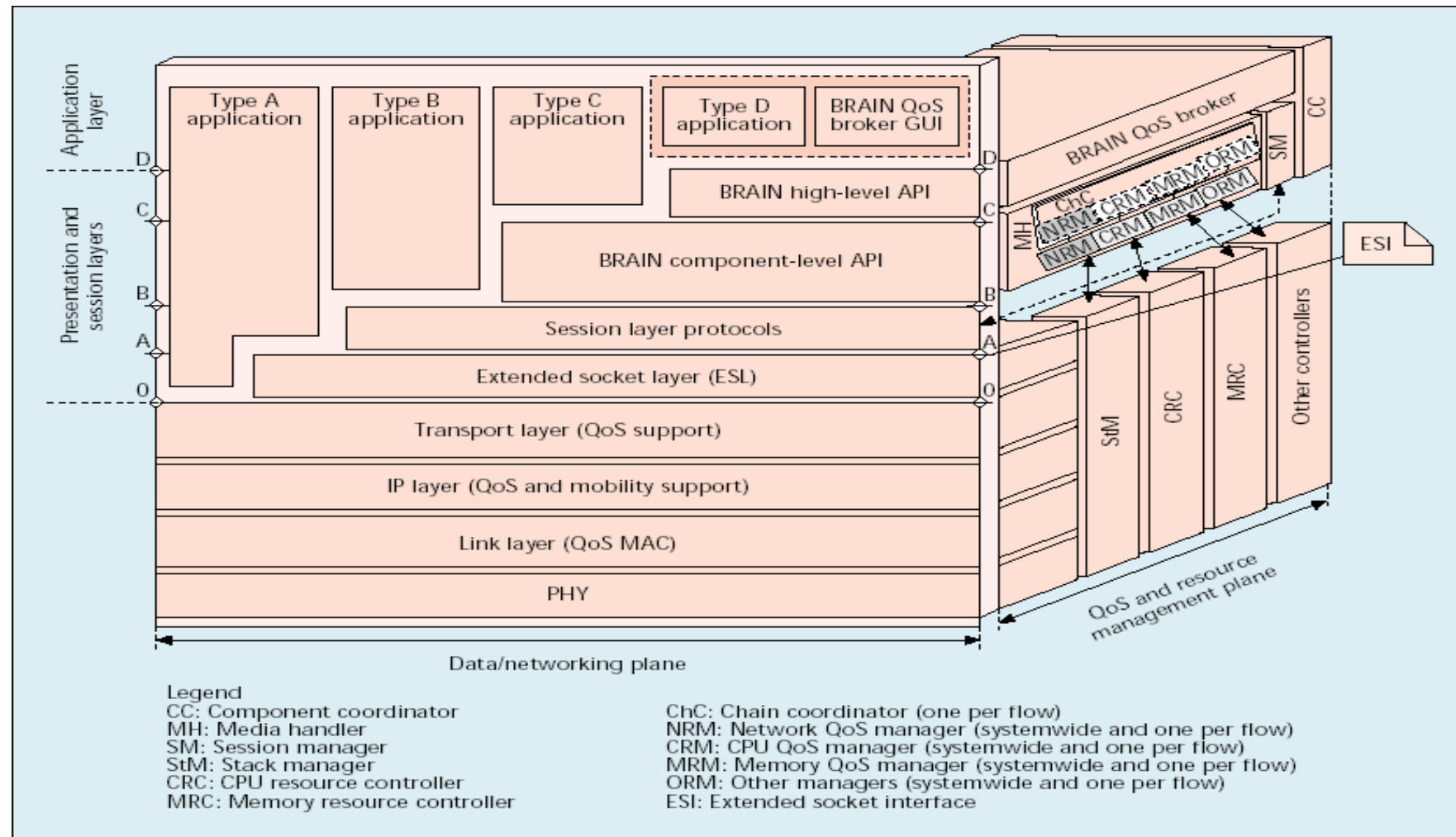
# Detail WAL



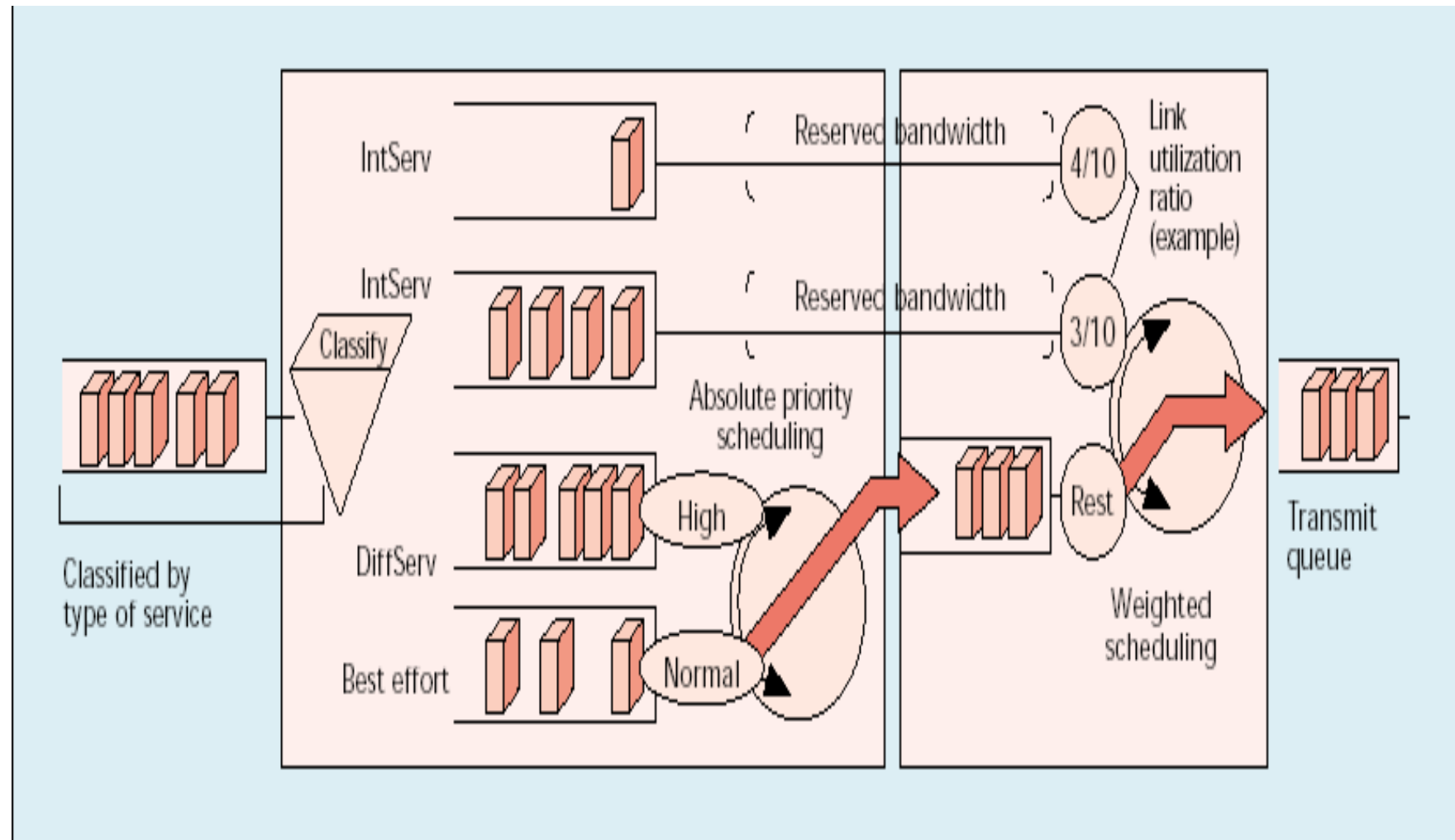
# BRAIN



# QoS Support



# IP QoS Modeling



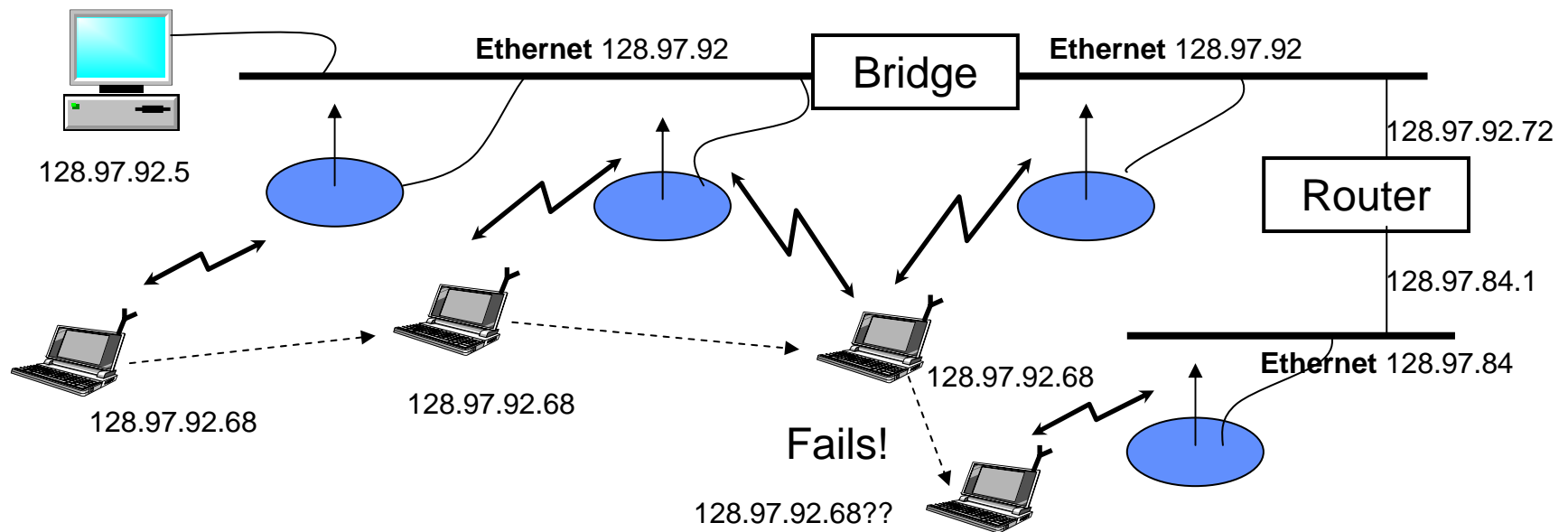
# Lecture Outline

- ◆ Mobility in wireless LANs
- ◆ Problems in making Internet mobile
- ◆ Canonical packet forwarding architecture for Mobile-IP
- ◆ Columbia's Mobile-IP schema

# Making the Internet Mobile

- ◆ Goal
  - Provide continuous IP connectivity to “mobile” users.
- ◆ Mobility == change in how MH accesses the internet
  - Physically move so that access to internet is via a different basestation.
  - Switch network interfaces
- ◆ Continuous connectivity
  - Datagrams for MH must be delivered to its current location
  - Mobility must be transparent to applications
    - ◆ Applications must not die or need to restarted
    - ◆ Performance transparency also desirable
- ◆ Desirable
  - Secure
  - Work across security domains
  - Require no changes to existing stationary hosts

# Mobility in Wireless LANs: Basestation as Bridges



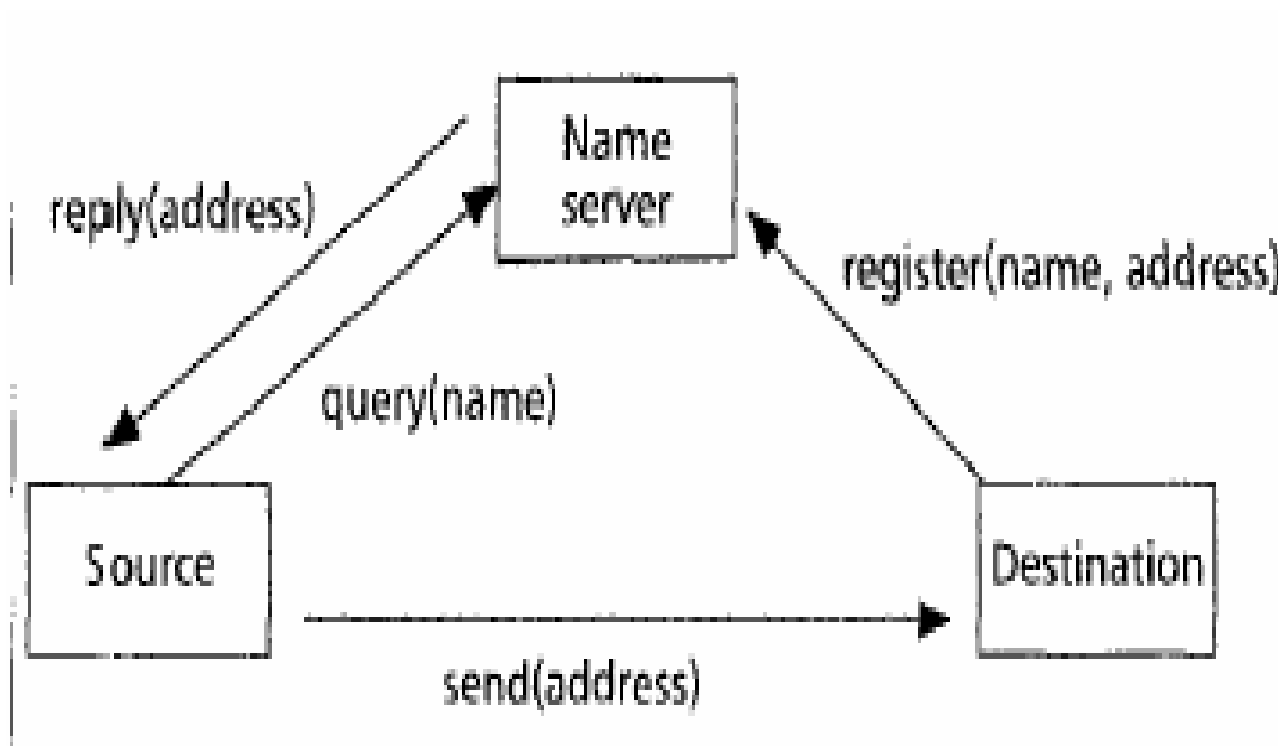
- ◆ Basestations are bridges(layer 2) – i.e. they relay MAC frames
  - Smart bridges avoid wasted bandwidth
- ◆ Works the within an ethernet(or other broadcast LAN)
  - Fails across network boundaries, and in switched LANs(e.g. ATM)



# Internet Naming and Addressing

- ◆ Collection of networks that are connected by routers
- ◆ Each internet host(each network interface) has two identifiers:
  - Internet (IP) Address(32-bit)
  - Host Name (string)
    - ◆ Domain Name System (DNS) maps host names to IP address
- ◆ Applications refer to hosts by names
  - Use Domain Name System (DNS) to map host names to IP addresses
    - ◆ DNS lookup done once only at connection set-up
  - Transport protocols developed that assume this static binding
    - ◆ E.g. a TCP connection is identified by
      - *<Source IP address, source TCP port, destination IP address, destination TCP port>*
- ◆ Packets carry source and destination IP addresses
  - Routers use routing tables to forward packets based on destination address
  - Packet sent directly to destination within a network (e.g. ethernet)

# DNS-based Resolution



# Hierarchical Addressing

- ◆ Routers maintain network topology in routing tables
- ◆ Flat IP address space would make routing tables huge!
  - Many many millions of hosts
- ◆ IP address space is therefore *hierarchical*
  - IP address is a tuple: (*network id*, *host id*)
  - e.g., consider 192.11.35.53

Network id			Host id
192	11	35	53

- ◆ Internet routers required to maintain network topology only at the granularity of individual networks
  - Only network id part of destination address used in routing
  - Makes routing tables manageable

# Key Observation: IP address serves two purposes!



- ◆ Endpoint identifier for transport and application layer
  - MH's IP address must be preserved to retain transport-layer sessions
    - ◆ All TCP connections would die if MH acquires a new IP address
- ◆ Routing directive for network layer
  - MH's IP address must be changed for hierarchical routing to work!
    - ◆ Packets will continue to get routed to the old network
    - ◆ DNS entry will also need to be changed

What should on do?

