

# Evaluation Method

- Interim and Final Report
- Attendance is not Checked, but, ...
- Questions or Comments are Mandated
  - In the quater, questions or comments with technical content must be made at least twice during lecture (may be in Japanese)
  - Good questions and comments will be awarded with points
  - Declare your name and student ID after each lecture, if you make questions or comments

# Advanced Lecture on Internet Applications

## 1. IPv4, IPv6, UDP, DNS

Masataka Ohta

[mohta@necom830.hpcl.titech.ac.jp](mailto:mohta@necom830.hpcl.titech.ac.jp)

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

# Structure of Lecture

- 2nd Quater
  - Advanced Lecture on Internet Infrastructure
    - Physical, Datalink and Network Layers
- 4th Quater
  - Advanced Lecture on Internet Applications
    - Transport and Application Layers

# Purpose of the Lectures

- Understand the Principle of the Internet and Knowhow of Internet Style Protocol Design
  - The end to end principle (rfc1958)
    - principle of global connectivity
    - principle of scalability
- We are in Protocol Era
  - Protocols designed for various applications
  - APIs are of secondary importance

# Reference Articles for Lecture

- RFC (Request for Comments)
- 「本当のインターネットをめざして」、情報処理学会誌、全36回(1999年4月号～2002年3月号)
- 「インターネットの真実」、週間東洋経済(2001年1月より2002年4月まで連載)
- Slides used in the Previous Fiscal Year (in Japanese only)
  - [ftp://chacha.hpcl.titech.ac.jp/2018/appli\\*.ppt](ftp://chacha.hpcl.titech.ac.jp/2018/appli*.ppt)

# Topics for the 4th Quarter (1)

1. IPv4, IPv6, UDP, DNS
2. Transport Layer: TCP, congestion control, long fat pipe, multihoming
3. File Transfer: TFTP, FTP, reliable multicast
4. Character Communication: character code and internationalization
5. Character Communication: TELNET, e-mail, SMTP, MIME

# Topics for the 4th Quarter (2)

6. Character Communication: Web, HTTP, HTML, GIF, Java Script
7. Character Communication: home appliance control
8. Stream Communication: RTP, time synchronization, clock synchronization
9. Stream Communication: telephone network and the Internet, internet phone

# Topics for the 4th Quarter (3)

- 10. Internet and Society: user authentication, accounting, radius
- 11. Internet and Society: IPR, legal issues
- 12. Internet and Society: standardization, RFC, operation, implementation, protocol design



# What is Protocol?

- Procedure to communicate over networks

# BTW, What is the Internet?

- Not e-mail
  - seriously thought so 20 years ago
- Not web, either
  - many still misunderstand so
- Is not applications
- The Internet is a network directly connecting terminals based on the principle of the Internet using IP (Internet Protocol)

# The End to End Argument

<http://web.mit.edu/saltzer/www/publications/endtoend/endtoend.pdf>

- The **function** in question **can completely and correctly be implemented only with the knowledge and help of the application standing at the end points of the communication system.** Therefore, **providing** that questioned function **as a feature of the communication system itself is not possible.** (Sometimes an incomplete version of the function provided by the communication system may be useful as a performance enhancement.)

# Example of E2E Argument data drop

- can't be prevented by network
  - noise, equipment down
    - buffering by network equipment is vulnerable to equipment down
- to prevent data drop
  - end system must resend
- what if the end system goes down?
  - safe unless end system goes down is the ultimate reliability

# E2E Argument and E2E Principle

- according to the E2E argument
  - network function can not be complete without help of end systems
  - network should have moderate functionality
    - can't be complete, anyway
  - lacking functionality may be provided by end systems
- many functions can be provided completely only by end systems

# End to End Principle

## Disintermediated Networking

- Implement things by terminals (end) not by the network
  - network equipment has only single function (to connect terminals) and is high speed
- Implement things by directly involved terminals without involving other terminals
  - scalable (no load concentration)
  - highly reliable (system works if only terminals are working and can communicate each other over some route)

# What is not the Internet (1)

## e-mail

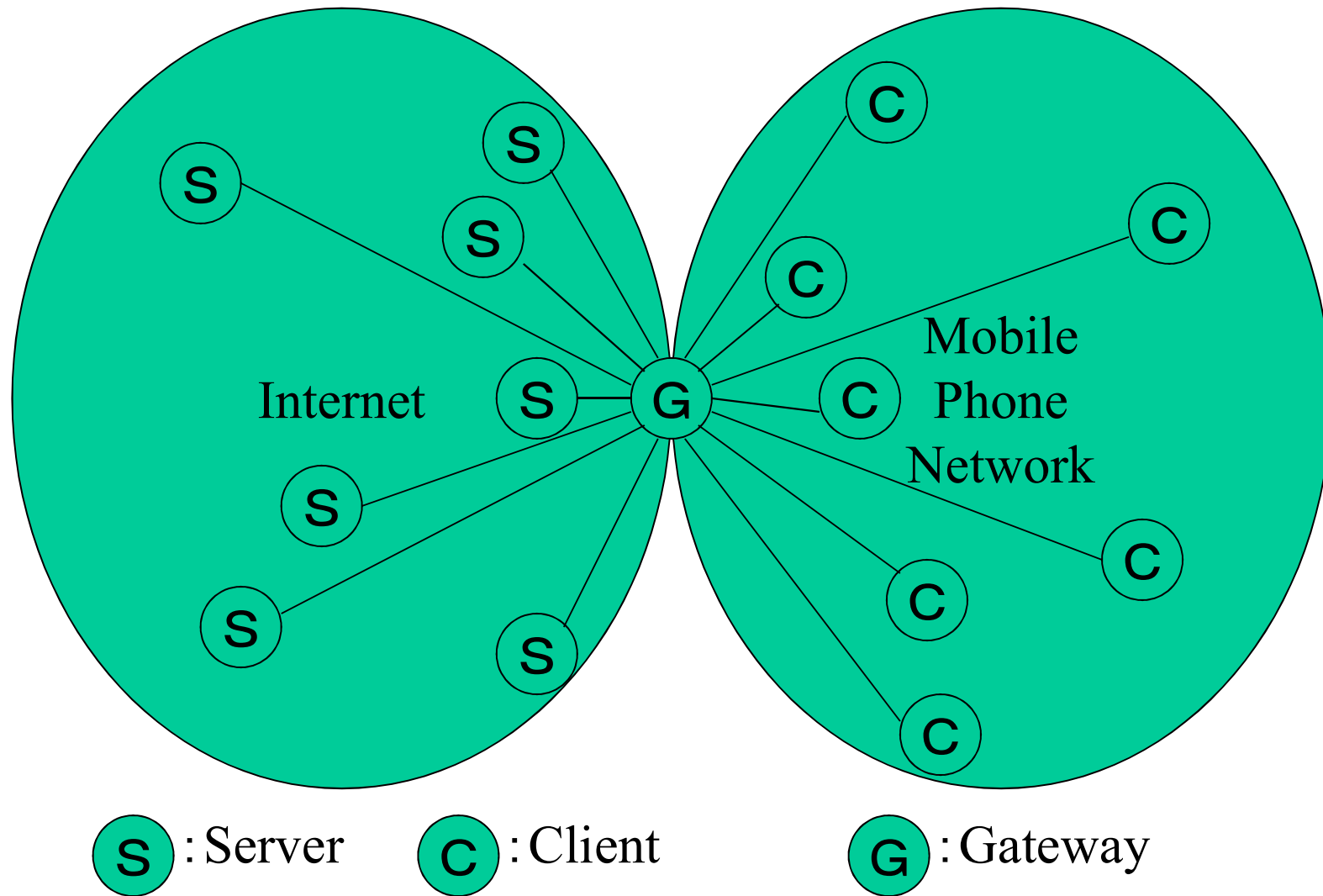
- UUNET (JUNET) was not the Internet
- Communication System for Personal Computers was not the Internet
- E-mail is an application works on the Internet
  - also works on other (phone) networks
- In the past, (oversea) e-mail was charged
  - some mail on mobile phone is still charged

# What is not the Internet (2)

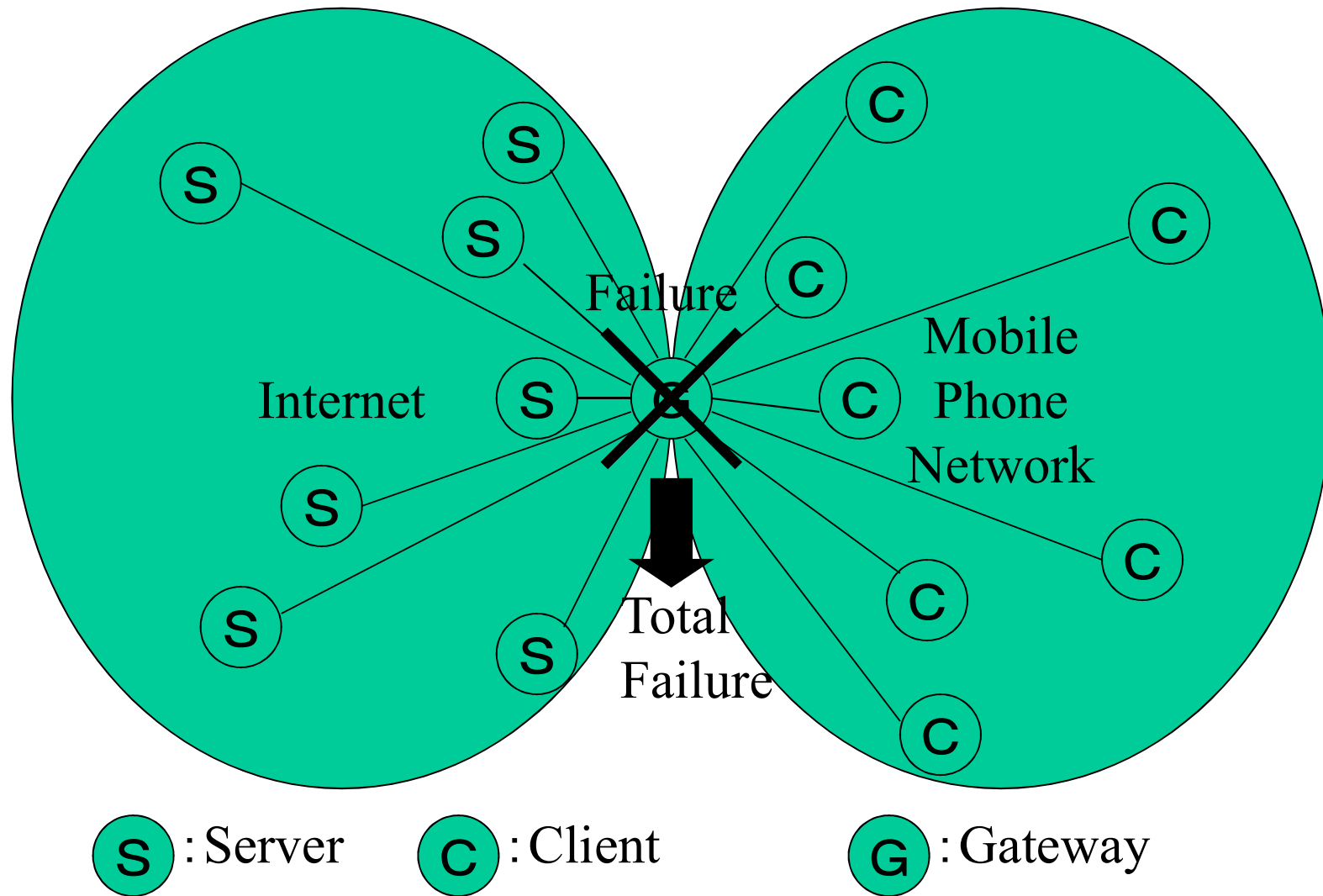
## Web

- Web is not the Internet
  - though Microsoft makes it obscure
- Web, too, is an application works on the Internet
  - also works on other networks
  - web browsing from mobile phone network is, in principle, charged
    - mobile phone network is not the Internet, of course