This is the primary problem in making Internet mobile!

# “Non-solutions” to Internet Mobility

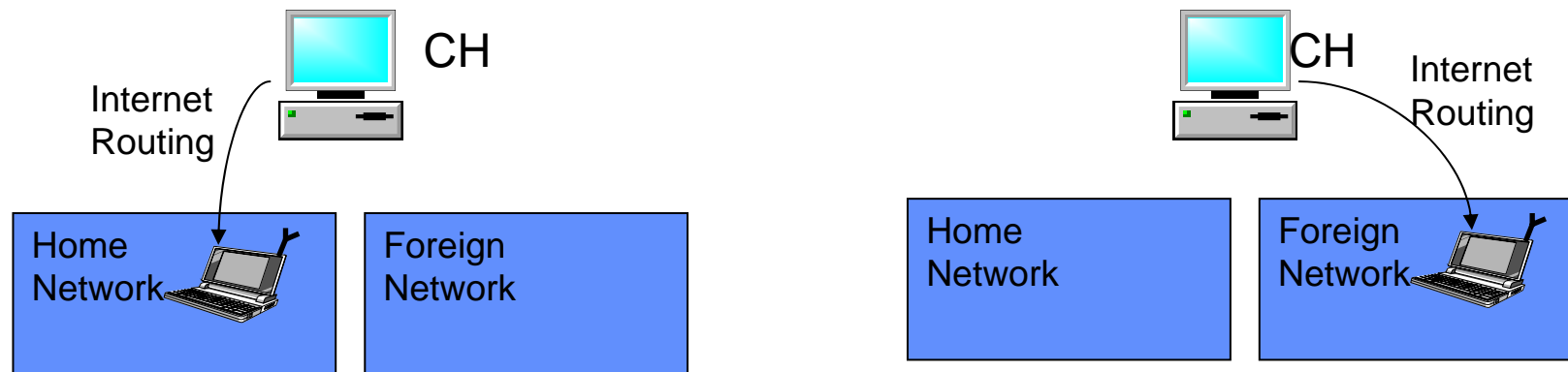
- ◆ Enhance DNS
  - Historically, DNS does not have dynamic *name-address binding* updates
    - ◆ Optimized for access cost
    - ◆ DNS clients cache DNS records
    - ◆ Hard to optimize for both access and update costs
  - Solves only part of the problem
    - ◆ TCP connections will still die!
- ◆ Keep per-MH routing information at all routers
  - Completely breaks the hierarchical routing model
  - Unbounded growth in routing table sizes at all routers
- ◆ Fix all the transport layer and higher protocols, and applications
  - Yeah, sure.....

Clean solutions: fix the network (IP) layer!

# Making IP Network Layer Mobile

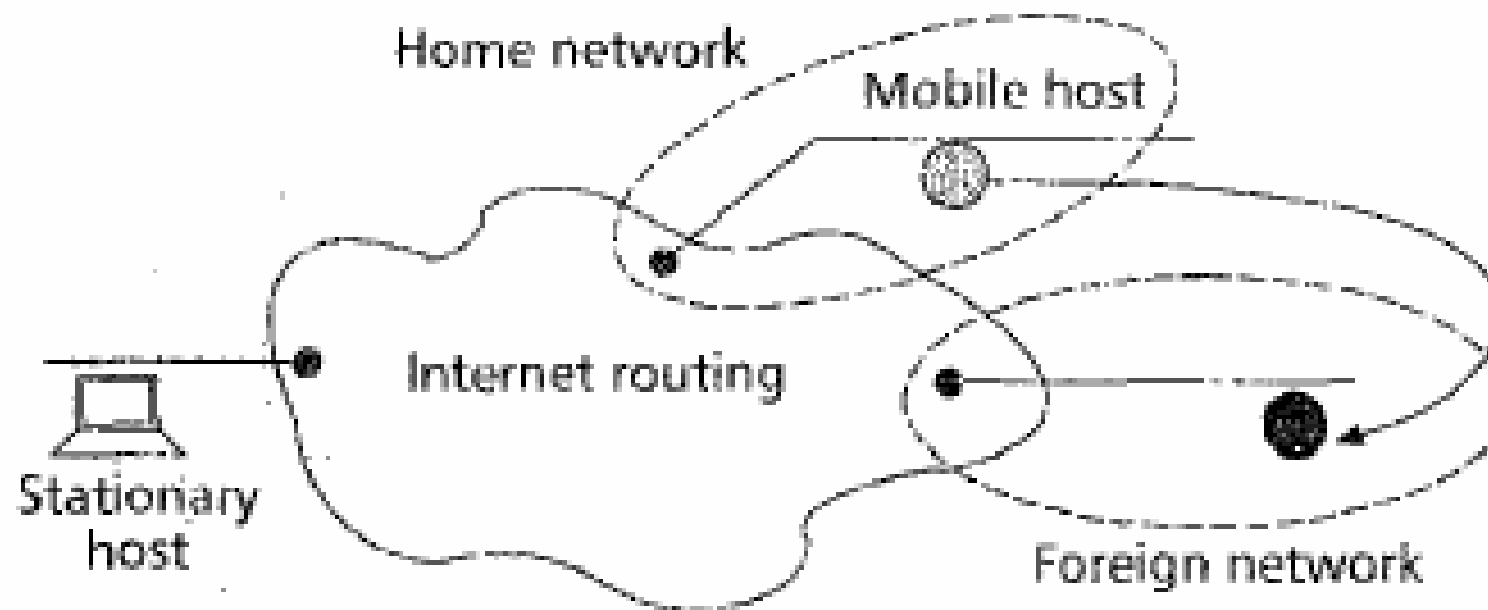
## •Challenge of Mobile-IP

**How to direct IP packets to MH that travels to a Foreign Network away from MH's Home Network?**



- ◆ MH is assigned a home address as its IP address
  - Home network is the network containing the home address
  - DNS queries for MH return the home address
- ◆ Mobile-IP only concerned with moves across networks
  - Moves within home network (e.g. ethernet) handled by link-layer bridging.

# Illustration of terms

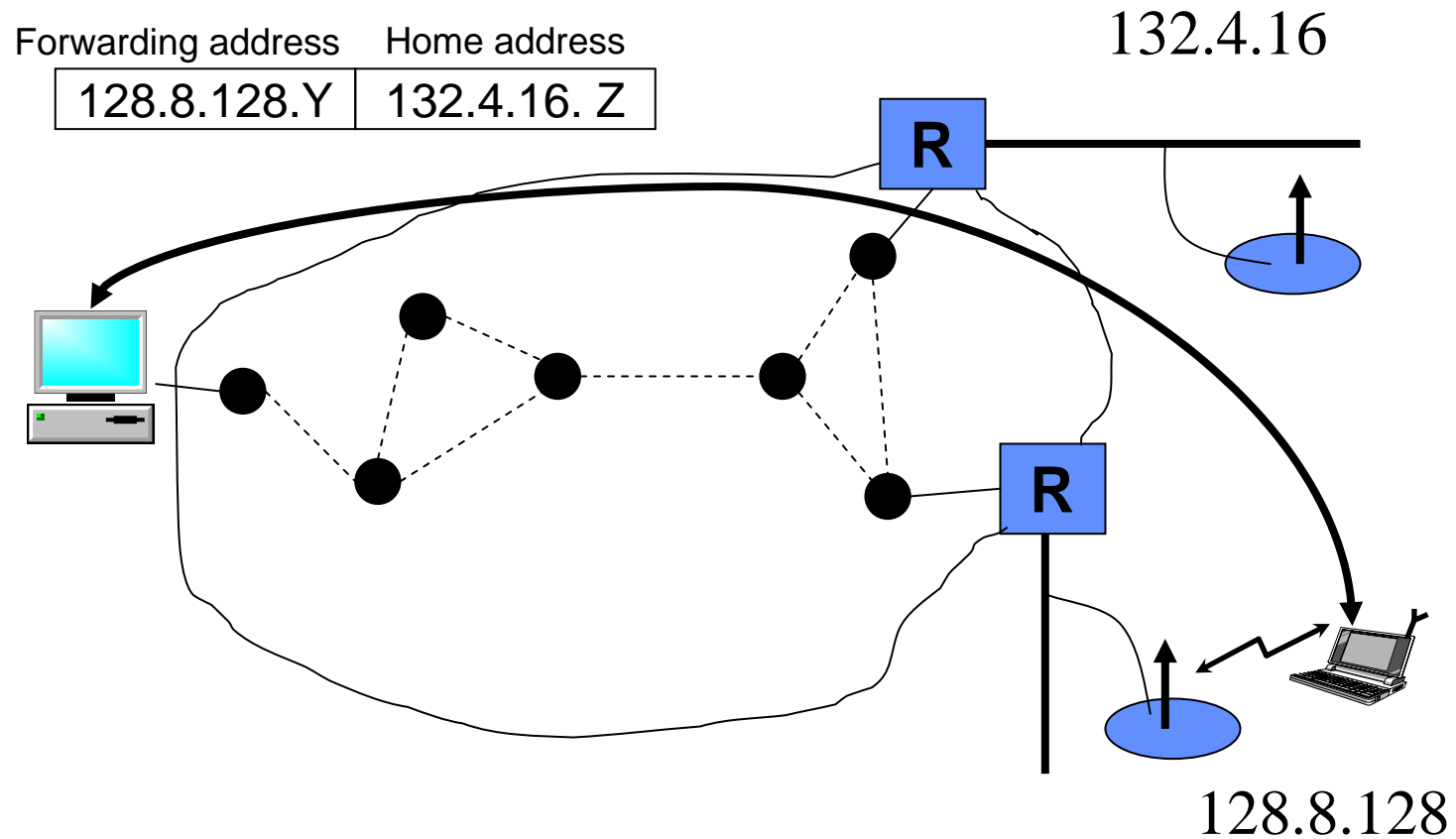


# Key to Mobile-IP Two-Tier Addressing

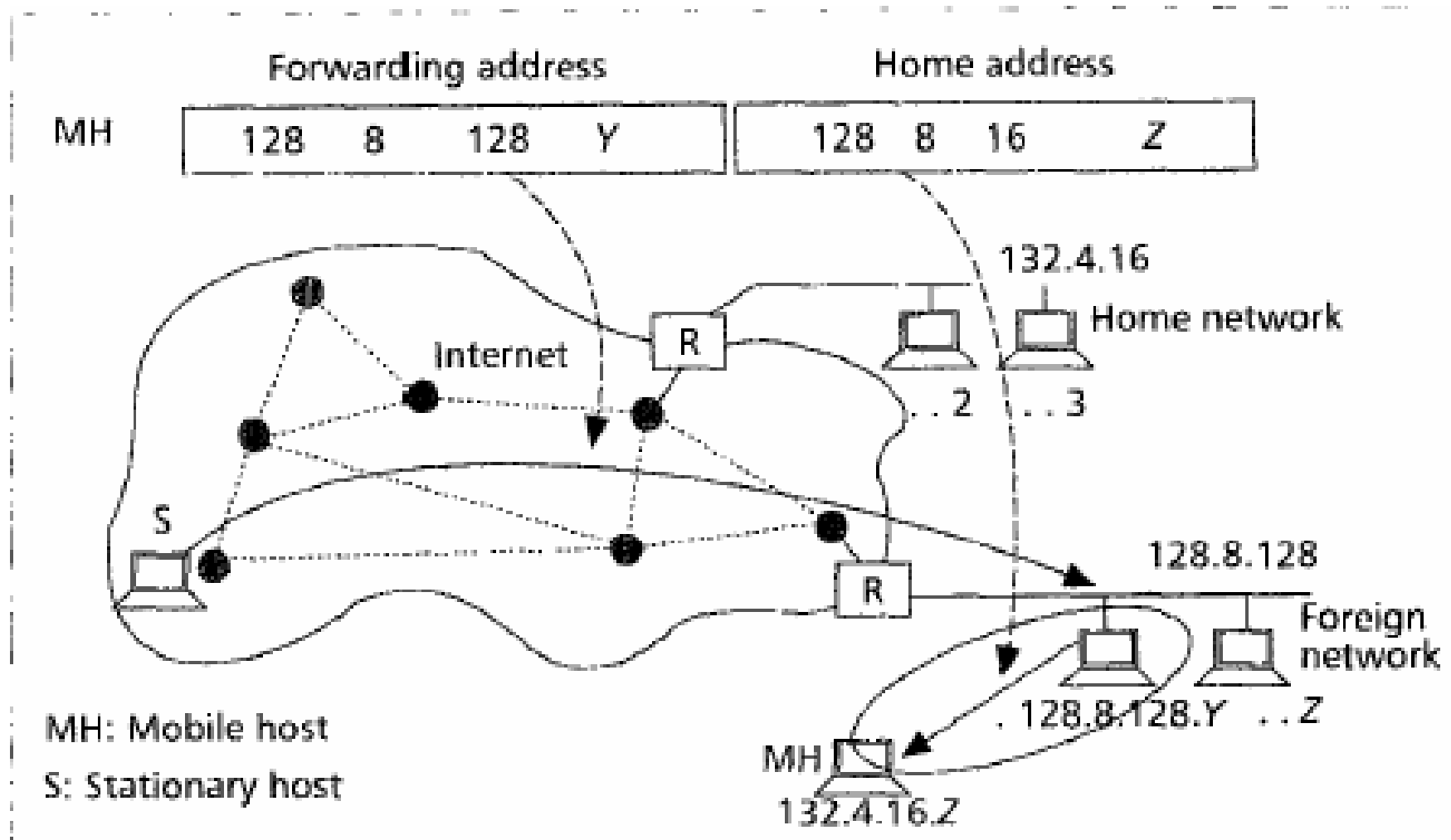
- ◆ MH has two IP addresses associated with it
  - Does not mean two IP addresses are assigned!
- ◆ First component of the address serves as the routing directive
  - Reflects MH's point of attachment to Internet
    - ◆ Derived from the foreign network
  - Changes whenever MH moves to a new network
  - Internet routers use this address to route to MH's point of attachment
- ◆ Second component of the address serves as the end-point identifier
  - This is the home address
  - Remains static throughout the lifetime of MH
  - Only this address used for protocol processing above network layer
    - ◆ MH remains virtually connected to the home network
- ◆ Two-tier addressing is only a logical concept
  - IP packet headers can't actually carry two addresses!
- ◆ MH to Stationary Host (SH) packets do not need special handling



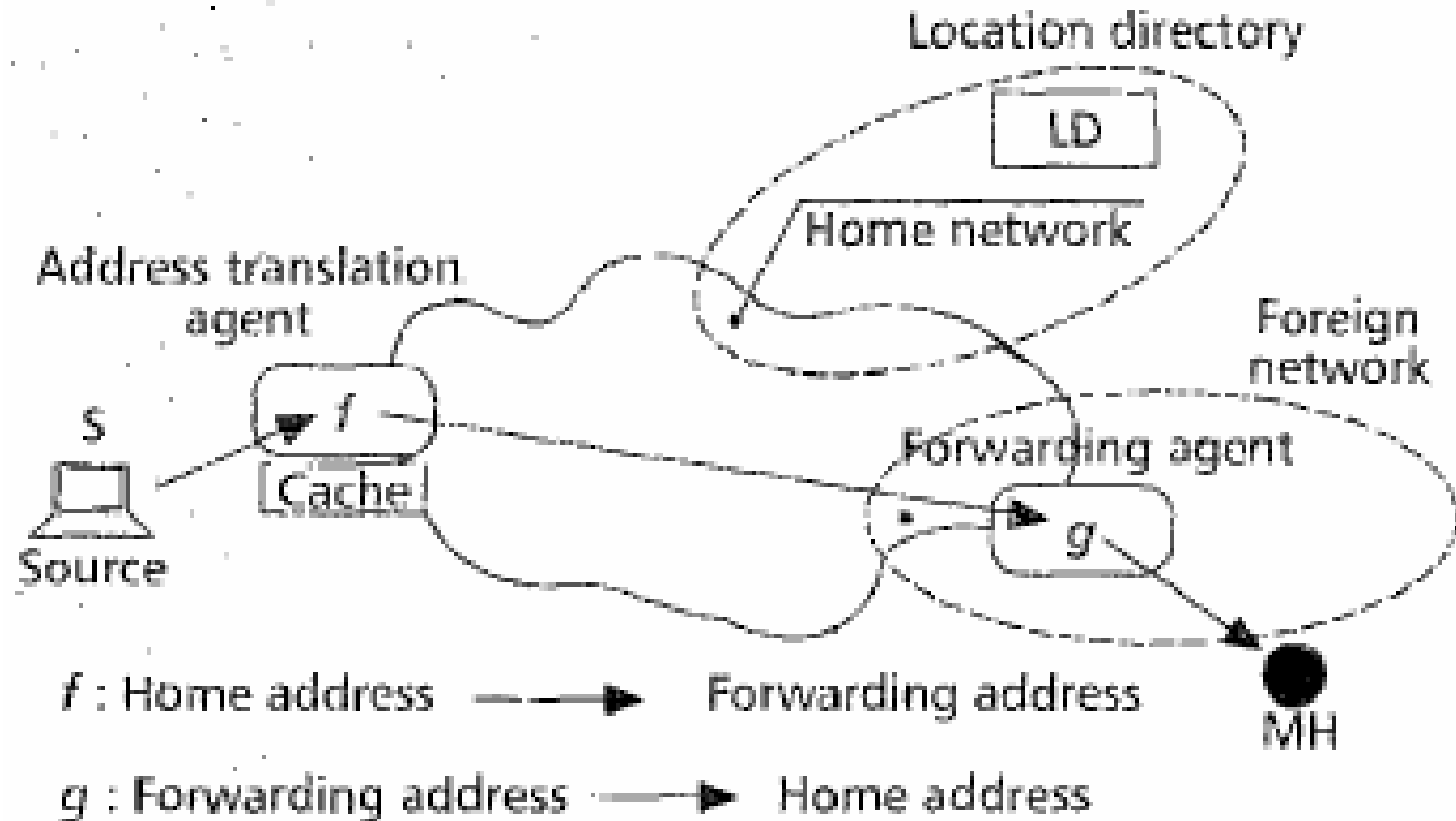
# Two-Tier Addressing for Mobile Hosts



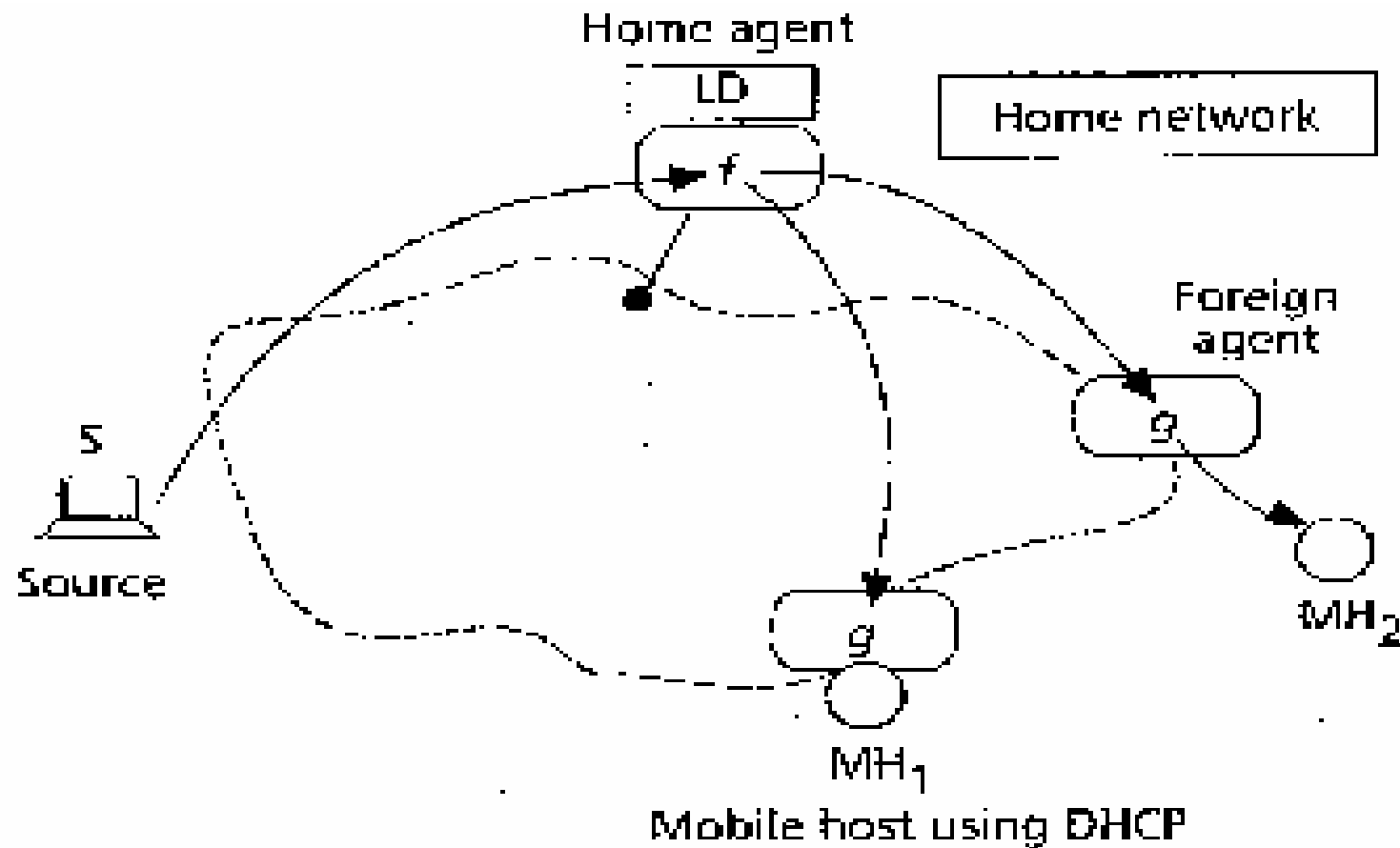
# Typical Example



# Packet Forwarding model



# Canonical Mobile-IP Architecture



# Components of Canonical Mobile-IP Architecture



- ◆ Forwarding Agent (FA)
  - Forwarding component of two-tier address is the address of FA entity
  - FA receives packets on behalf of MH
    - ◆ Packets contain FA's address as destination
  - FA maps forwarding address to MH's home address
    - ◆ FA:  $g(\text{forwarding address}) \rightarrow \text{home address}$
  - FA then relays the packet to MH
  - FA represents a function, not a machine

## Issues:

- Where can FA be located?
  - ◆ MH, BS, somewhere else
- How does MH find the FA in a foreign network? (and, vice versa)
  - ◆ Route advertisement and registration protocol
  - ◆ FA periodically advertises its presence (beacons)

# Component of Canonical Mobile-IP Architecture (contd.)



- ◆ Location Directory (LD)
  - Records association between home and forwarding addresses
    - ◆ Contains most up to date mapping of MH to its FA
  - MH sends updates to LD on moving
  - Issues:
  - Centralized vs. distributed realization
    - ◆ Centralized is infeasible – too many MHs in the Internet
  - How to distribute?
    - ◆ Cost operation
    - ◆ Security
    - ◆ Ease of location
    - ◆ Ownership
  - Possible distribution policy: *owner-maintains*
    - ◆ Some agent in home network maintains LD information for a MH responsible for security, authentication, updates, and distribution
    - ◆ a CH does not need to find the right LD component to query router in home network can forward to the correct LD component

# Component of Canonical Mobile-IP Architecture (contd.)



- ◆ Address Translation Agent (ATA)
  - CH sends packets to MH at its home address
  - ATA replaces MH's home address with FA's address in packets
    - ◆  $ATA: f(\text{home address}) \rightarrow \text{forwarding address}$
  - address translation involves:
    - ◆ Querying the LD
    - ◆ Obtain address of the FA corresponding to the MH
    - ◆ Use FA's address to forward packet to MH's location
  - Issues:
    - ◆ Where to locate ATA
      - At CH: but will need to change software in millions of hosts! elsewhere
    - ◆ Querying LD for every packet is expensive: cache LD entries?
      - Improves performance
      - but, requires maintaining consistency between LD and cached entries!

# Location Update Protocol (LUP)

- ◆ LUP is the reliable mechanism for
  - Keeping LD up to date
  - Keeping cached LD entries consistent with master LD
- ◆ Choice of LUP depends on caching policy
  - Together they determine scalability and routing characteristics
- ◆ What if no LD caching
  - ATA must be collocated with LD to avoid per-packet queries
  - Packets from CH will first travel to home network before being sent to FA no optimal paths!
- ◆ What if there is caching?
  - Routing efficiency is improved no more travel to home network
  - but, vulnerable to security attacks cache updates must be authenticated otherwise, traffic to MH may be redirected away!