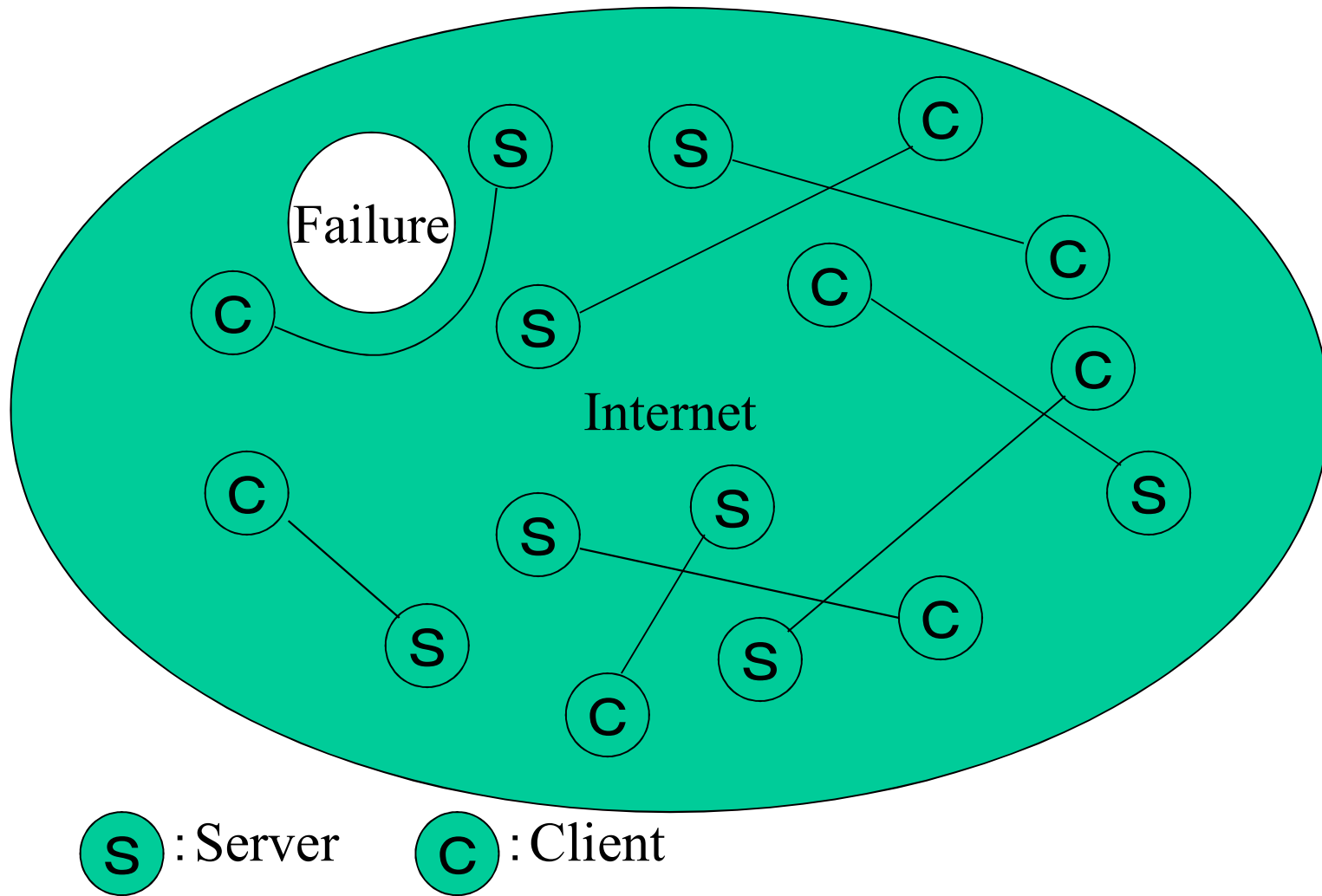




Web Browsing of i-mode from Mobile Phone Network  
Servers and clients are separated and communicate through the gateway



Web Browsing from Mobile Phone Network  
Servers and clients are separated and communicate through the gateway



Internet

Servers and clients are mixed and communicate with shortest pathes

# What is not the Internet (3)

## Phone

- Phone is not phone network
- Phone is an application works on phone networks
  - also works on other networks
  - phone over the Internet is free of charge
- Phone, today, is a promotion tool to sell Internet services
  - no need to support phone network
  - mobile phone is to sell mobile Internet services

# What is not the Internet (4)

## Phone Network

- Phone is not phone network
- Phone is an application works on phone networks
  - also works on other networks
  - phone over the Internet is free of charge
- We don't need phone networks
  - neither are mobile phone networks
  - phone numbers are not necessary, either
    - IP addresses! (though alternatives may exist)

# Networks

- Physical Distribution Networks
  - postal service, parcel services, convenience stores
- Information Communication Networks
  - Publishing Network (Book, News Paper, CD, Movie)
  - Financial Network
  - Phone Network
  - Broadcast Network
  - the Internet

# Internet Disintermediate ICN

- Price Destruction of ICN
  - Publishing, financial, phone and broadcast networks will disappear
  - IC cost of the society decreased
    - ISP business itself is not profitable
- Publishing, financial, phone and broadcast services will:
  - remain, but, on the Internet
  - social activities increase

# Publishing Network

- Mass Distribution of Same Information
- Delay of the Distribution may be Tolerated
- Protected by Copyright Act
- The First Victim of the Internet
  - Collapsing



# Financial Network

- Manage Transfer of Money
- Partly, Physical Distribution Network, but, today, mostly ICN
- Security!!!
  - Not that there is no accident
  - Who will pay the loss on accidents

# Phone Network

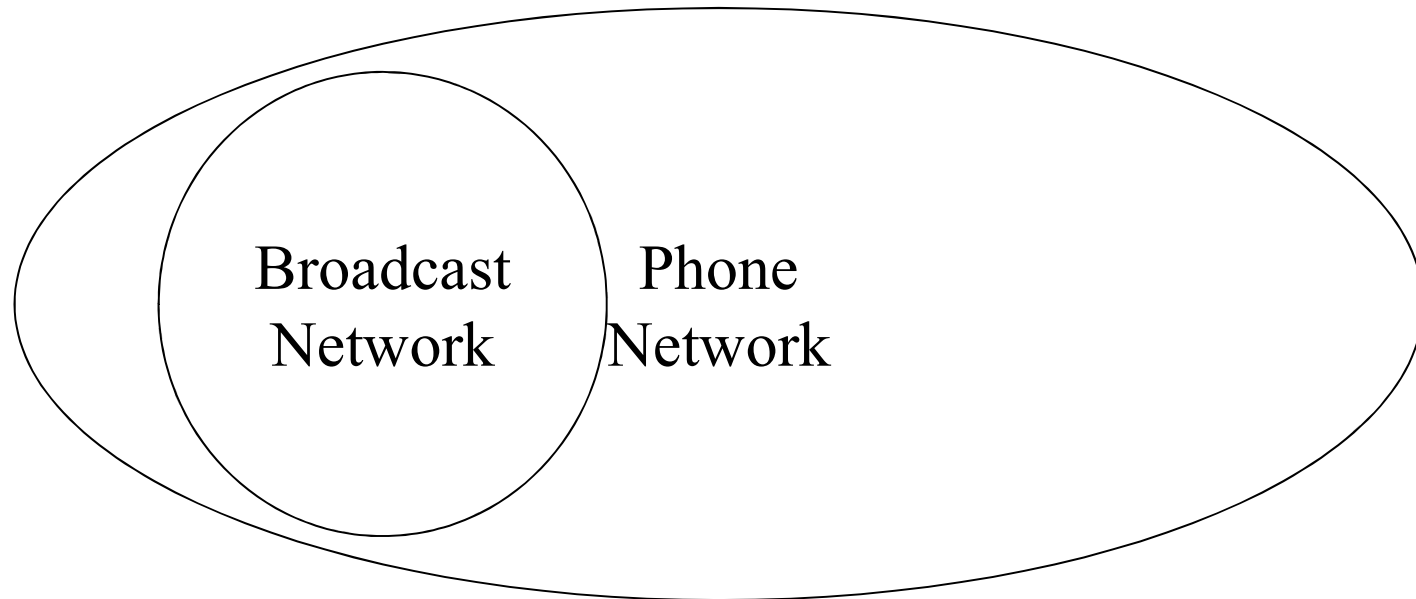
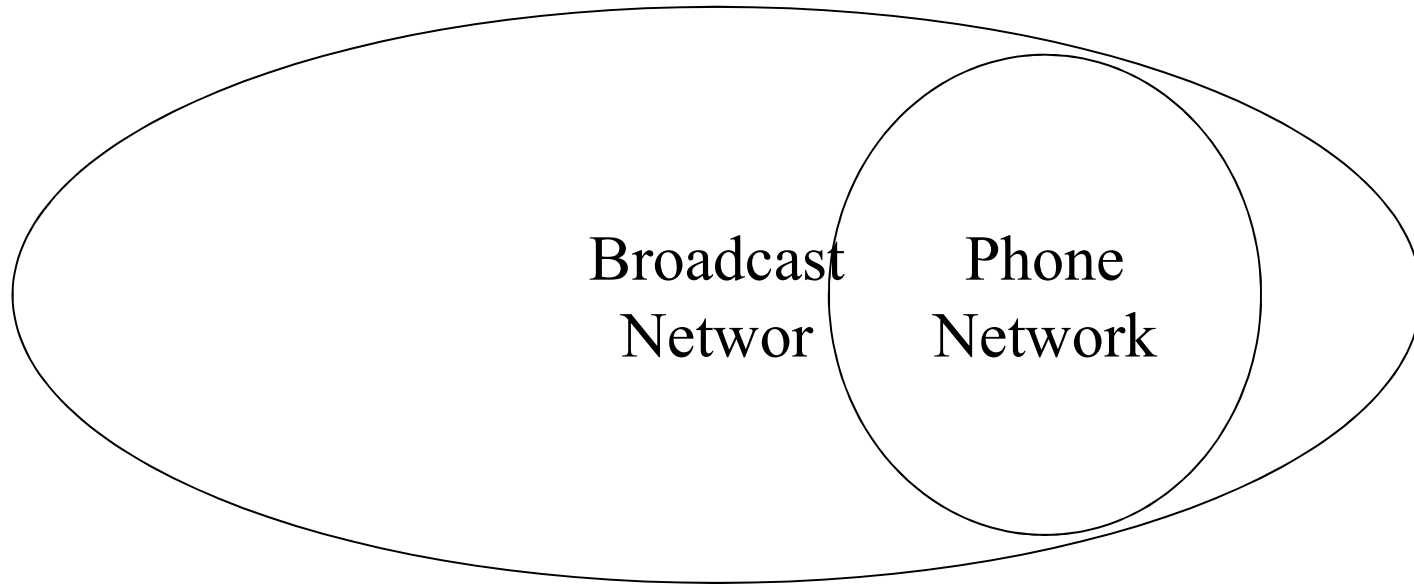
- Network for Realtime Voice Transfer
  - Allocate bandwidth for voice transfer
  - Minimize (guarantee) delay for voice transfer
- Dedicated line service may be Offered
  - but, primary service is voice transfer
- Slow and Expensive
- Was Protected as National Company
  - Liberated by Telecommunication Business Act

# Broadcast Network

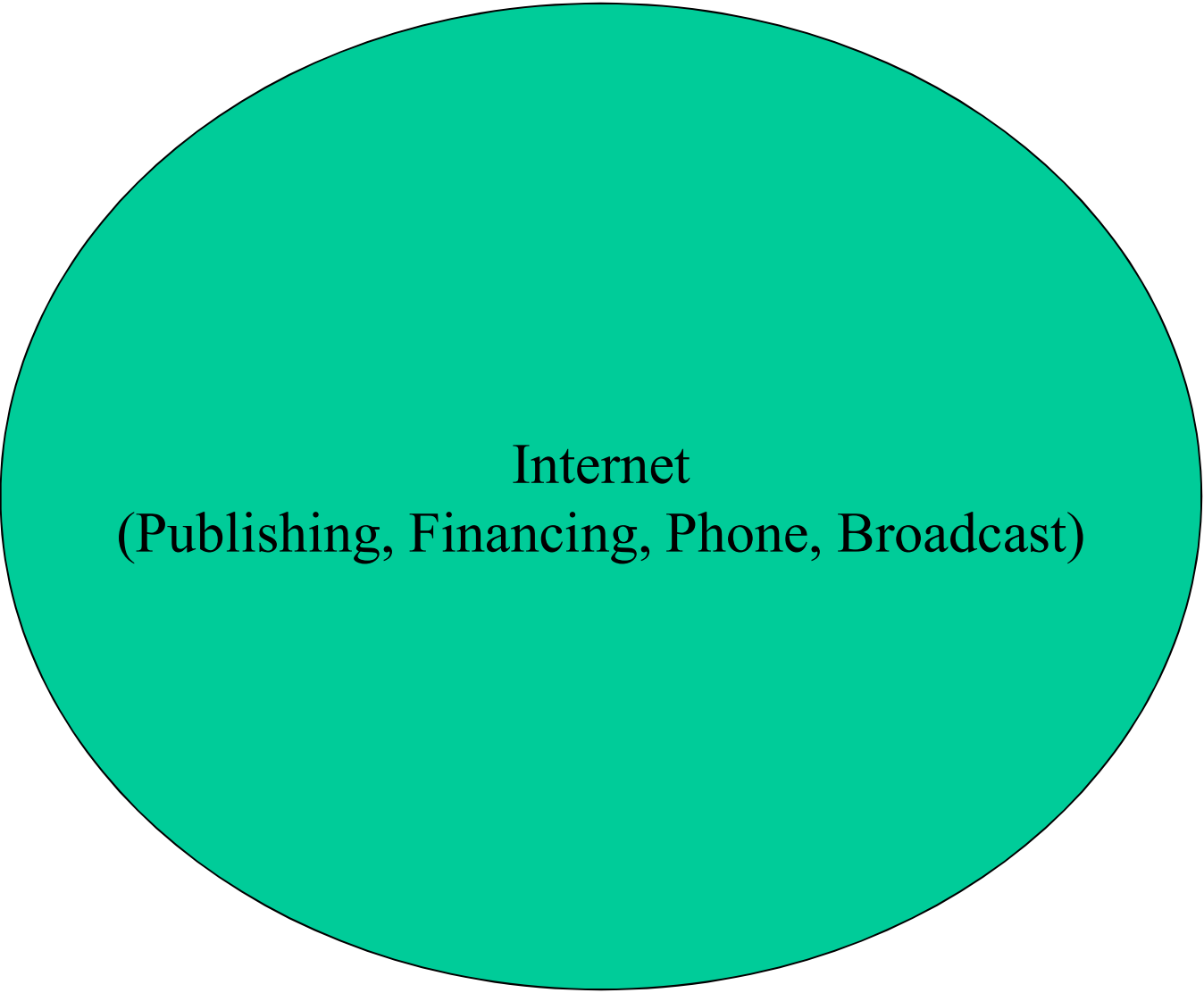
- Network to Transfer Voice/Image to Many in Realtime
  - Allocate bandwidth for the transfer
  - Minimize delay
- Wide Area One to Many Communication over Radio Waves
  - Broadcast/Multicast
- Protected by Broadcast Act

# Integration of Broadcast and Telecommunication

- Viewed from Phone Network
  - one to many transfer over phone network
    - integration of broadcast network to phone network (BISDN)
- Viewed from Broadcast Network
  - receive feedback through phone network
  - one to one communication over radio wave possible?
    - integration of phone network to broadcast network
- Viewed from the Internet?



Two Visions on “Integration of Broadcast and Telecommnication

A large teal circle with a black outline, centered on the page. Inside the circle, the text "Internet" is written in a black serif font, with "(Publishing, Financing, Phone, Broadcast)" written below it in the same font.

Internet  
(Publishing, Financing, Phone, Broadcast)

Integration of IC Services by the Internet

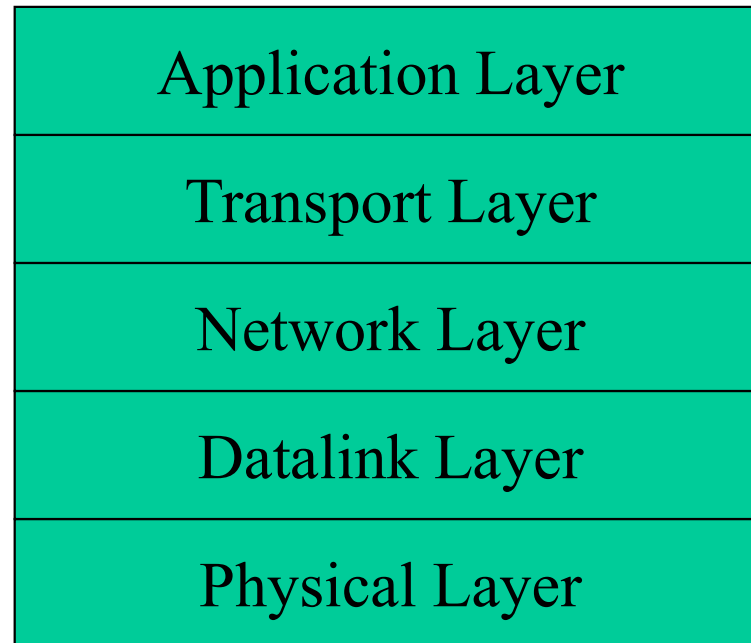
# Layering of Protocols

- Have Layers based on Level of Abstraction
  - 7 layer model of OSI (Open Systems Interconnection)
  - 5 layer model of the Internet
- Corresponds to Subroutines (Structuring) in Programming

Application Layer	Layer 7
Presentation Layer	Layer 6
Session Layer	Layer 5
Transport Layer	Layer 4
Network Layer	Layer 3
Datalink Layer	Layer 2
Physical Layer	Layer 1

Layering Structure of OSI





Layering Structure of the Internet

# Physical Layer

- Map Physical Phenomena and Information
  - Voltage high/low  $\leftrightarrow$  0/1
  - Light on/off  $\leftrightarrow$  0/1
  - Amplitude and Phase  $\leftrightarrow$  0/1/.../63
- Multiple Similar Physical Layers may be Integrated (Repeaters)
- Corresponds to Control Firmware in Peripheral Devices in

# Datalink Layer

- Joint between Network and Physical Layers
- If Physical Layer have more than 2 devices
  - distinction between them is necessary
- Multiple Dissimilar Physical Layers may be Integrated
  - Local Relays (Bridges)
- Corresponds to Device Drivers in Programming

# Network Layer

## Internetworking Layer

- Integrate Many Datalink Layers to form a Global Network
- Global Relaying (Routers, Gateways)
- Corresponds to File Manager and Inter-process Communication in Programming

# Transport Layer

- Network Layer Identify Terminals
- Transport Layer Identify Communications
  - Multiple communications may exists between a pair of terminals
    - different communication is processed by different process
    - may require different bandwidth etc.
      - unimportant in the current best effort Internet
- Corresponds to Process Manager in Programming

# Session, Presentation and Application Layers

- Corresponds to Internal Structure of each Process in Programming
- Corresponds to Internal Structure of each Process in Terminals
- Without Interworking, there is no Point to have 3 Layers
  - Layering of the Internet has the Application Layer only

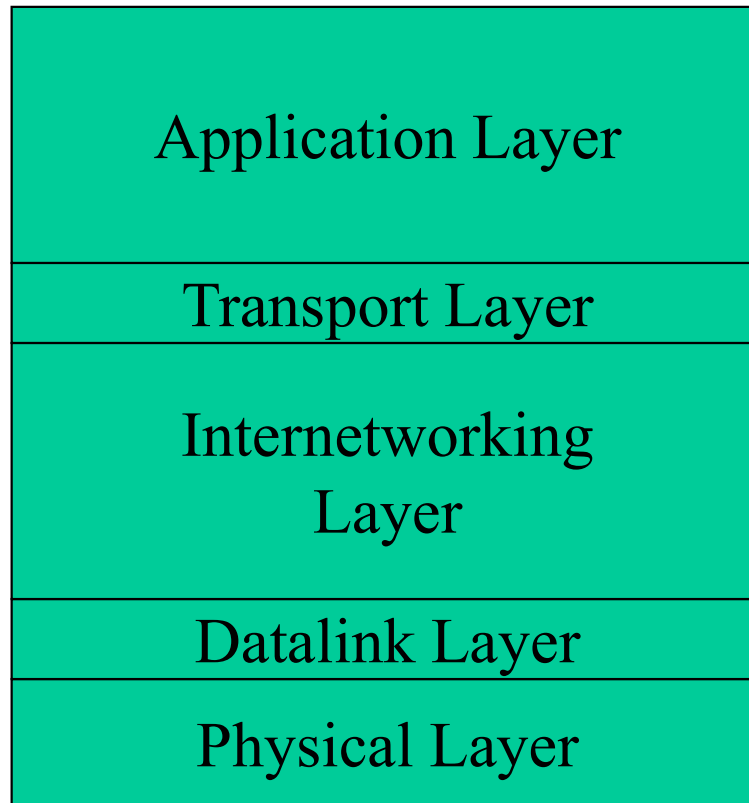
# Transport and Application Layers

- With Best Effort Network, Distinction is within Each Terminal
- Assigning Packets of Each Communication to Corresponding Process is Transport
- Further Distinctions is not Meaningful
  - Protocols shared by many applications (e.g. TCP (assure reliability and manage bandwidth)) are, traditionally, classified as Transport

# Layering of the Internet

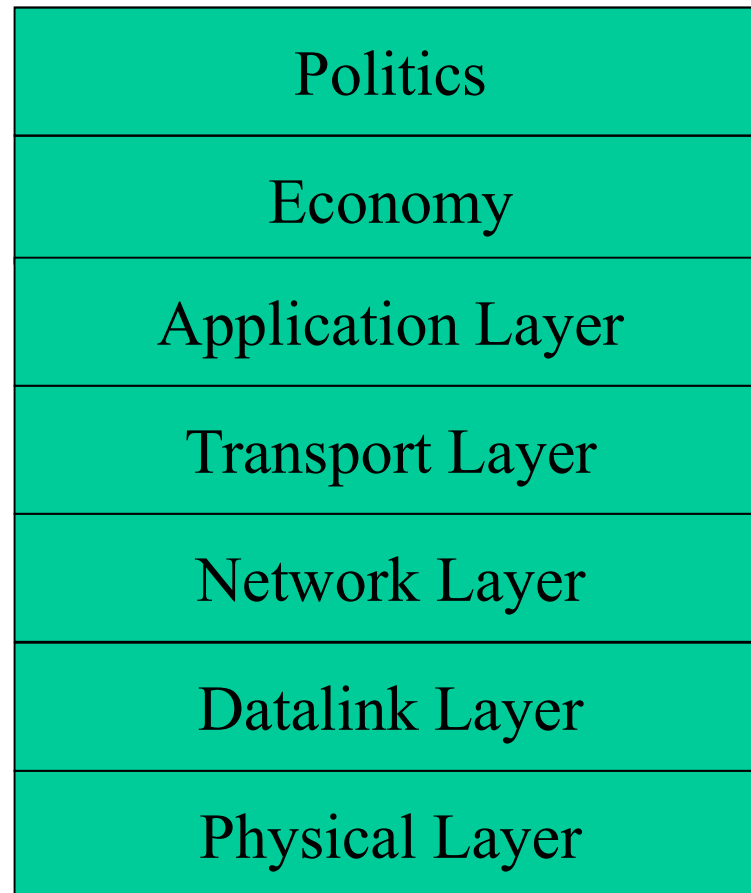
- Physical and Application Layers are Essential
- The Internetworking Layer does as Much Things as Possible
- Datalink and Transport Layers should Avoid to do Thing