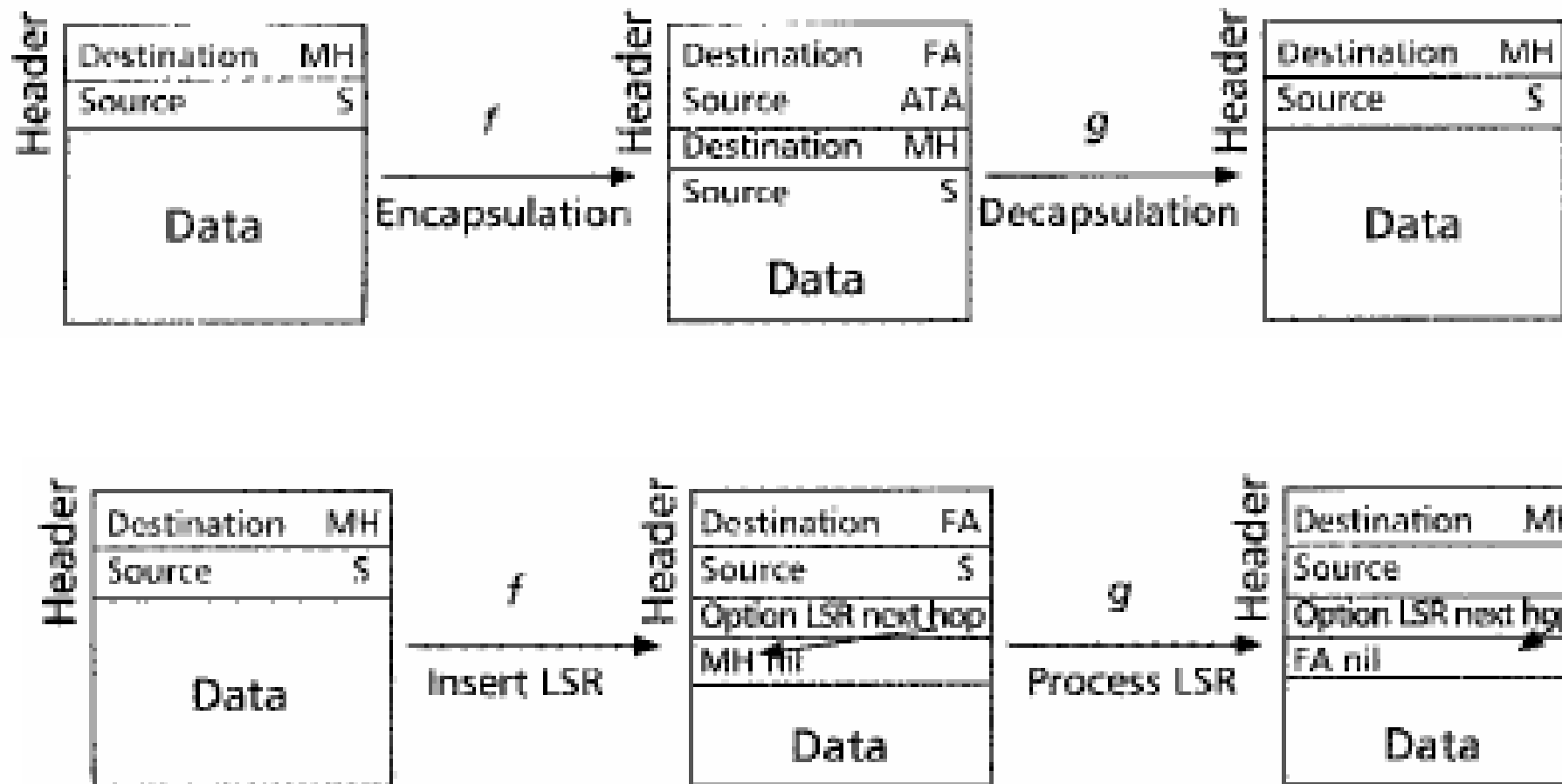


# Address Translation Mechanisms

- ◆ Encapsulation approach (IP-in-IP tunnel)
  - ATA appends new header at the beginning of datagram
  - Outer header contains the forwarding address
  - Inner header contains the home address
  - Internet routes according to outer header
  - FA strips the outer header and delivers datagram locally to MH



# ATM (Address Translation Mechanisms)



# Address Translation Mechanisms (contd.)

- ◆ Loose Source Routing approach
  - Option in IP packets to specify a sequence of IP addresses to follow path is automatically recorded in the packet destination can send reply back along reverse path
  - ATA can use LSR to cause packets to MH to be routed via FA co-locate ATA at CH, and FA at MH
    - ◆ MH sends to CH using LSR, ATA/CH reverses the path

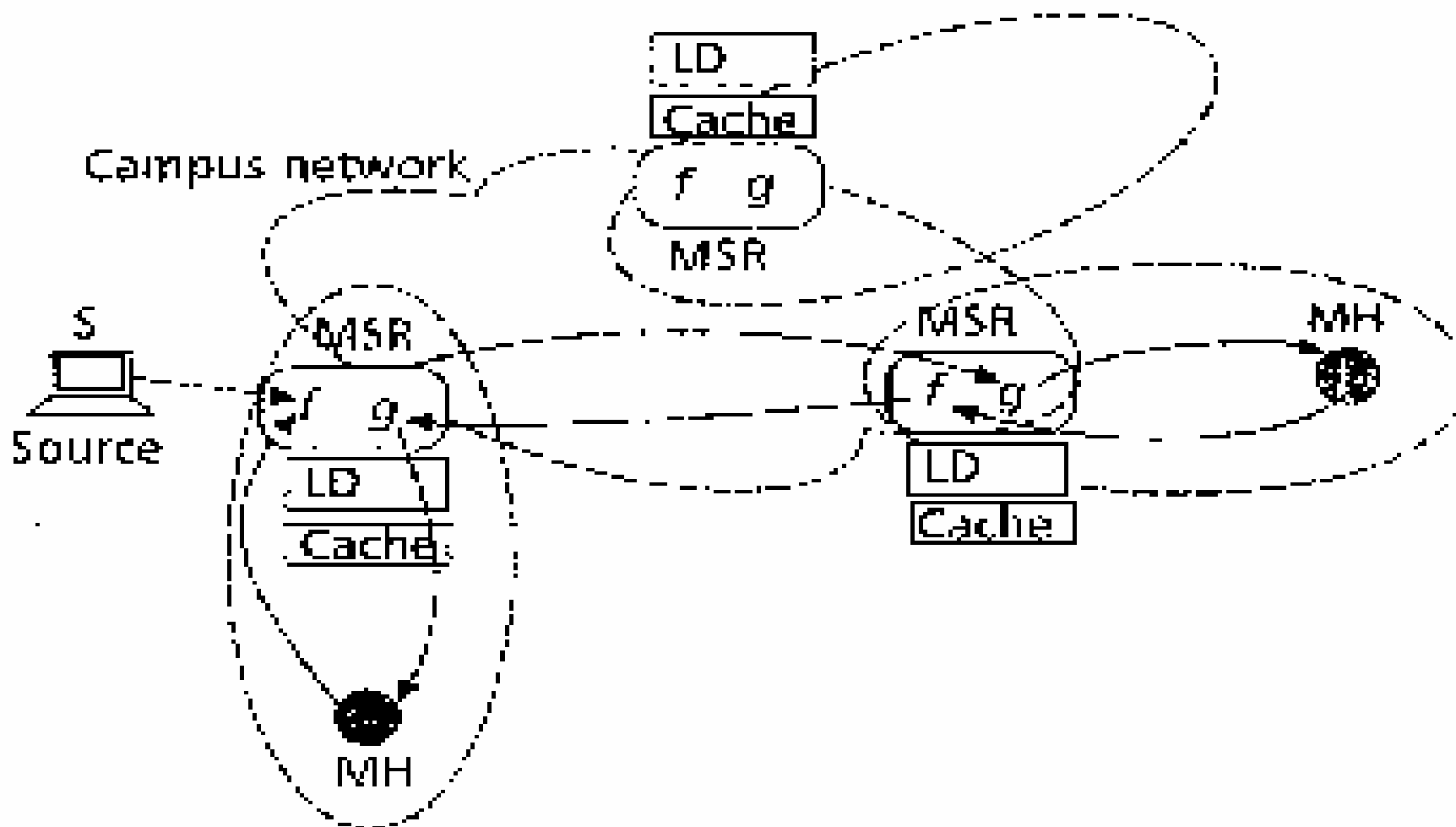
# Various Mobile-IP Proposals

- ◆ Many Mobile-IP systems have been proposed (and some implemented)
  - Columbia's Mobile-IP
  - Sony's Virtual (VIP)
  - IBM's LSR Scheme
  - Stanford's MosquitoNet Scheme
  - IMHP (Internet Mobile Host protocol)
  - IETF's Mobile-IP for IPv4
  - IETF's Mobile-IP for IPv6
  - etc.
- ◆ All are special cases of the canonical mobile-IP architecture
  - Make different choices of
    - ◆ FA location
    - ◆ ATA location
    - ◆ Choice of LUP address translation mechanism

# Example: Columbia's Mobile IP

- ◆ Campus environment with a reserved subnet for MHs
  - MHs home address are from the reserved subnet
- ◆ Group of cooperating Mobile Support Routers (MSR)
  - MSRs advertise reachability to wireless subnet via beacons
  - MHs connect to campus backbone through MSRs
  - MSRs forward traffic to/from MHs
- ◆ On moving, MH registers with the new MSR
  - New location is provided to the previous MSR
- ◆ CH sends packet to MSR closest to CH
  - This MSR either delivers the packet or, forwards it to the right MSR after encapsulation
  - Right MSR is located by a multicast WHO\_HAS query to other MSRs
- ◆ Wide area operation uses a pop-up mode
  - A temporary address is used by MH as a forwarding address
  - MH does its own encapsulation/decapsulation

# Columbia Proposal



# Columbia's Mobile-IP Mapped to Canonical Architecture

- ◆ MSR performs both encapsulation & decapsulation
  - Both f and g are collocated at MSR
  - MSR acts as FA for MHs in its coverage area
  - MSR acts as ATA for packets addressed to other MHs
- ◆ LD is distributed realization of the owner-maintains scheme
  - Each MSR maintains a table of MHs in its coverage
  - MSRs are a distributed realization of home router
  - Tables of MHs in MSRs together constitute an owner-maintained LD
- ◆ Caching policy for LD entries is “need-to-know”
  - MSR sends WHO\_HAS query if it does not know MH's location
- ◆ LUP is lazy-update
  - When MH moves, only primary and previous copy of LD entry is updated
  - Cached entries are assumed correct by default
  - Stale cache entry causes packet delivery failure, triggering WHO\_HAS
- ◆ 100% backward compatible – no existing internet entities are affected

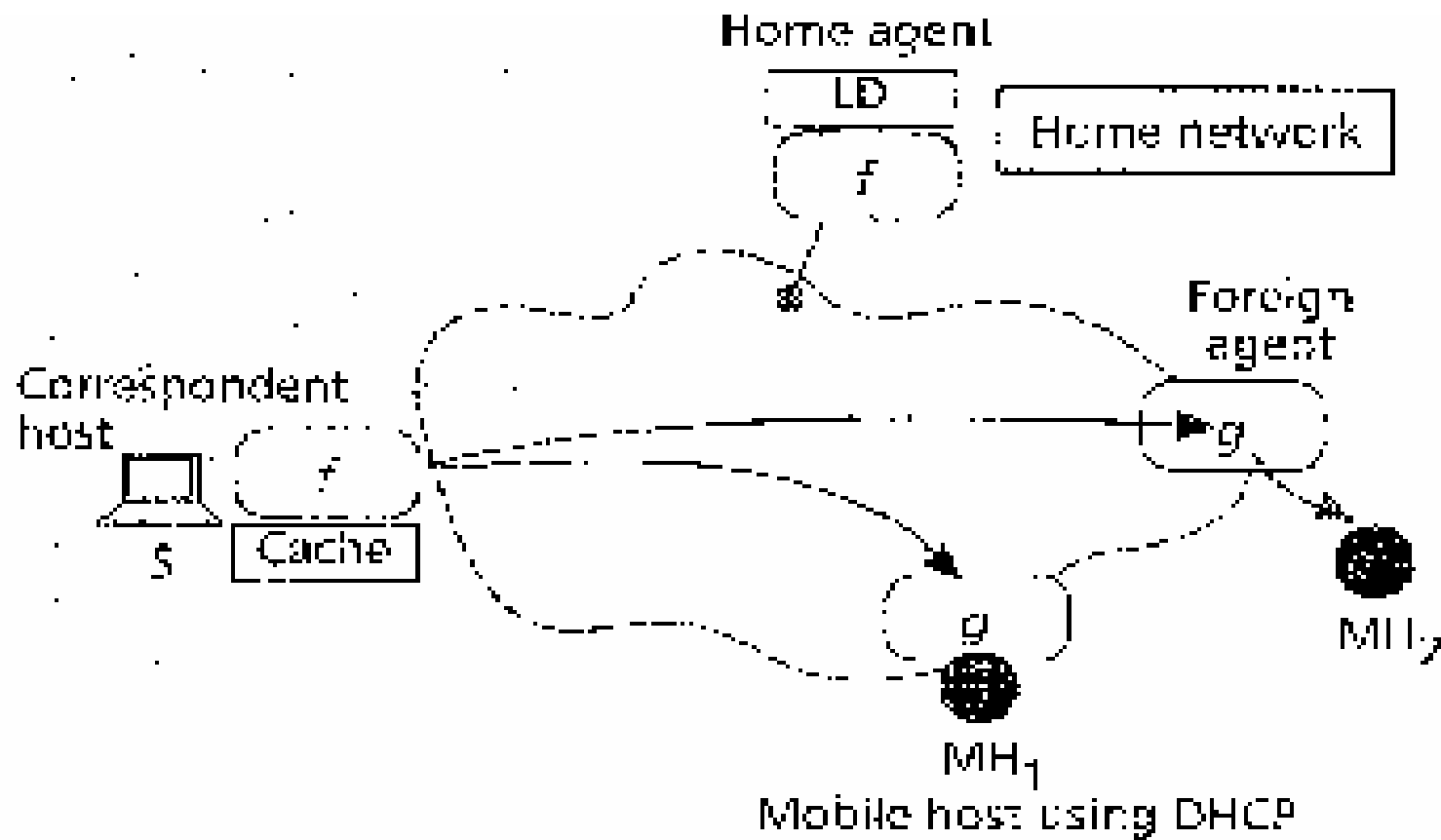
# Performance Characteristics of Columbia Mobile-IP



- ◆ Control
  - LD cache at ATA is updated when packet routing is needed
  - Limits control traffic
  - But, slow “first” packet due to WHO\_HAS query results in SYN packet being lost in TCP (start of transmission)
- ◆ Overhead of IP-in-IP
  - 20 bytes (4% on 500 byte packets)
- ◆ Routing
  - Requires routing to nearest MSR to be optimal
  - Not optimal for pop-up mode
- ◆ Implementation on 33 MHz 486 based MSRs
  - 1.4 ms for WHO\_HAS
  - 45 microseconds for encapsulation (per packet overhead)