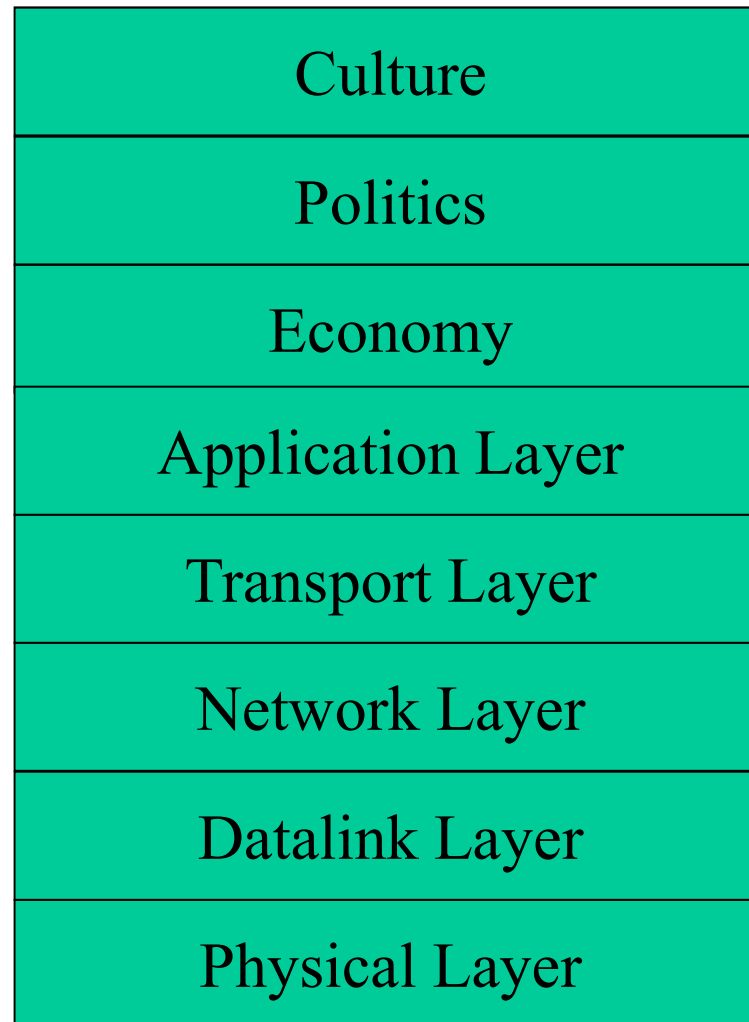


Here is the Essence of  
the Internet

Layering Structure of the Internet



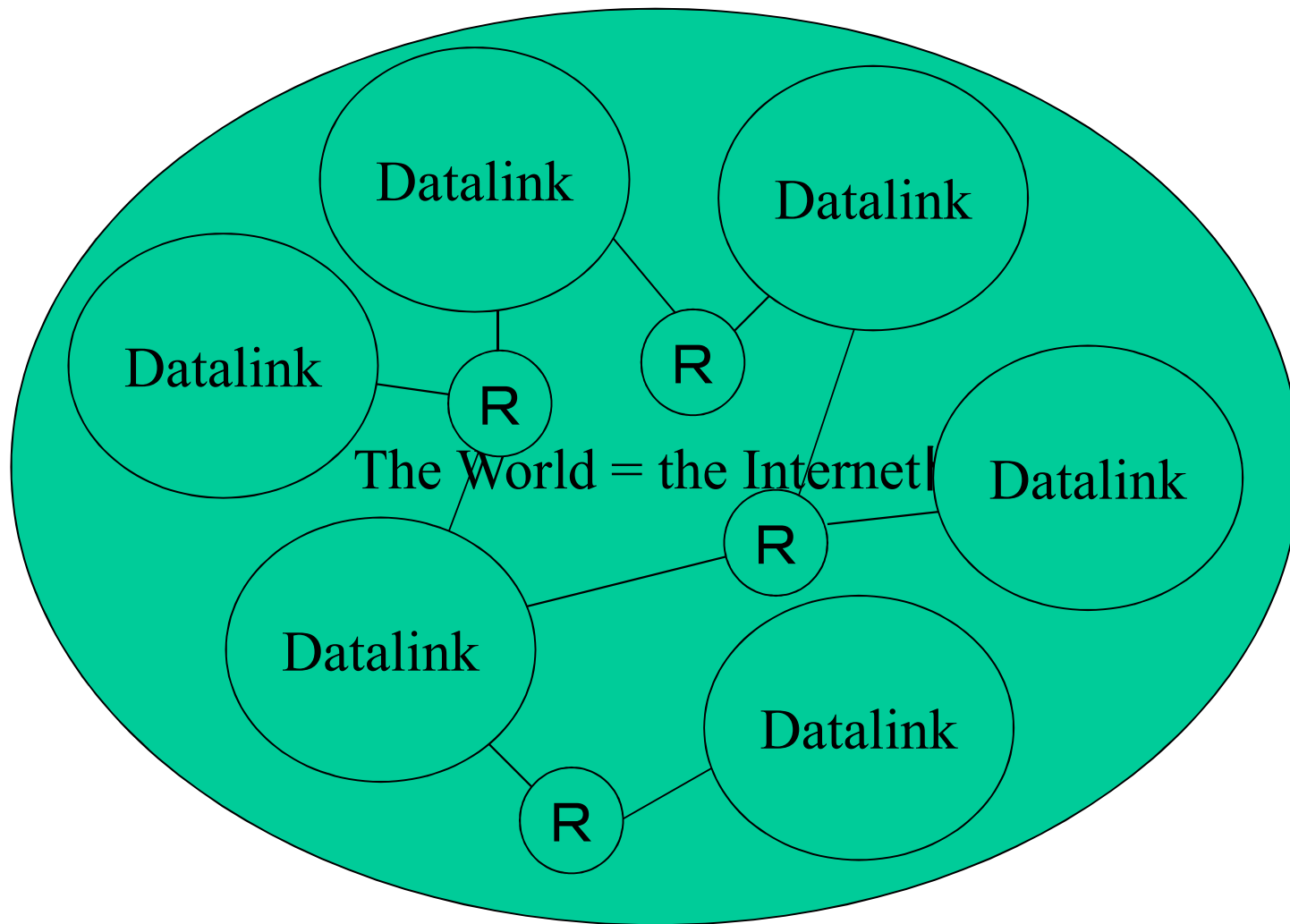
Layering Structure over the Internet?



Layering Structure over the Internet!

# Structure of the Internet

- CATENET Model
  - Many small (w.r.t. # of devices) datalinks interconnected by IP (Internet Protocol) routers

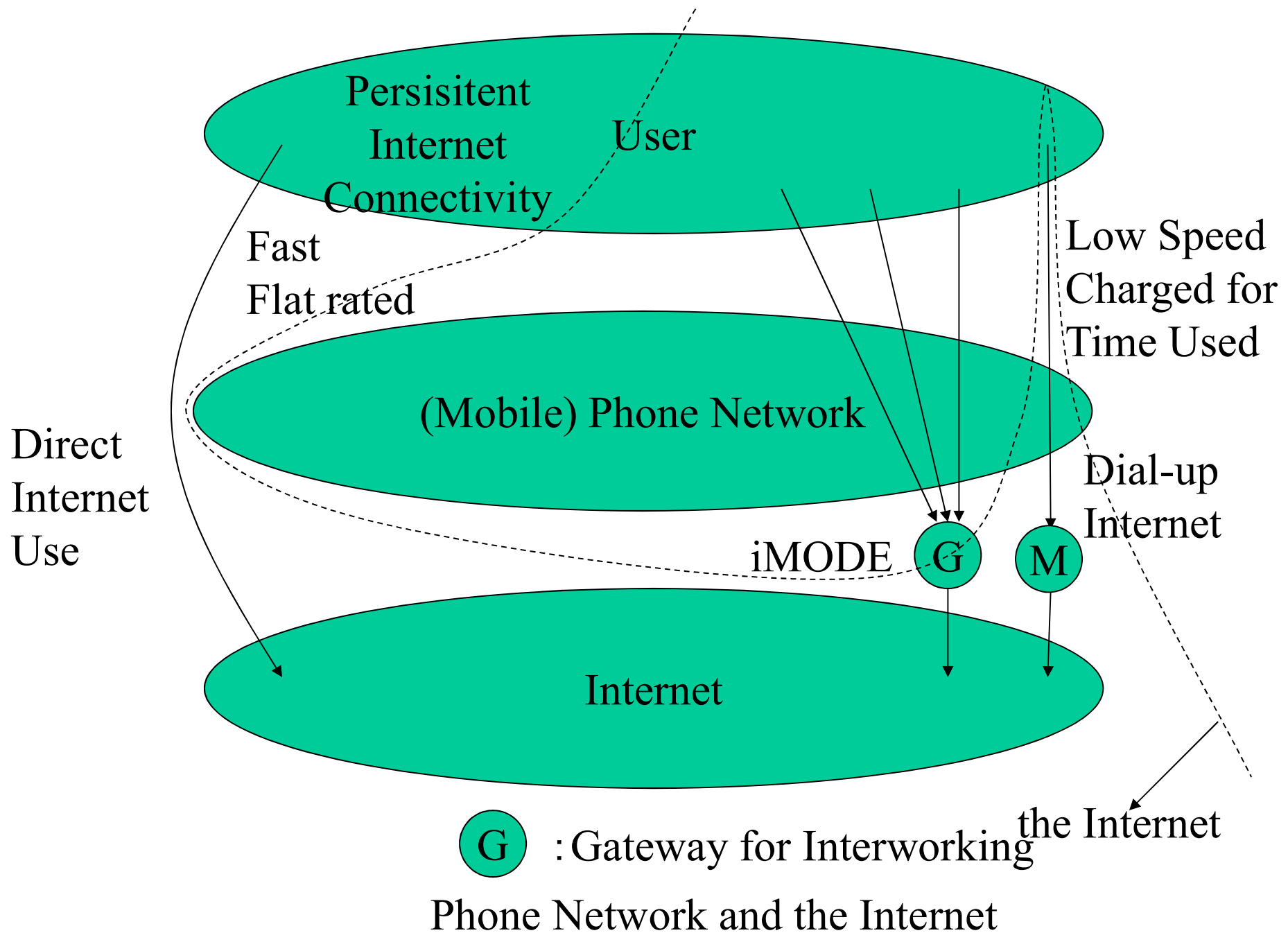


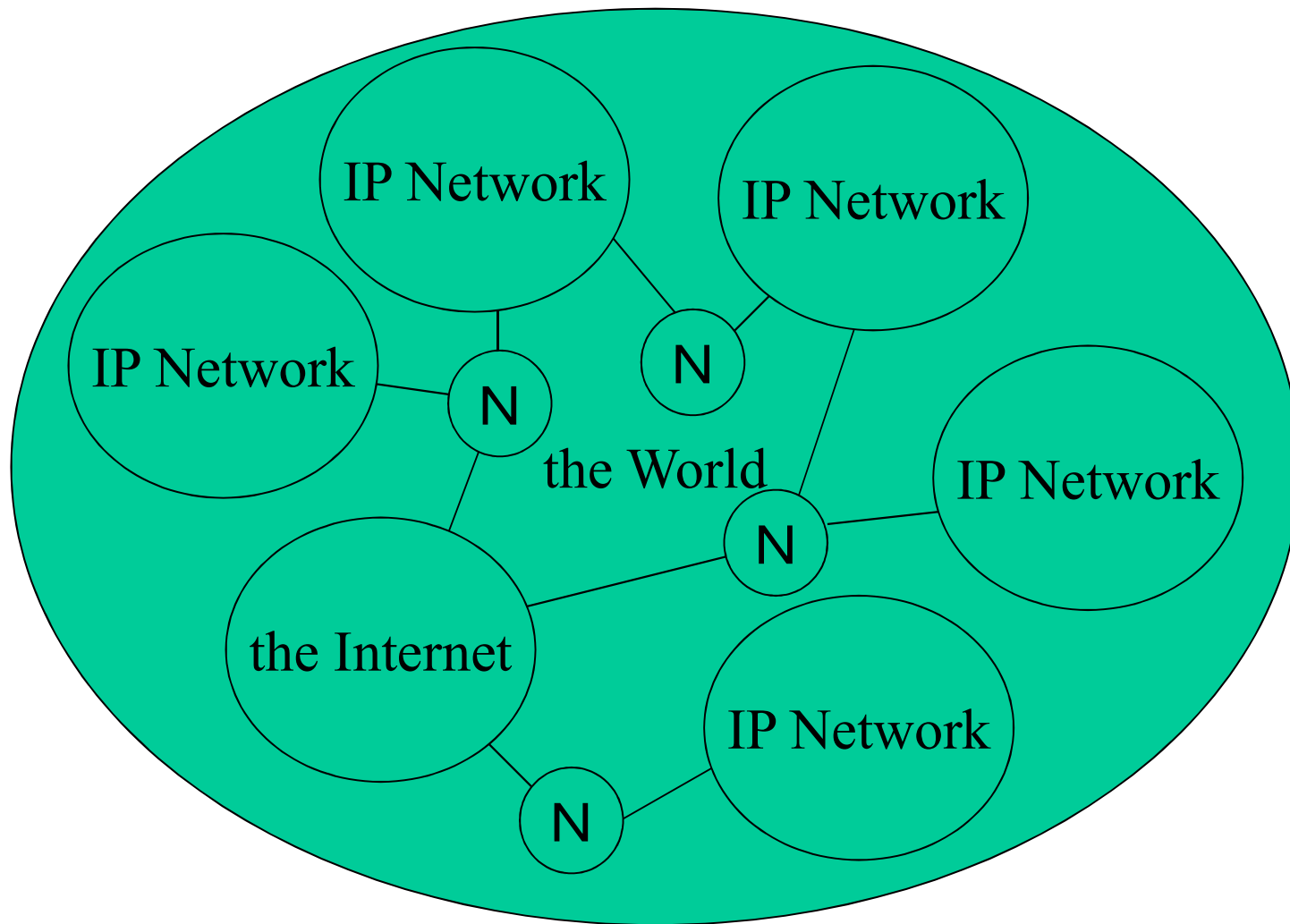
 : Router

CATENET Model

# The Internet and Structure of Networks

- Example of Internet
  - Dial-up Internet
- Example of non Internet
  - i-mode
    - IP, but, relaying at transport layer
  - Legacy NAT
    - IP, but, addresses etc. are modified, which is not visible to terminals
      - Interworking at the transport layer and above





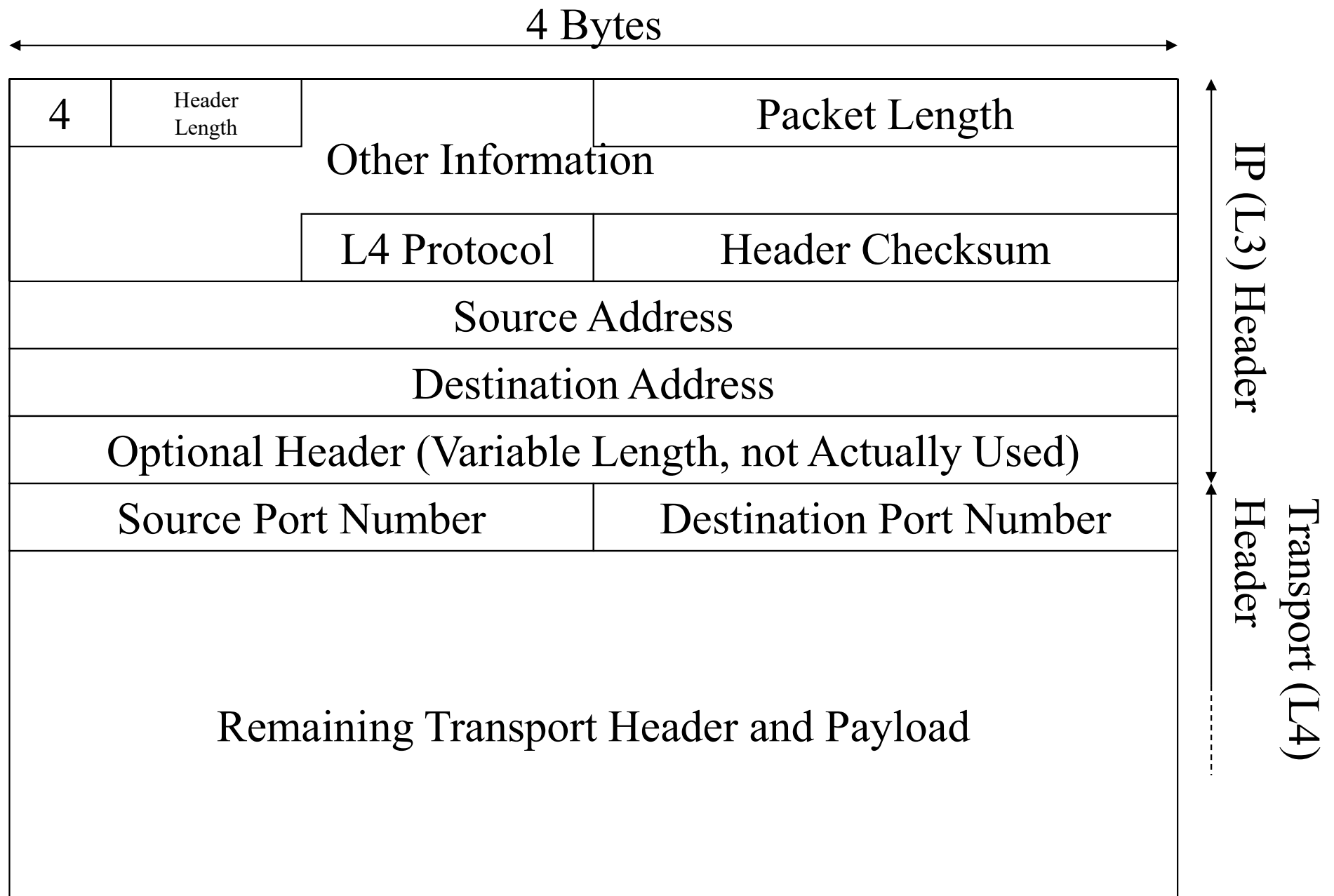
 : NAT Device

NAT and the Internet

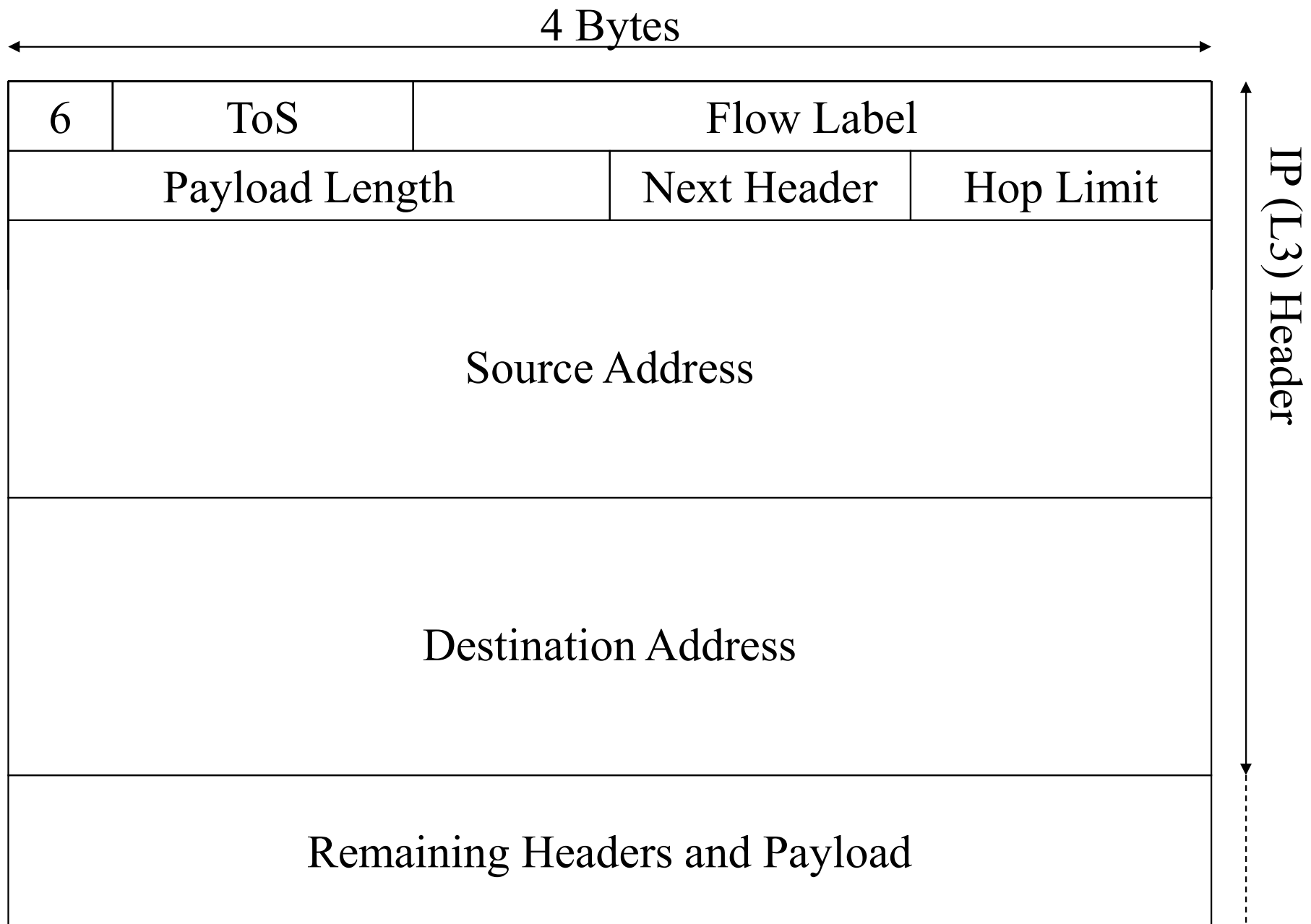


# Data Format of the Internet

- Data are Assembled to Form Packets
- Each Packet has its Own Destination
  - datagram, not virtual circuit
- With IPv4, 20B Internetworking Layer Header is Attached
- In Addition to a Transport Layer Header



Format of IPv4 Packets



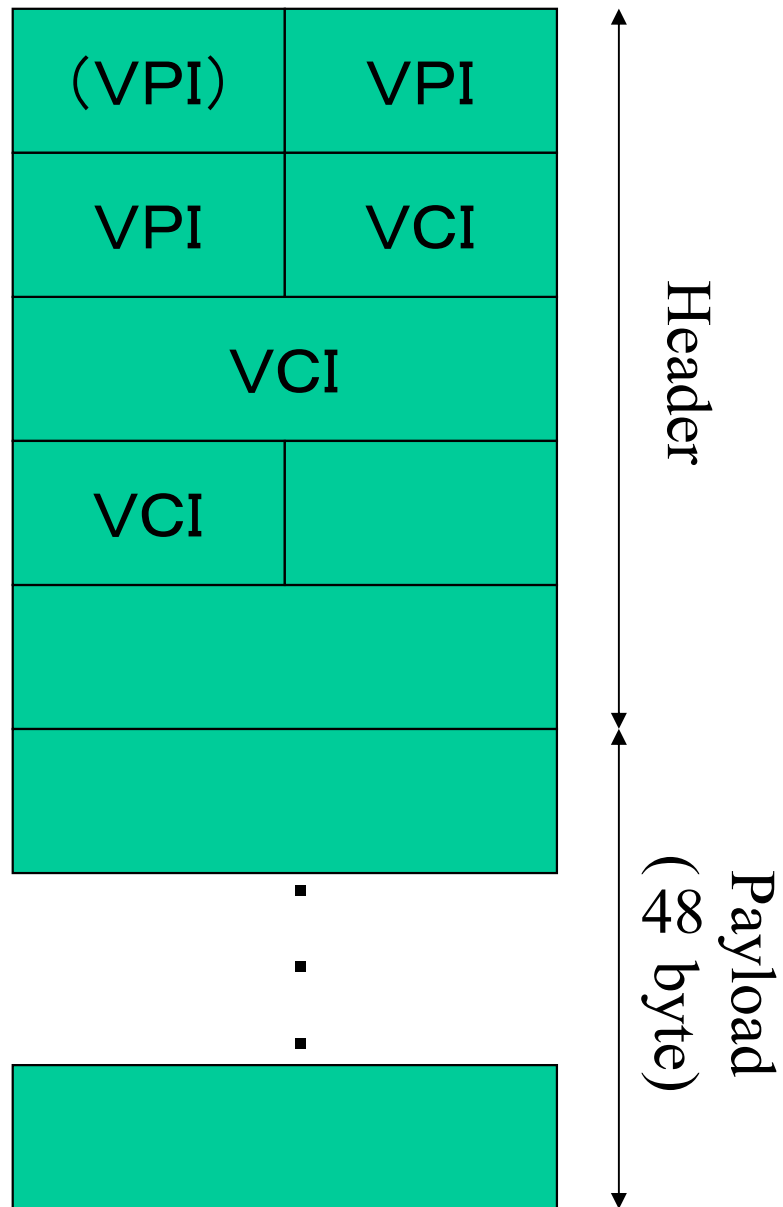
IPv6 Packet Format

# Function of IP Routers

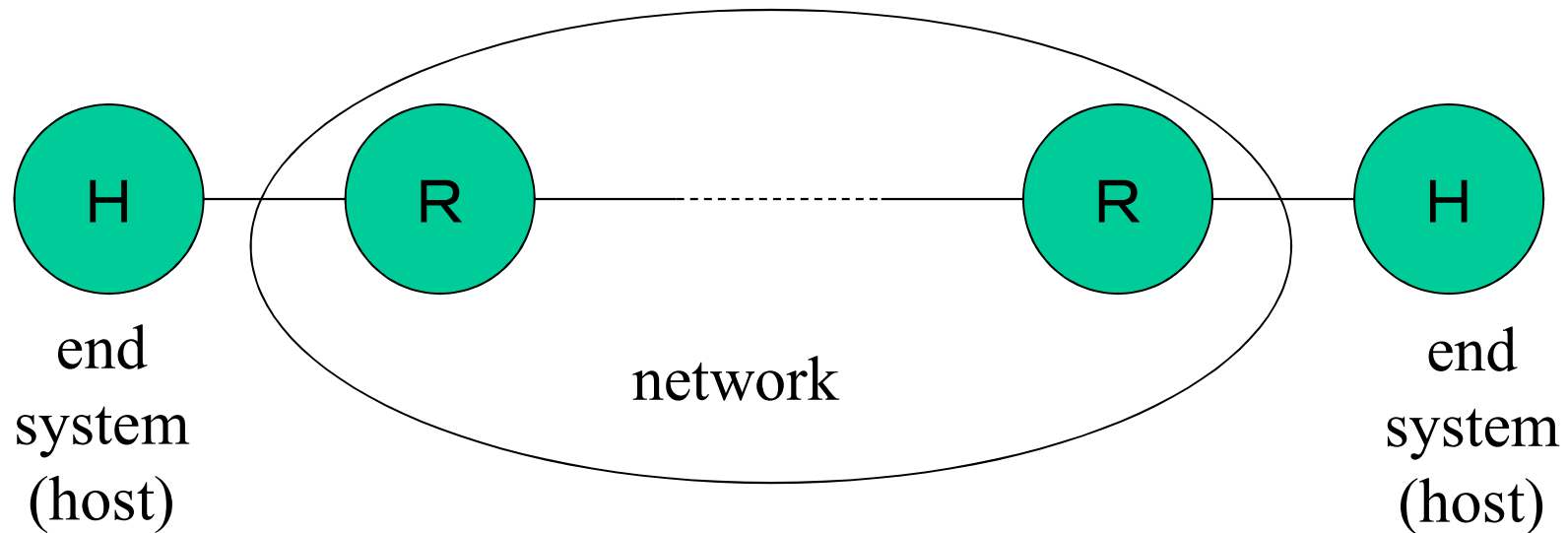
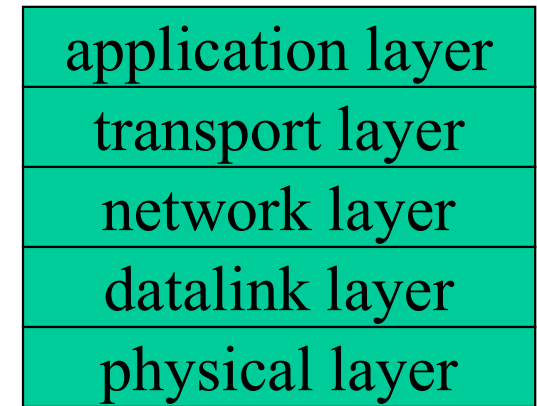
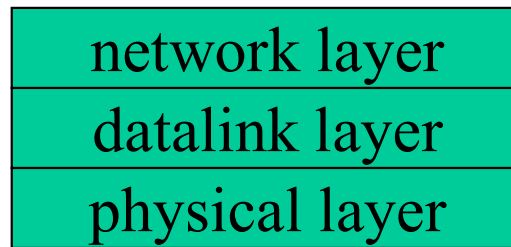
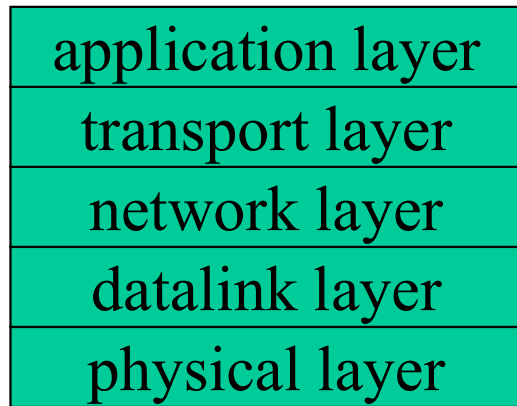
- decrement TTL and forward packet **based on destination address**
  - routing table is constructed **in advance** by routing protocols
  - no advance signaling, no BW guarantee
- with IPv4, may divide packets for datalinks with small MTU (fragmentation)