# Route Optimization



# Route Optimizations



**Figure 4. Behavior when CH is Close to MH**



**Figure 5. A Smart Correspondent Host.**

# Security Issues

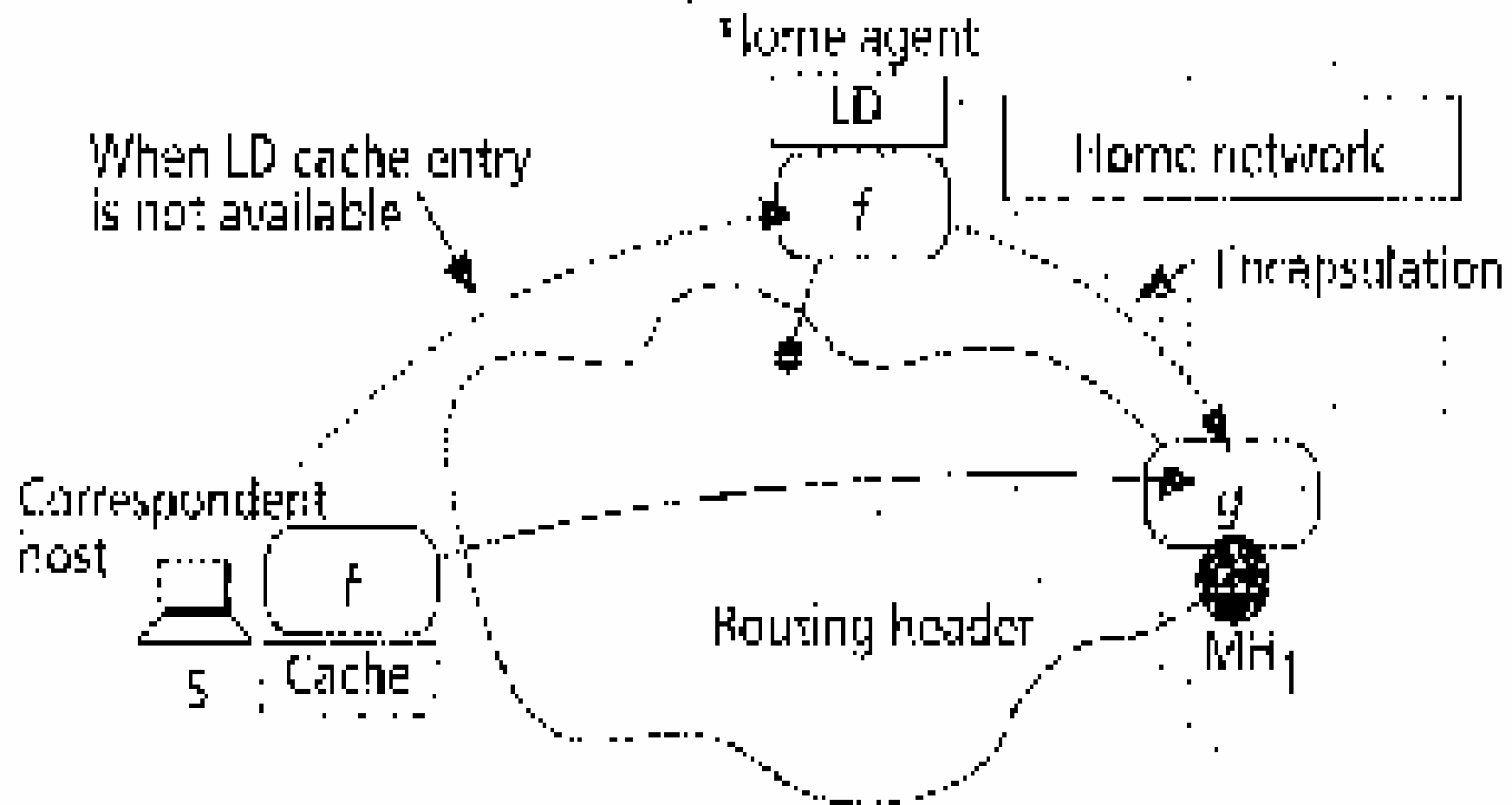


**Figure 2. Problem with Source Address Filtering**



**Figure 3. Bi-directional Tunneling**

# IPv6 Mobility Proposal

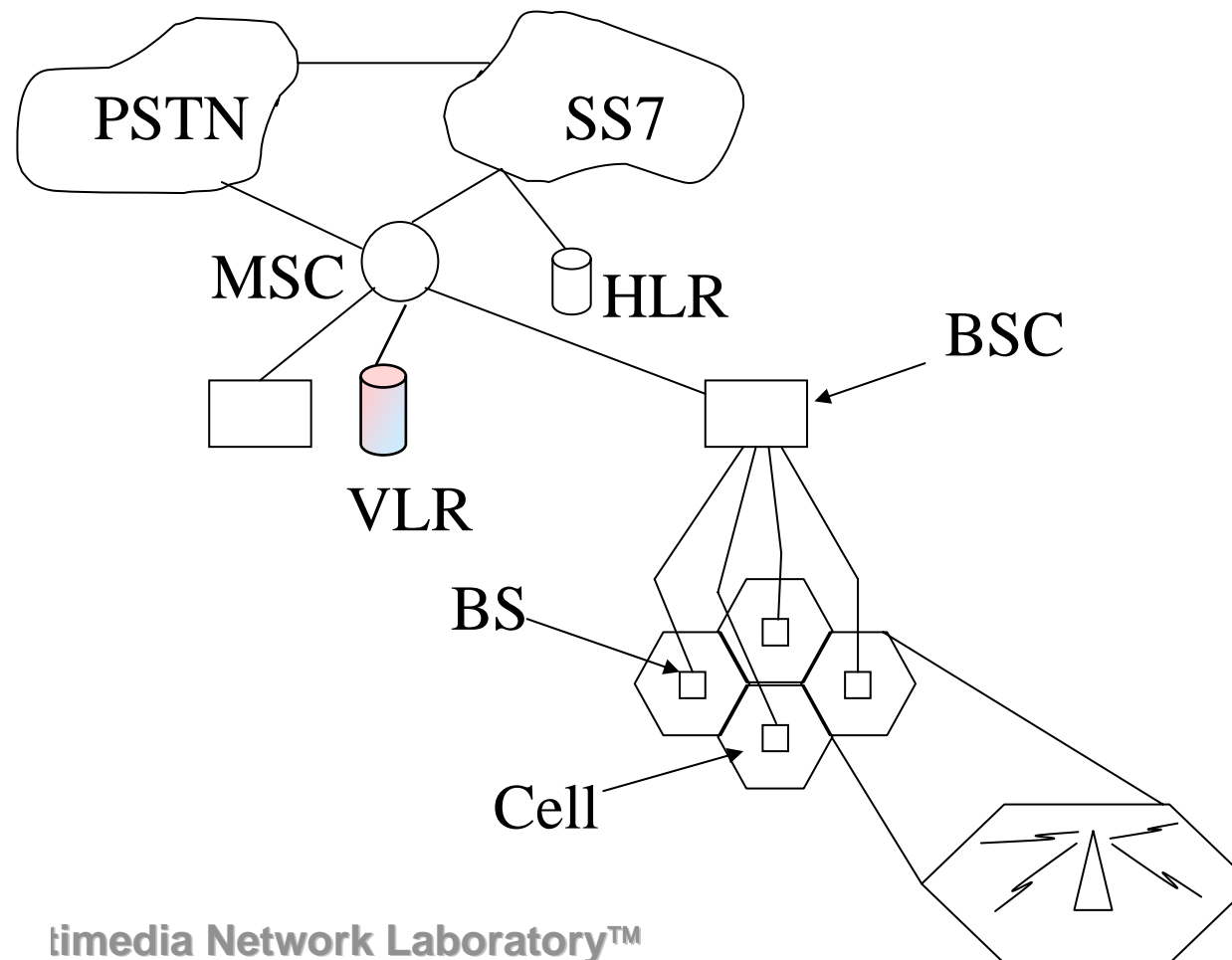


# Evolutions of PCS

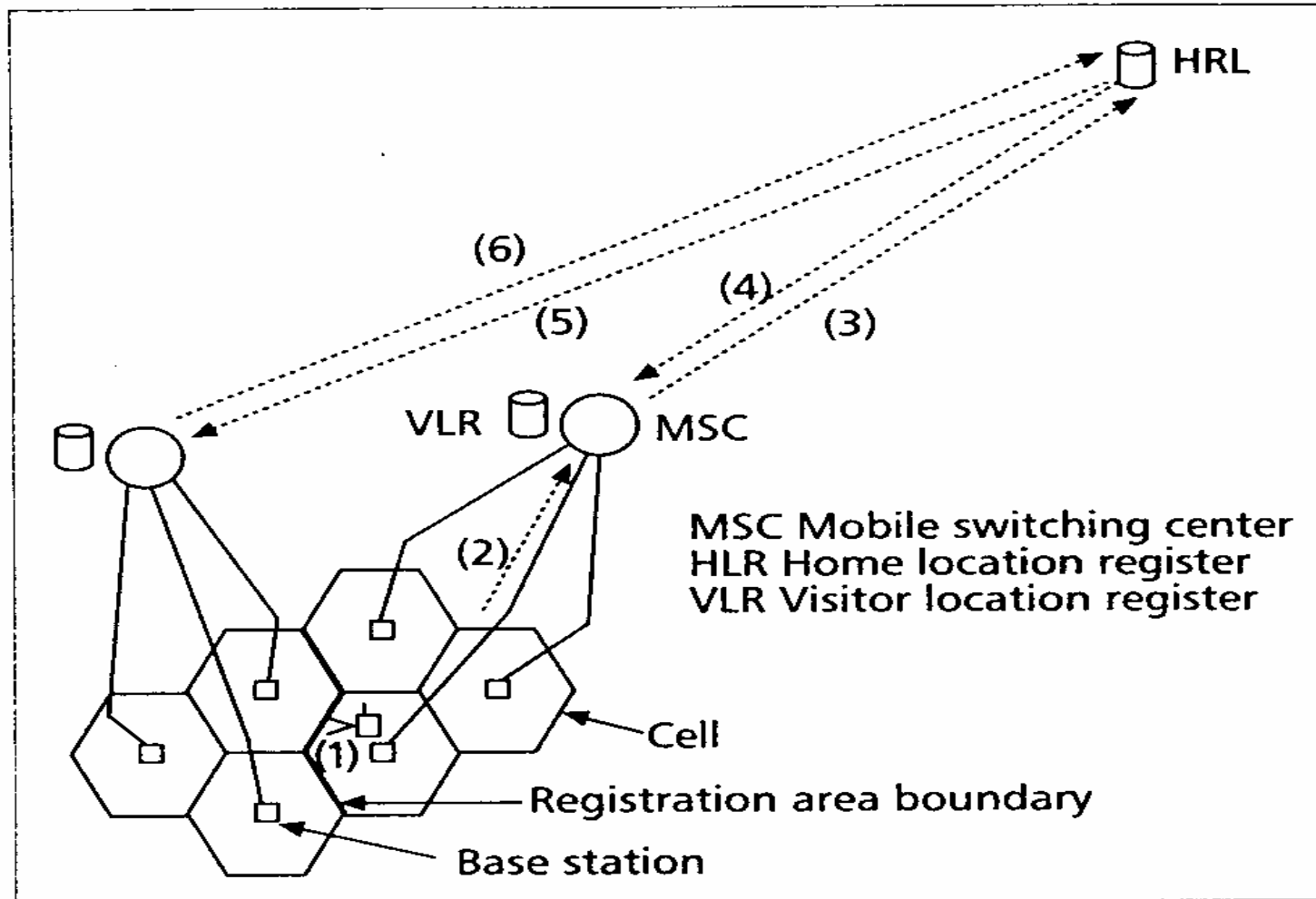


PCS Requirements

# PCS network architecture

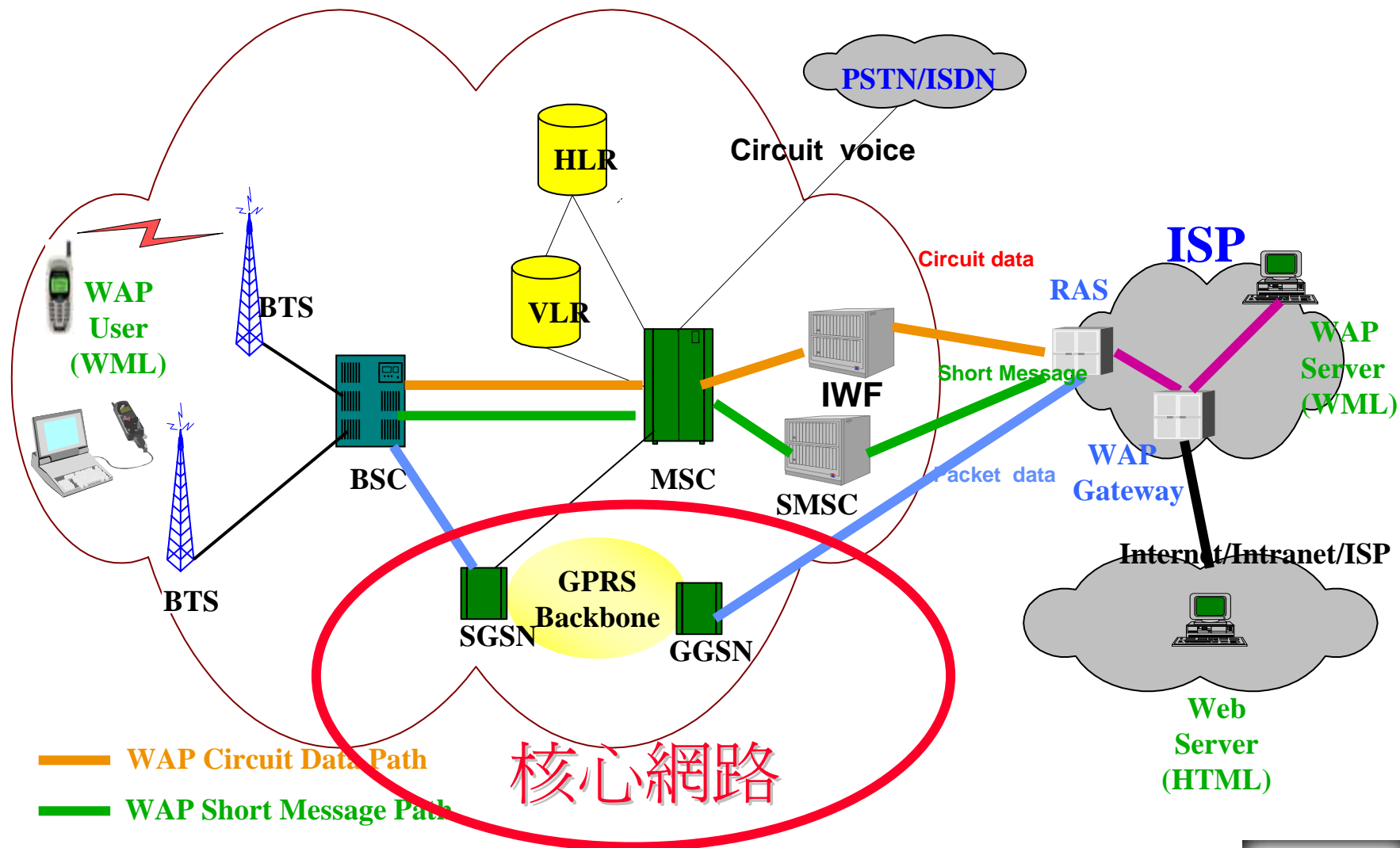