# ATM (Asynchronous Transfer Mode)

- packet (cell) network of phone companies
- data is divided into fixed length 48B cells and 5B simple header is attached
  - faster than processing complex header?
- cell header contains ID (VPI/VCI)
  - ID is obtained from network, **in advance**, by telling destination to the network (signaling)
    - overhead in time (**even several seconds**) and processing
    - in theory, can provide QoS guarantee
    - **Virtual Circuit**



A cell of ATM

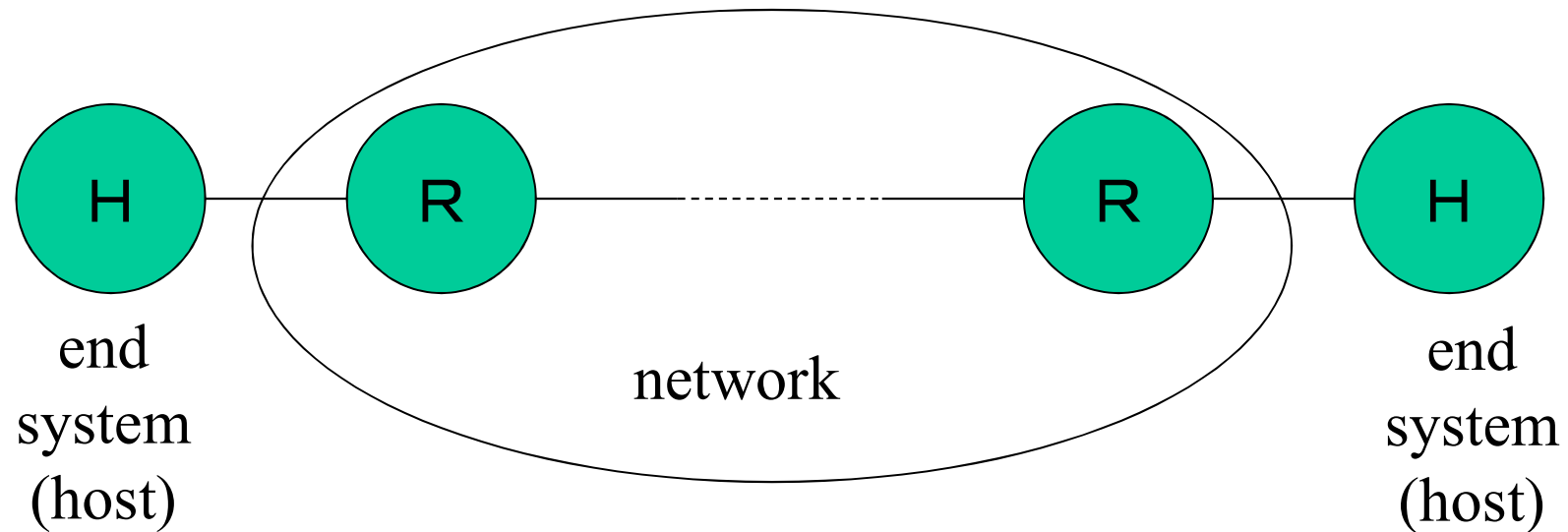


best effort internet

application layer
transport layer
network layer
datalink layer
physical layer

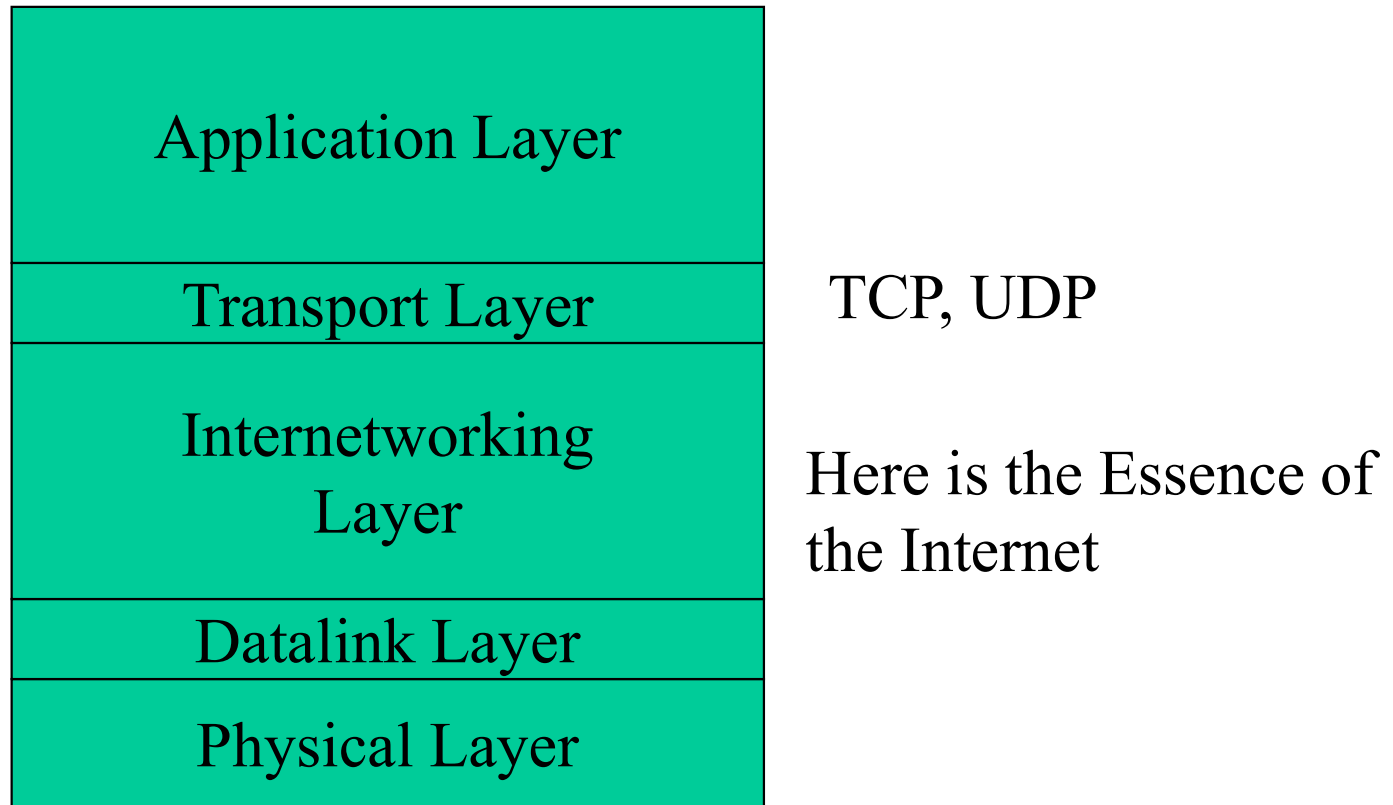
transport layer
network layer
datalink layer
physical layer