# Location Update Procedure



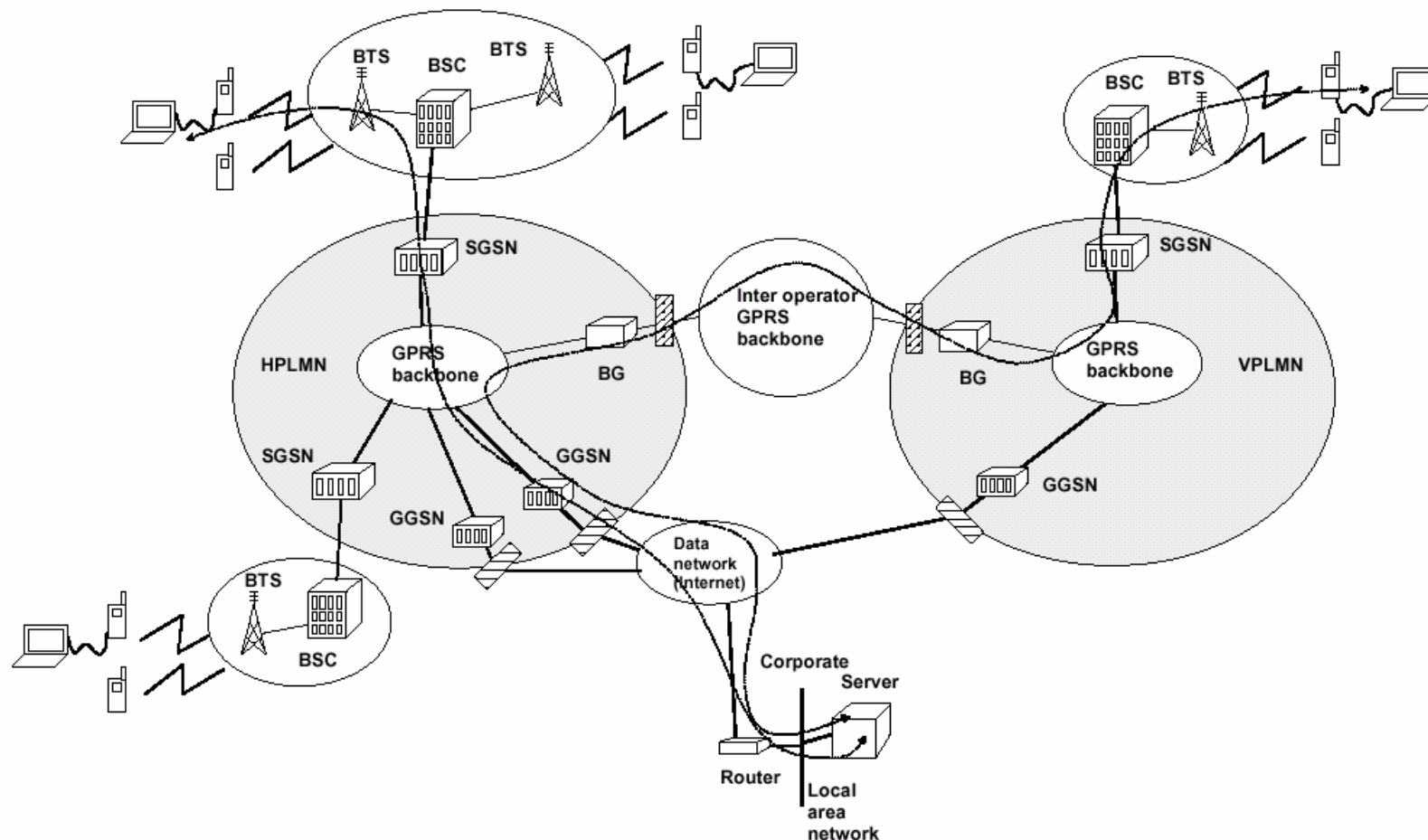
■ Figure 3. *Location registration procedures.*

# GPRS

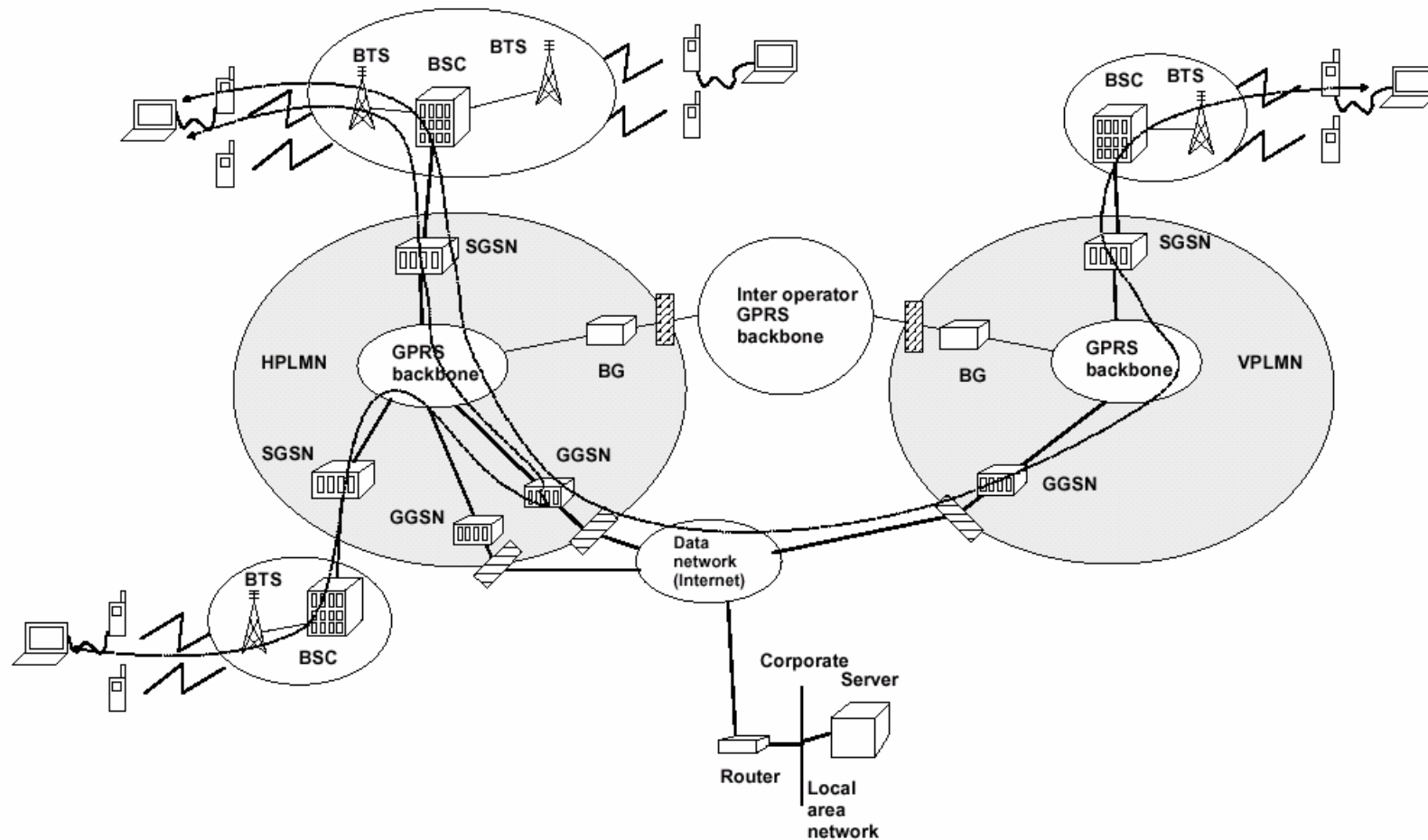




# Data transfer MS-fixed



# Data transfer MS-MS



# Coming Challenges for IP



Location Managements~ handoff, roaming

QoS Transport~ Backbone delivery

# Mobility

## ◆ User mobility

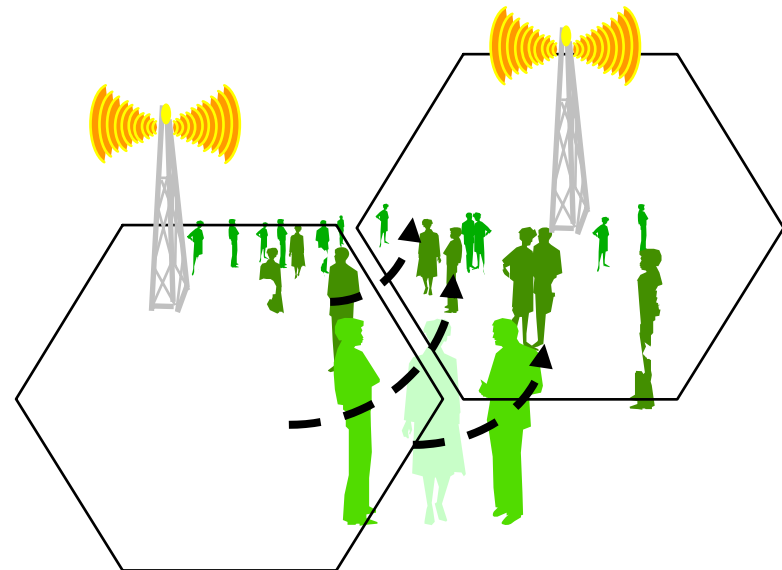
- Micro
- Macro



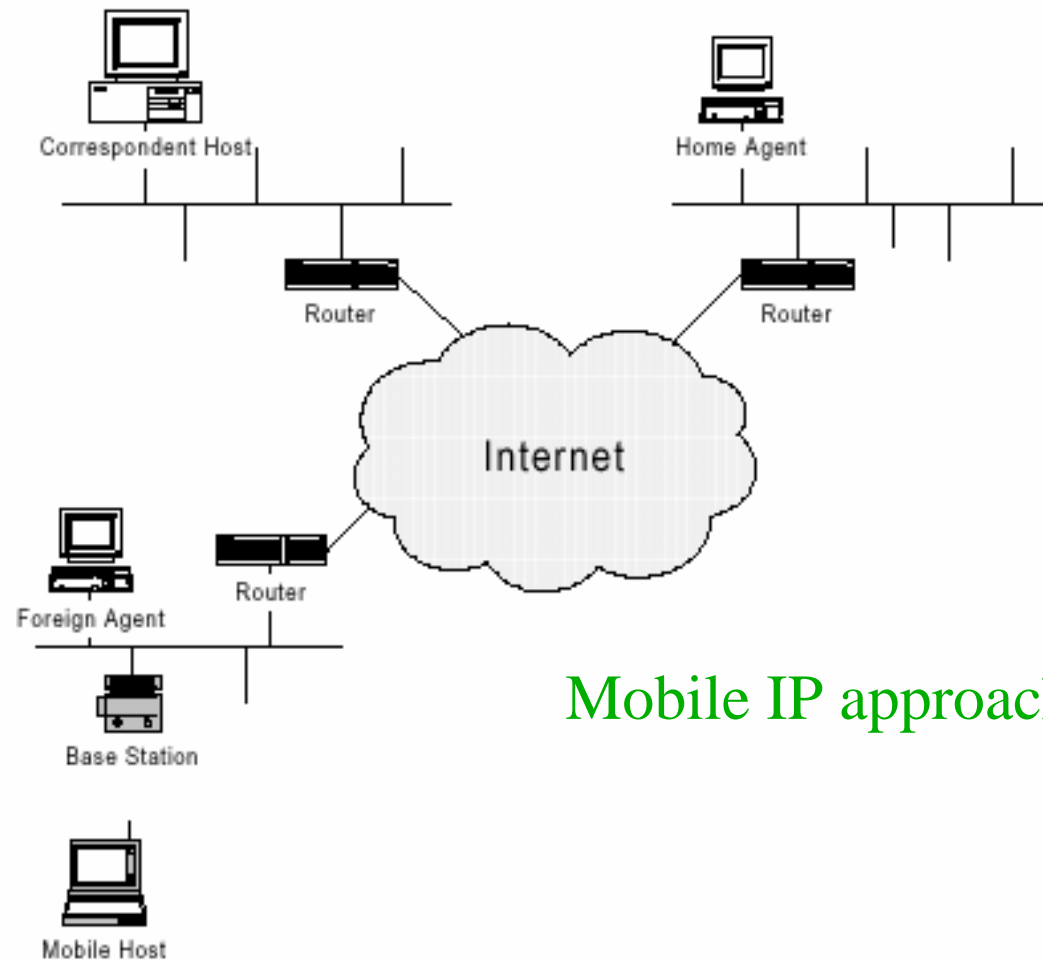
- Handoff issue
- Location management
- Paging