application layer
transport layer
network layer
datalink layer
physical layer



internet with QoS guarantee





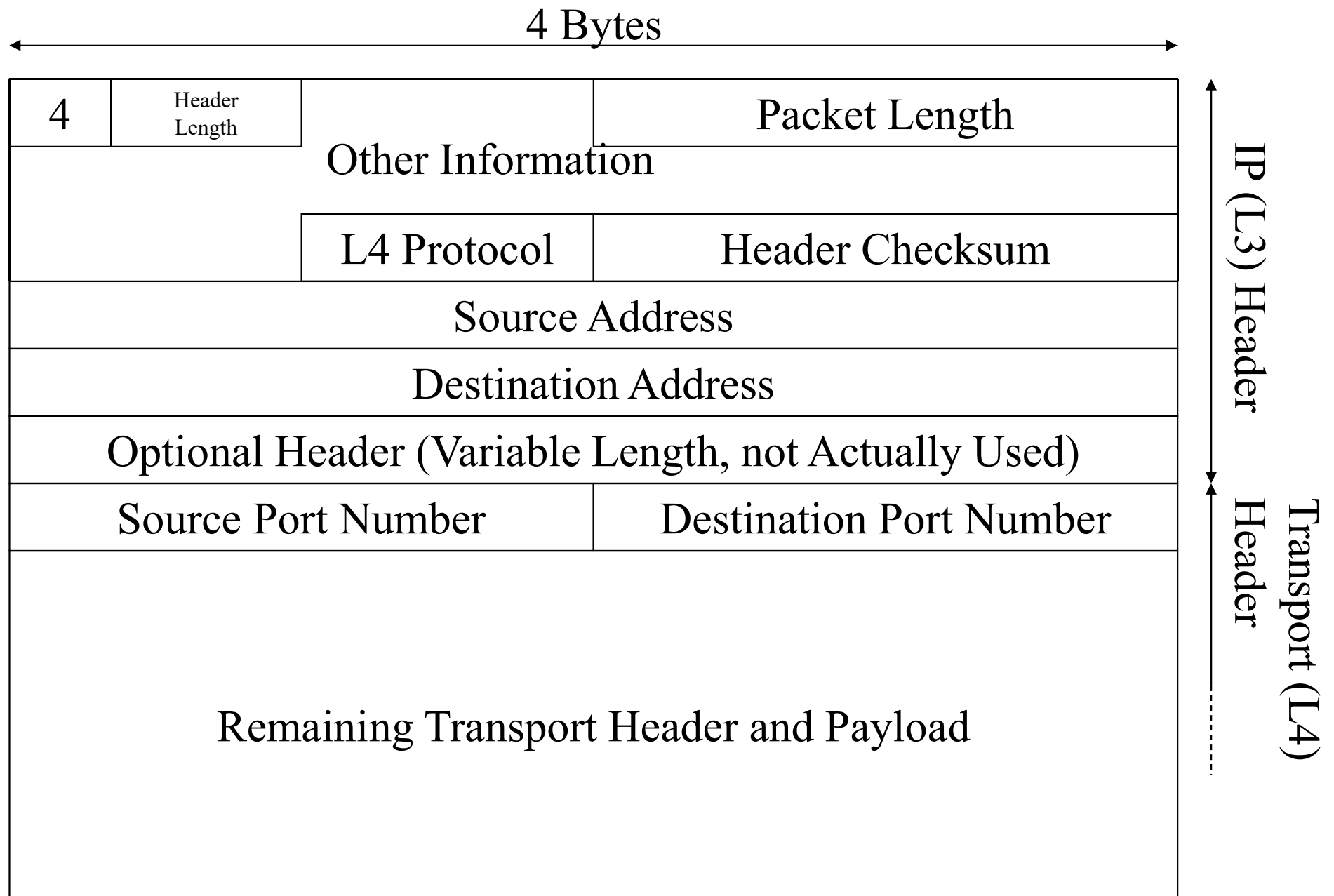
Layering Structure of the Internet

# TCP and UDP

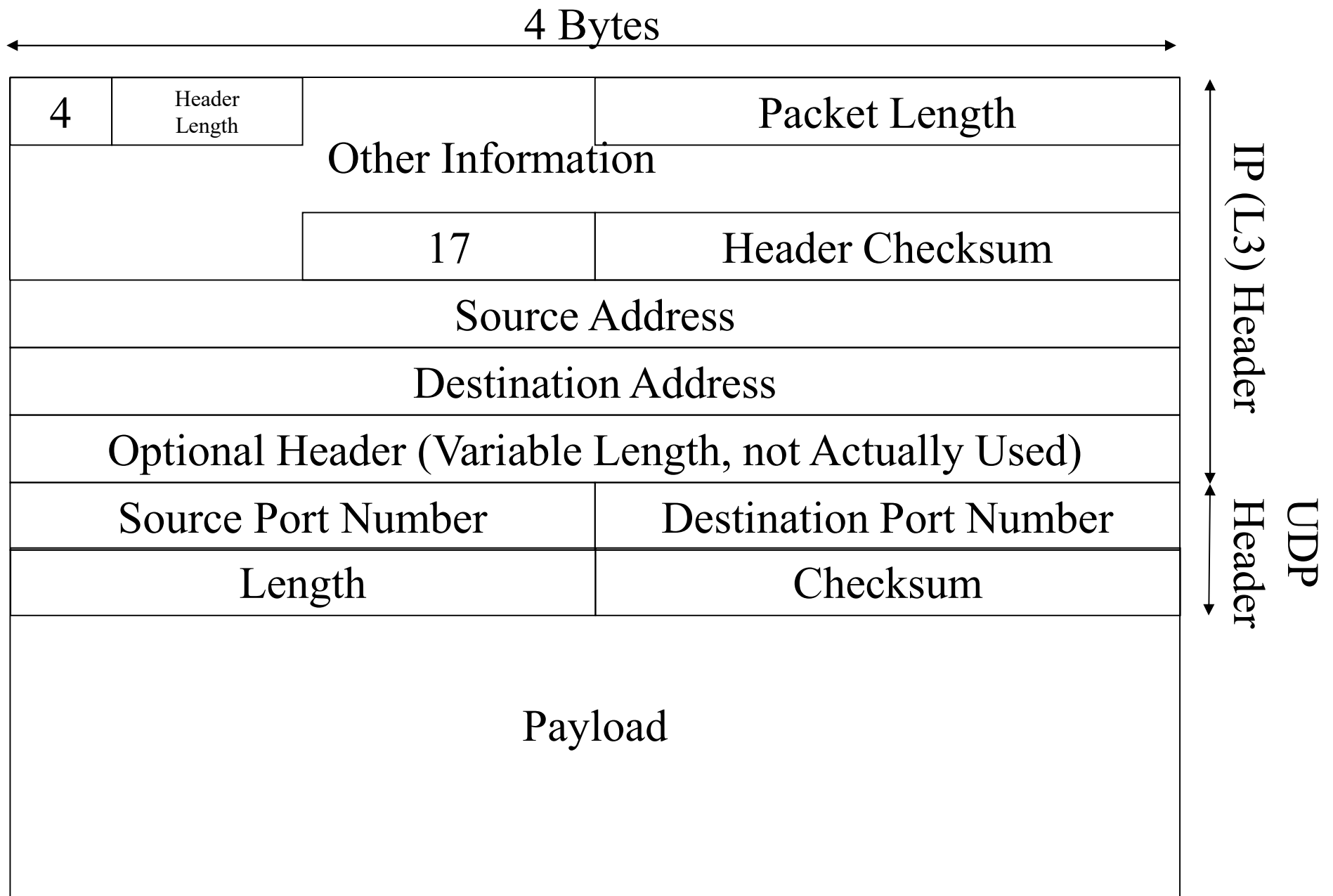
- TCP (rfc793)
  - Transmission Control Protocol
  - retransmit when data error or drop is detected
  - adjust transmission rate
- UDP (rfc768)
  - User Datagram Protocol
  - do nothing (let applications do something)
    - nothing except for delivery to applications

# UDP

- simple and, by itself, light weight
  - complex functions may be performed by applications
- connectionless at transport layer
- may be used by light weight applications
- must be used for communication not suitable for TCP (1:1 bidirectional)
  - unidirectional communication
  - multicast, broadcast



Format of IPv4 Packets



Format of IPv4 UDP Packets

# Destination Port Number

- destination host deliver packets to application processes based on port numbers
- source host can know port numbers from, say, URLs
- default port number may be assigned for well know services
  - eg., 53 for DNS

# Source Port Number

- may be used as destination port number for reply or ICMP error (error generated by intermediate routers)
- may be omitted (0)

# Length

- length of UDP header and payload in byte
- unnecessary?
  - IP header already have length
  - an IP packet can't hold multiple UDP packets



# Checksum

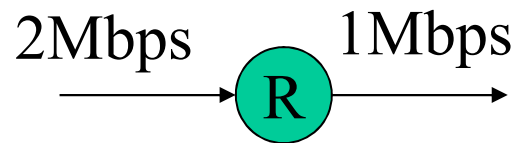
- one's complement sum
  - 0 has two representations (0...0, 1...1)
- may be omitted for IPv4
  - 0...0 means omission
- may not be omitted for IPv6
  - 0...0 is not allowed

# Phone Network vs the Internet

- which is more error prone?

# Reason of Packet Drop

- packet is lost upon transmission error
  - not so common
- routers must drop packet if buffer is full
  - primary reason of packet drop in the Internet



# Congestion Control

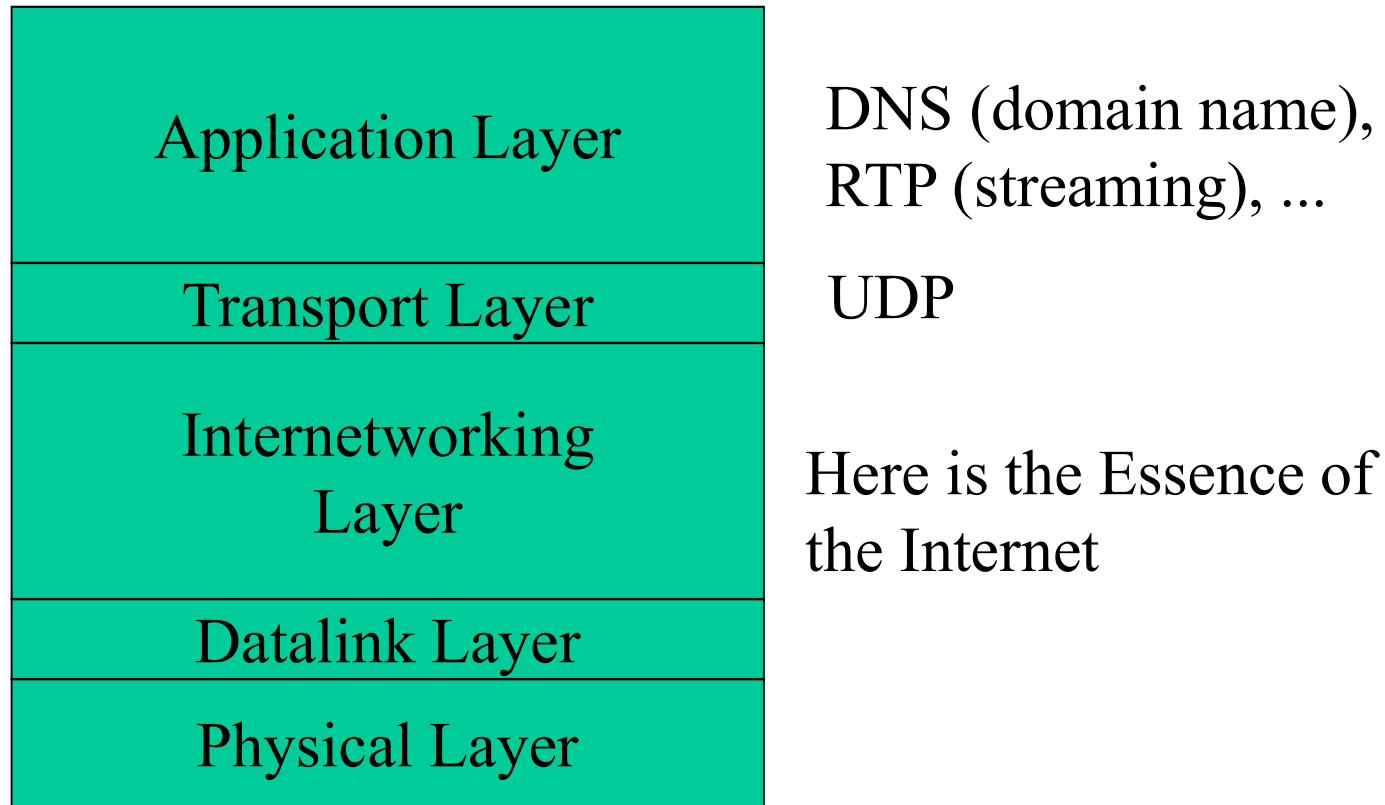
- BW is not managed in the Internet
- if everyone send packet at will, large amount of packet loss may occur
- everyone will be happy if packets are sent at rate a little below link BW
- though merely gentlemen's agreement
  - combined with TCP, widely spread
  - can not break the agreement unless both sides cooperate

# Practice of Congestion Control by TCP

- control TCP rate according to congestion situation of the Internet
  - if congested, reduce transmitter's window size
- what is congestion?
  - packet drop
- packet drop is detected by timeout or acknowledge number not increasing

# Congestion Avoidance with UDP

- no general theory
  - basically impossible for streaming
- how to detect packet drop
  - timeout? depends on application protocol
- how to act against packet drops?
  - retransmit the packets
    - increase retransmission interval if packet drops are frequent



Layering Structure of the Internet

# DNS (Domain Name System)

## (rfc1034, 1035)

- loosely coupled distributed database to map domain name to its value (IP address etc.)
  - domain name consists with labels separated by “.” representing hierarchy
    - nocom830.hpcl.titech.ac.jp
- one of the most important application (?) of the Internet
- mainly use UDP (TCP is sometimes used)
  - at port number 53

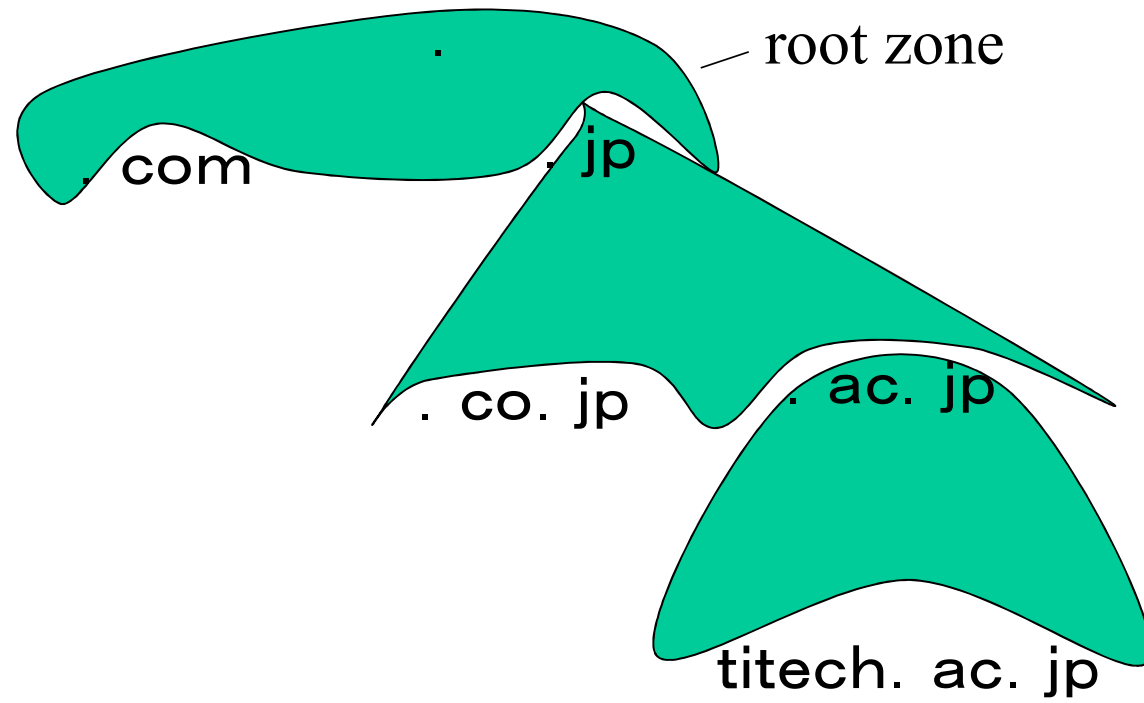


# Domain Name and Name Servers

- domain name forms tree with “.” as root
  - the last “.” may be omitted
- delegate authority through tree structure
  - nocom830.hpcl.titech.ac.jp
  - IANA, JPNIC, TIT, Ohta lab.
- each part of tree under authority (zone) is served by multiple name servers
  - loosely coupled distributed database

# Order of Name Servers

- parent zone offer name server information of child zones
- primary server (only one)
  - have master information of a zone
- secondary server (at least one)
  - obtain zone information from other servers
  - periodically check zone information of other servers
    - if updated, perform zone transfer (TCP)



# Resolver and Name Server

- resolver
  - mechanism to look up DNS database upon request from upper layer applications
    - may be included in applications
  - exchange packets with name servers and resolvers
  - may cache data
- the first server to ask is statically configured

# Recursive Resolvers

- resolvers to send several queries if the first query does not result in full final response
- NS, CNAME etc. makes additional queries necessary
- non-recursive resolvers are stub resolvers
  - stub resolvers are statically configured with addresses of recursive resolvers
    - stub resolvers can share concentrated cache of a recursive resolver

# Information Held by Nodes

## RR (Resource Record)

- RR have fields of
  - class
    - not used
  - type
    - IP address, mail server, etc.
  - TTL
    - time to live for cache
  - data
    - data depending on type

# RR Types for DNS itself

- SOA (start of authority)
  - zone management information
- NS (name server)
  - domain name of name server
- CNAME (canonical name)
  - alias
- (glue) A (address) or AAAA (IPv6)
  - IP address of name server

# SOA (Start of Authority) RR (1)

- specified at a root of a zone
  - primary name server
    - domain name of the primary NS of the zone
  - mail address of zone administrator
    - converting “@” to “.”
  - serial number
    - serial number of zone information