## ◆ IP mobility support

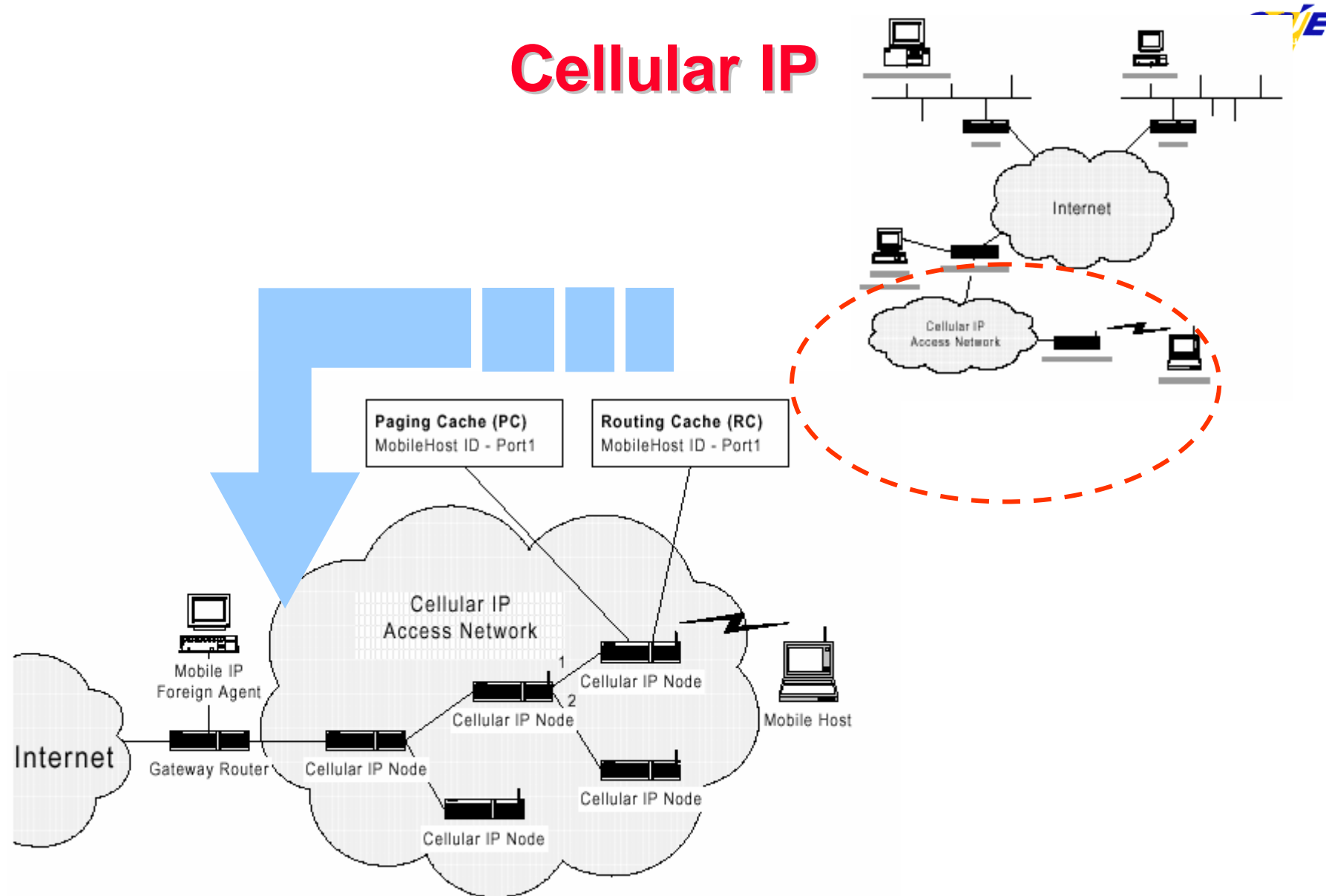
- Mobile IP
- Cellular IP
- HAWAII



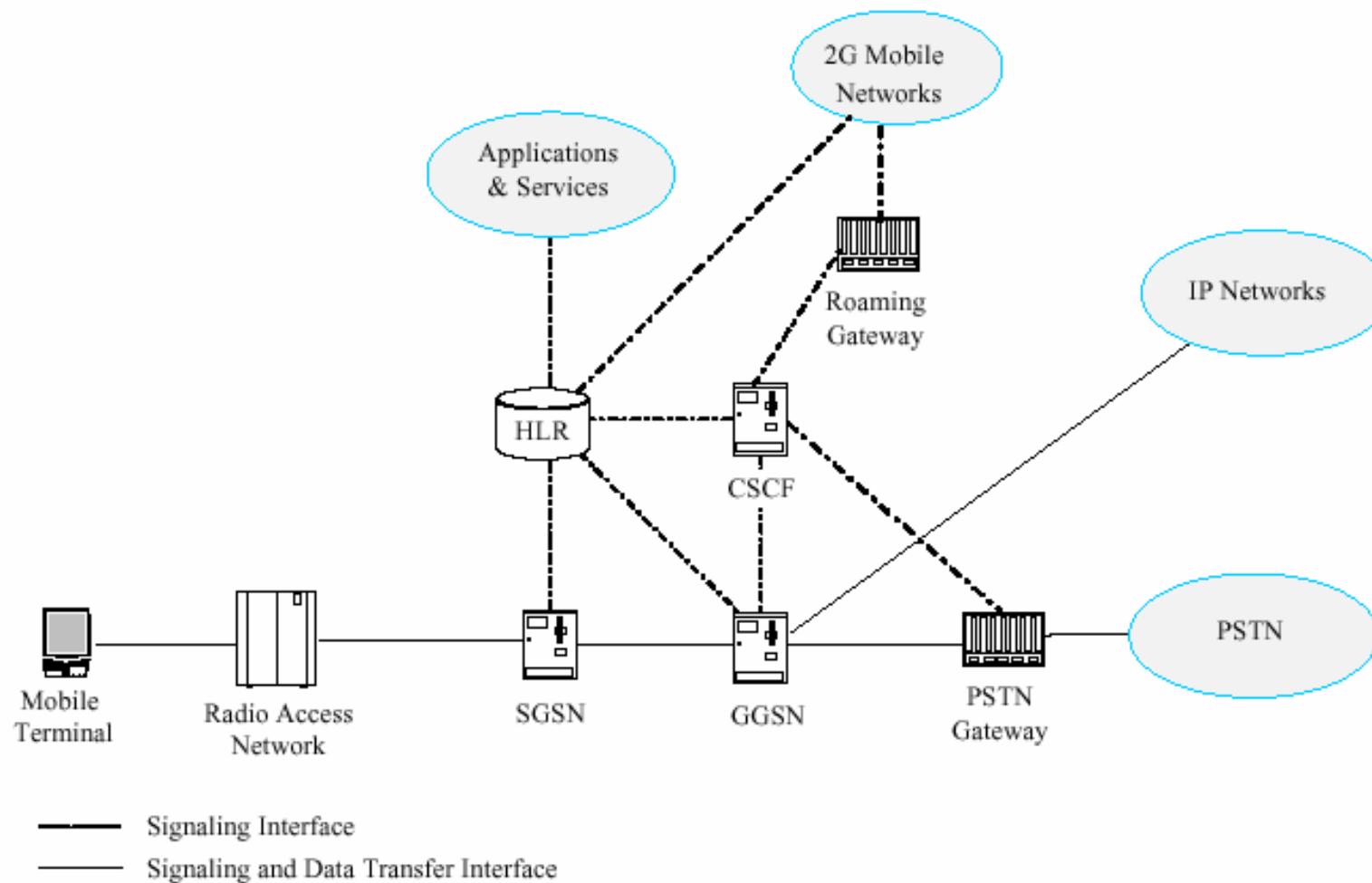
# Nomadic wireless access



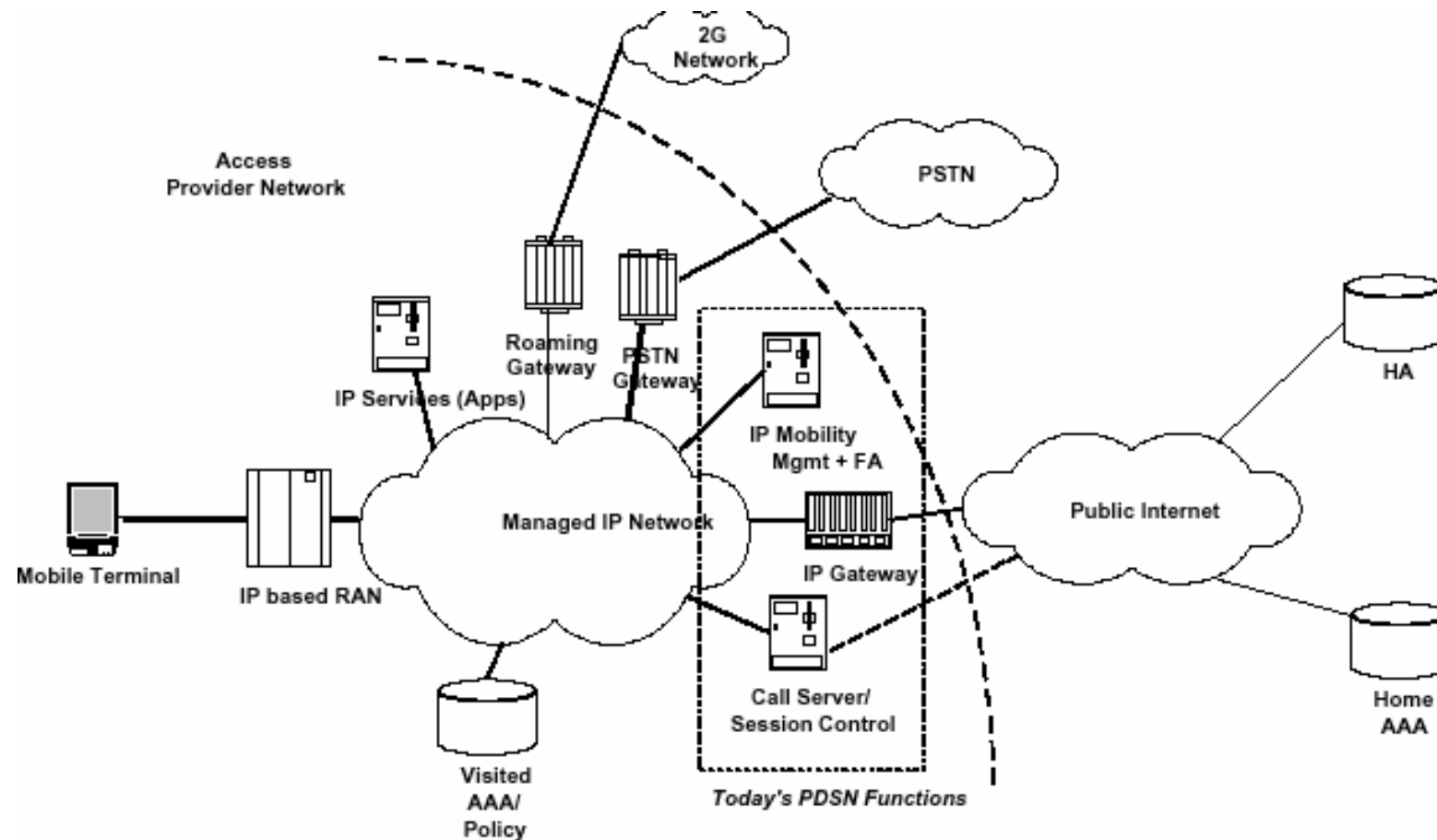
# Cellular IP



# 3GPP IP reference architecture

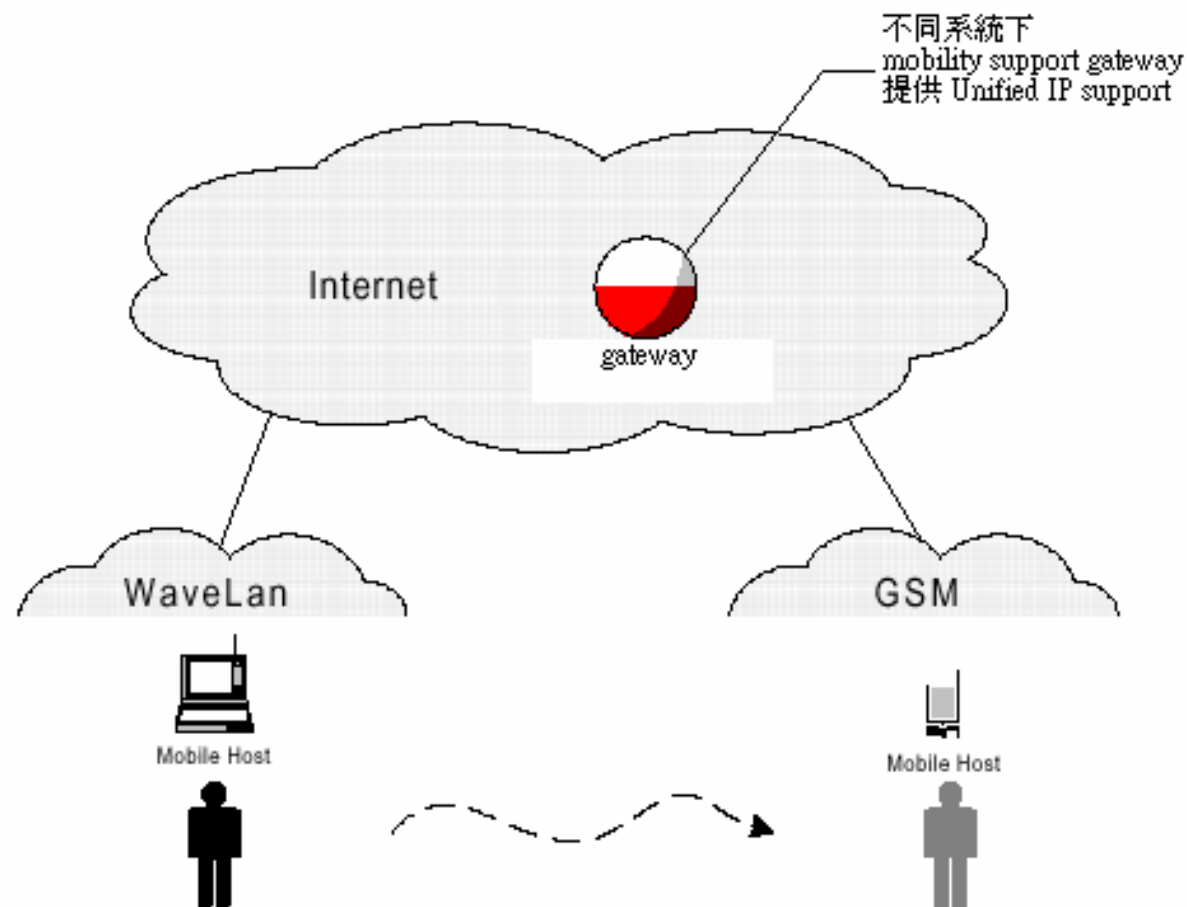


# 3GPP2 IP reference architecture



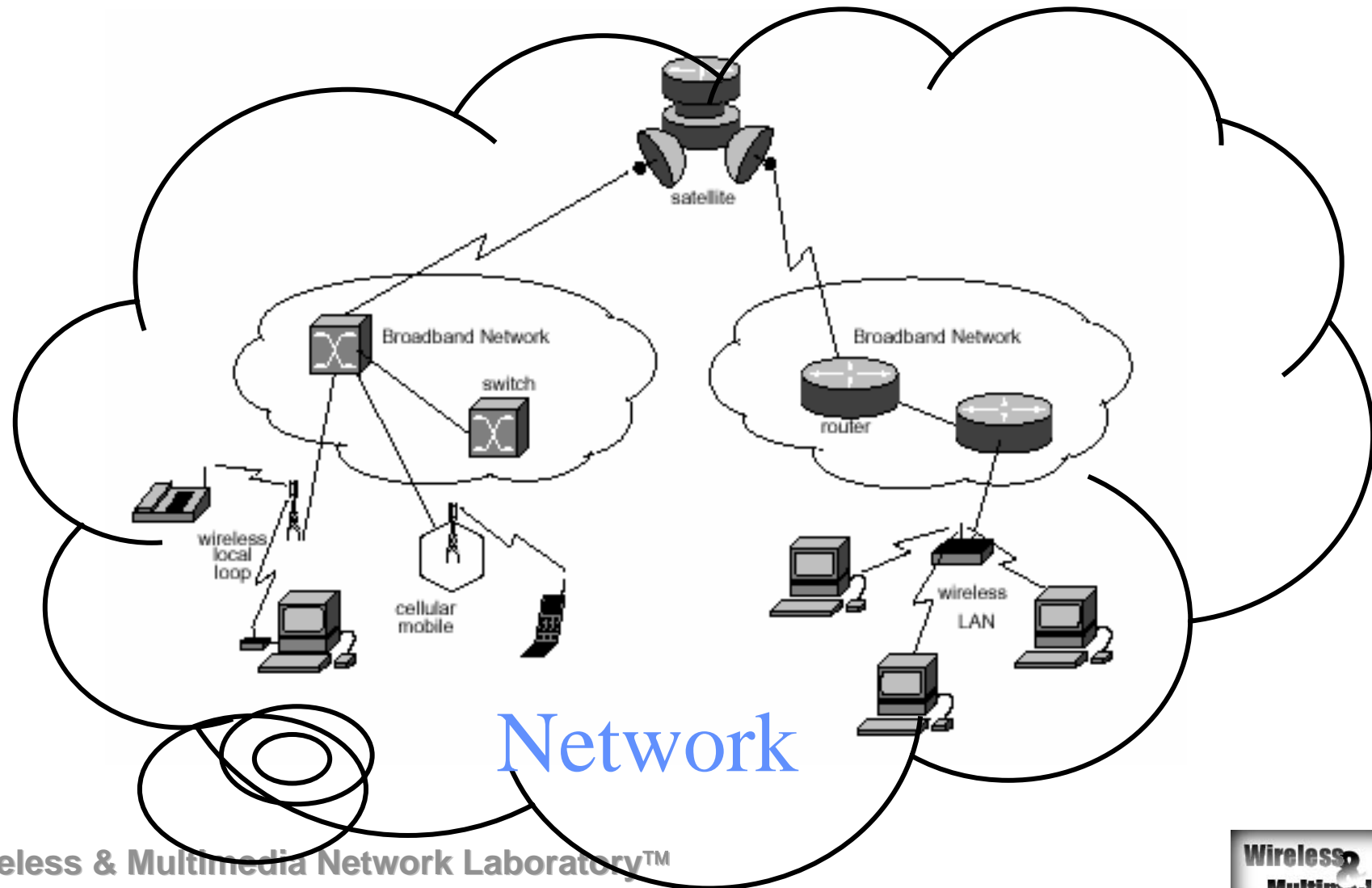


# Heterogeneous access network



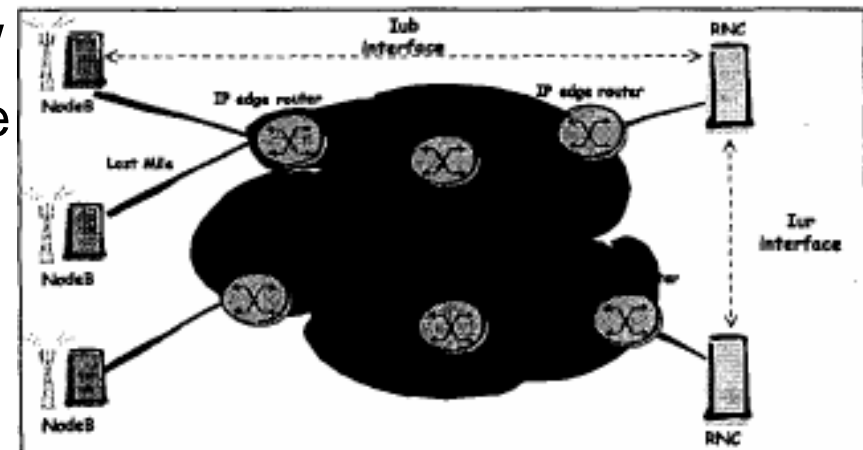
The Mobile People network architecture

# Heterogeneous End System



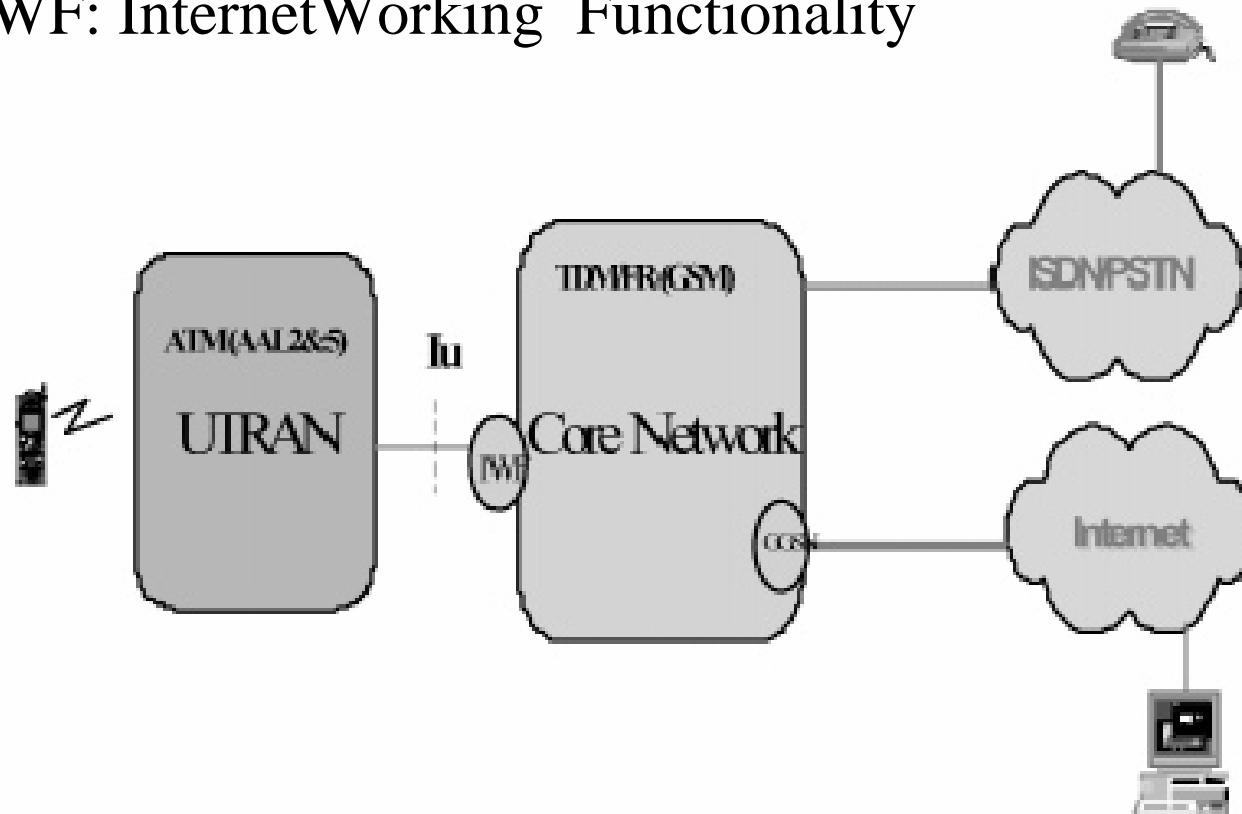
# Last Mile QoS Issues

- ◆ Last mile connect NodeB and RAN. It is usually low bandwidth links.
- ◆ limit the transmission time for a packet.
- ◆ Three choices
  - Fragmentation on a layer below
  - Fragmentation on a layer above
  - Fragmentation in IP Layer

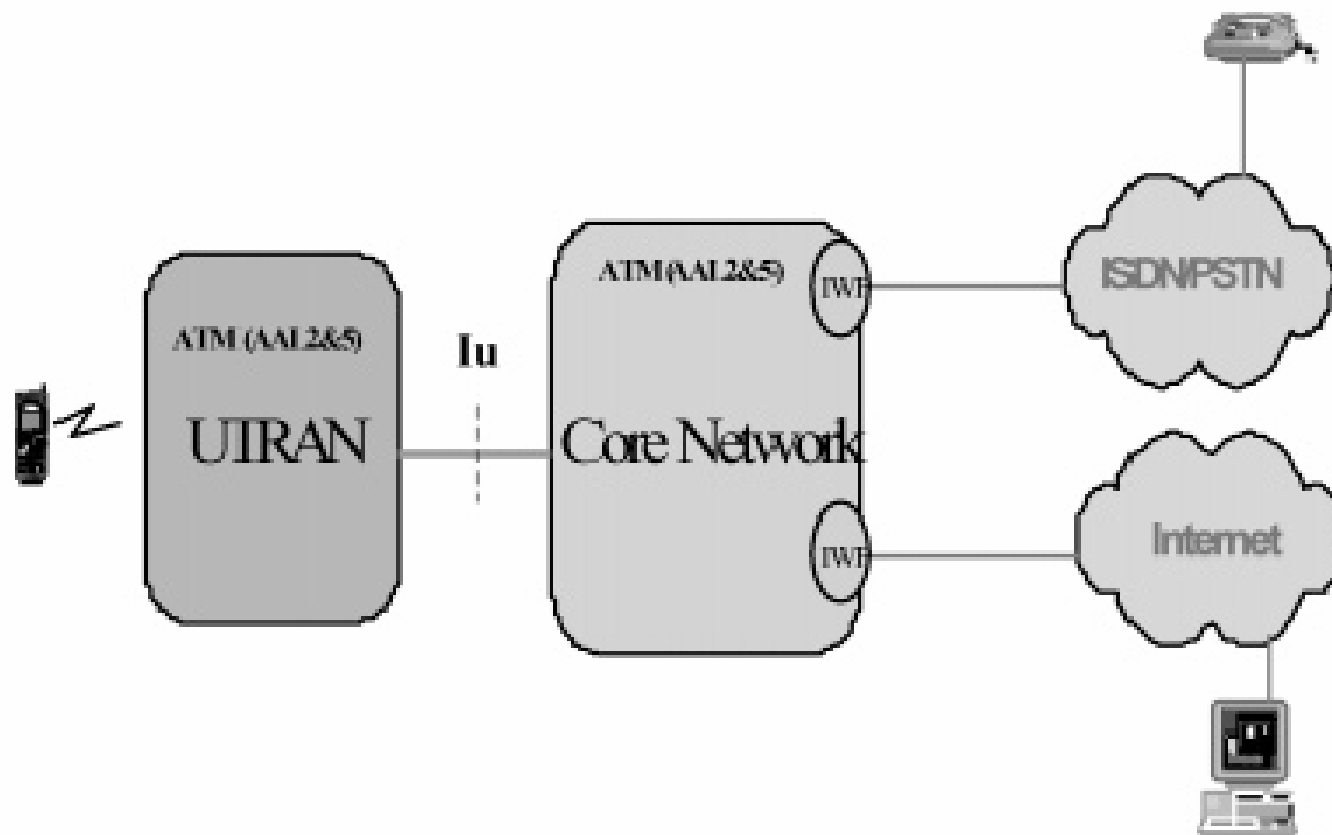


# Option1

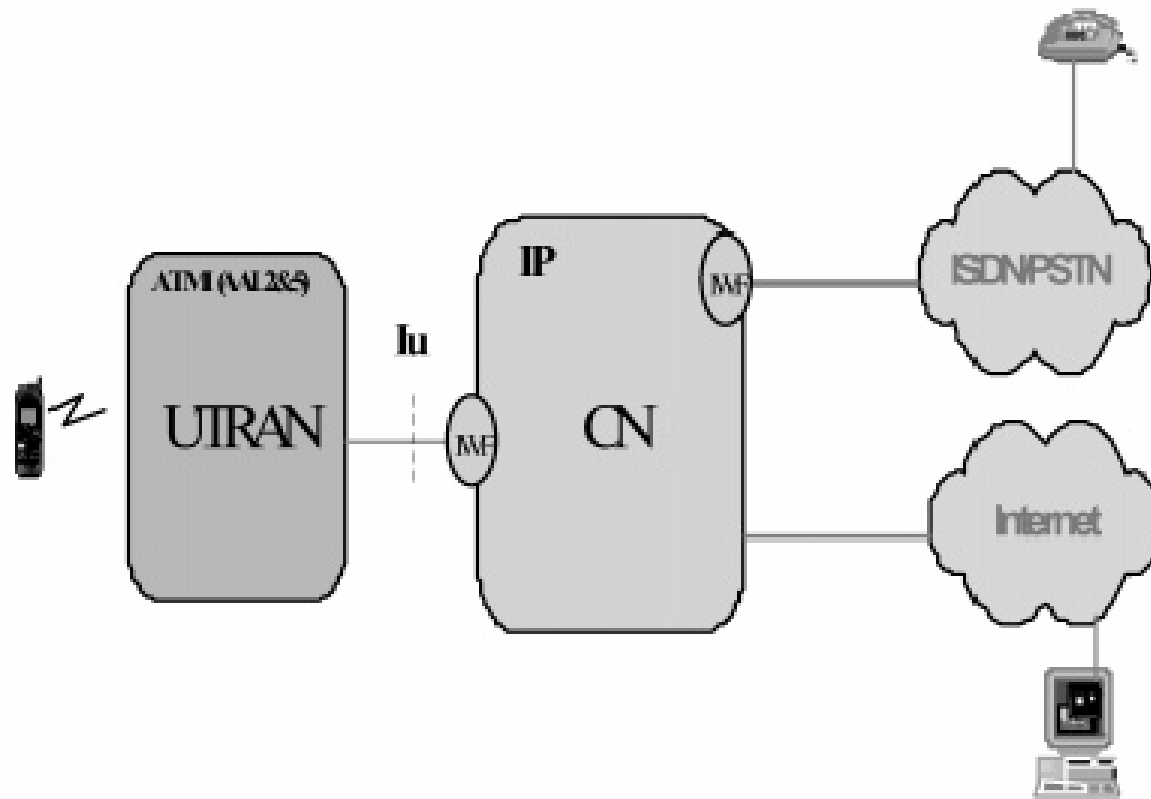
IWF: Internet Working Functionality



# Option 2



# Option 3



# Option 4