# SOA (Start of Authority) RR (2)

- refresh interval
  - interval to check serial number by secondaries
- refresh retry
  - retry interval after refresh zone transfer failure
- expire
  - secondaries zone information is valid until “expire” even if refresh zone transfer keep failing
- minimum TTL
  - minimum TTL of RRs in zone (default)

# Zone Transfer

- transfer zone information from primary to secondaries
  - over TCP
- may also be used between secondaries
  - secondaries are statically configured which servers to check
  - first, check serial number
    - if serial number is updated, perform zone transfer

# NS (Name Server) RR

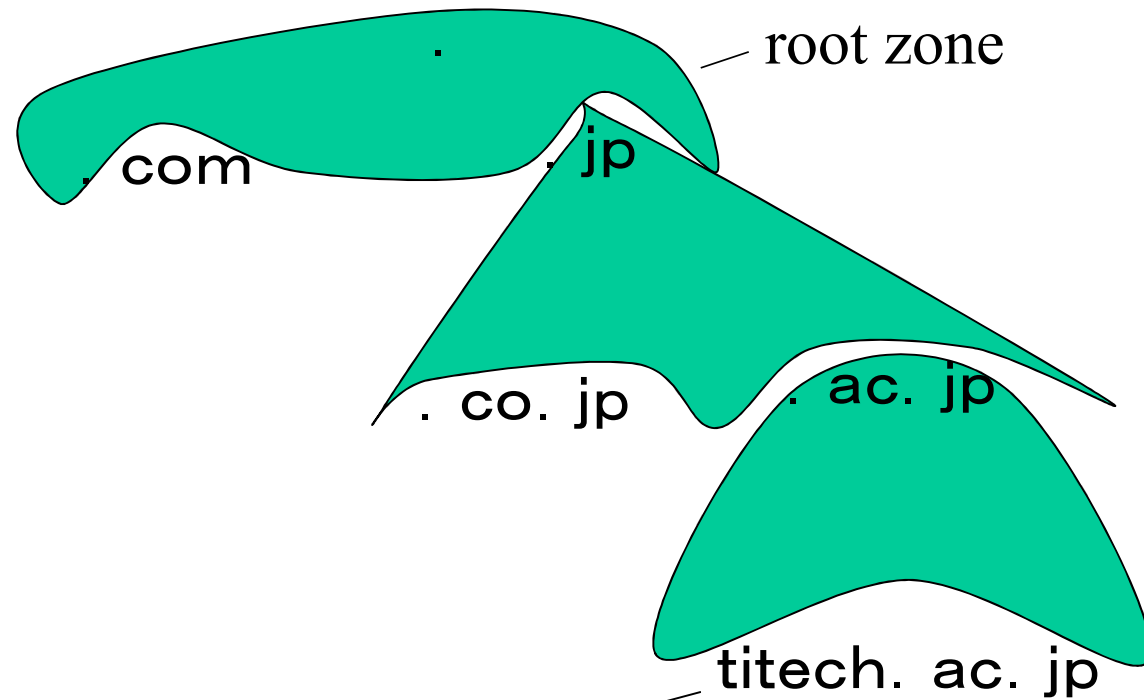
- specify domain name of NS
- at root of a zone
  - specify NSes of the zone
- at lower edges of a zone
  - specify NSes of child zones
  - glue A (AAAA) may also be specified

# CNAME (Canonical Name) RR

- specify canonical name
  - used at leaf node as alias

# Glue A (Addrees) RR

- if NS of a child zone is in or below the child zone
  - address of NS of the child zone can be resolved by asking NS of the child zone
    - loop!
- address of NS of the child zone may also be specified in parent zone



titech. ac. jp	NS	ns1. noc. titech. ac. jp
ns1. noc. titech. ac. jp	A	131. 112. 125. 4

name server and glue A

# RR Types for Applications

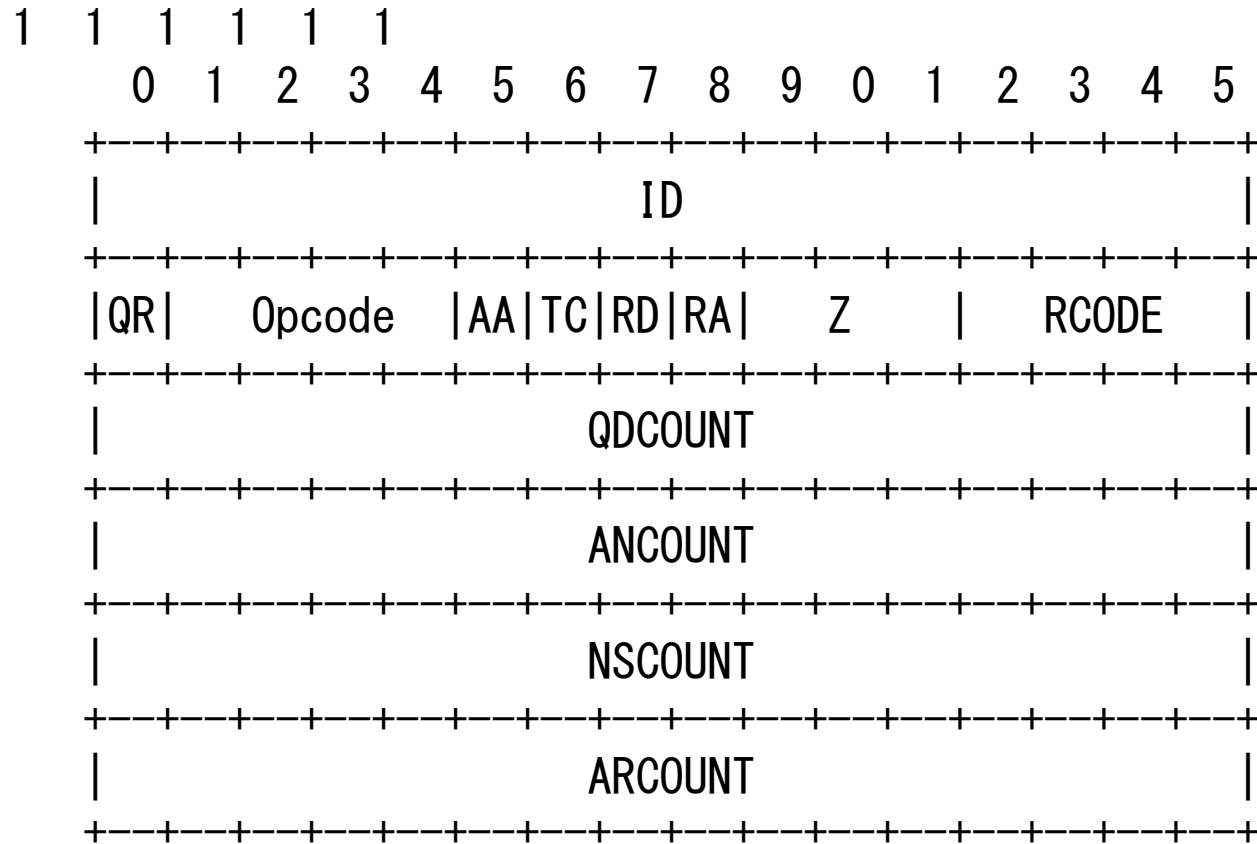
- A
  - IPv4 Address
- AAAA
  - IPv6 address
- MX
  - priority and domain name of a mail server
- PTR
  - reverse lookup from IP address to host name

# Message Format (over UDP)

+-----+	
Header	
+-----+	
Question	the question for the name server
+-----+	
Answer	RRs answering the question
+-----+	
Authority	RRs pointing toward an authority
+-----+	
Additional	RRs holding additional information
+-----+	



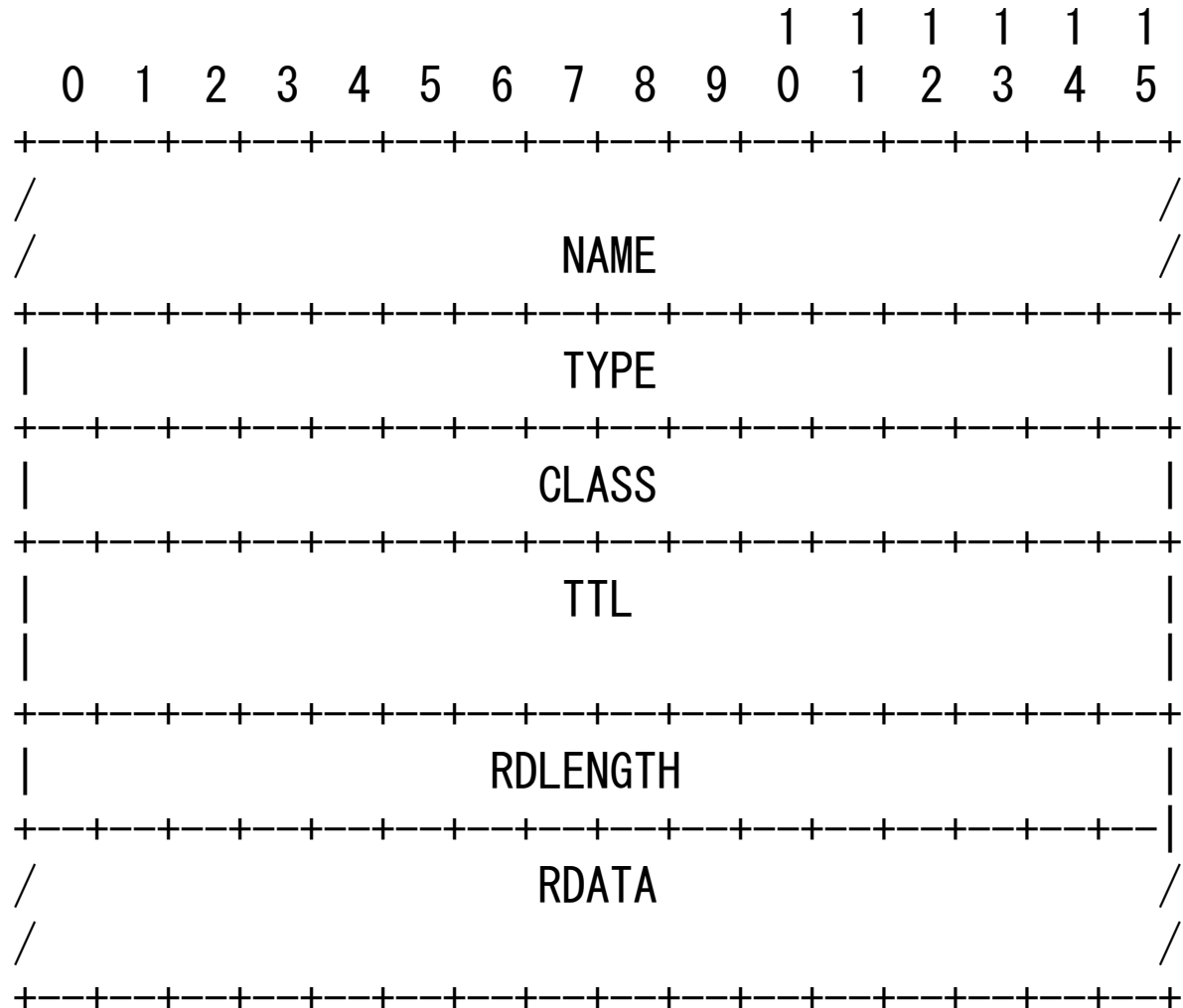
# Header Format



# Question Section Format

0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
/	QNAME														/
/															/
	QTYPE														
	QCLASS														

# Resource Record Format



# How DNS Works

- applications
  - ask resolver perform query
- resolver
  - may maintain cache
  - ask queries to name servers
  - may ask again for reply with intermediate result
    - stub resolvers ask recursive resolver for final result
- name server
  - reply queries using zone information of NS
  - may also have resolver function

# DNS and Cache

- resolver may cache old queries
  - reduce load on name servers
- cache interval is specified by TTL of RRs
  - short TTL for frequently updated information
  - long TTL for seldom updated and often queried information
    - especially information on “.”
- additional section may be filled from cache

# Resolver Operation

- if answer is in cache
  - reply the answer
- if not
  - ask some name server
- if referred to other name server
  - ask again to the name server
- if answer is CNAME and query type is not CNAME
  - ask again with the canonical name

# Name Server Operation

- if queried node is in zones served by the NS (or below them)
  - reply the answer (or referral)
- if answer is in cache
  - reply the answer
- OW,
  - reply referral to name servers of lowest possible (at worst, **but too often**, “.”) upper zone

# Message Size of DNS

- All IPv4 hosts are required to accept at least 576B packets
  - IPv4 header is at most 60B
- UDP payload can be 508B (not 512B) long
  - though DNS message can be 512B long!
- long DNS data can not be sent over UDP
  - fragmentation with UDP? ordering?  
retransmission?
    - instead, just rely on TCP



# DNS and the E2E Principle

- DNS globally identify (IP addresses of) end systems by ASCII strings
- tree structure of DNS is, in general, unrelated to network topology
  - identification local to end systems is impossible
  - must have some NS in the network
    - location of NS is unrelated to end systems in zones served by the NS
  - is against the E2E principle
    - NS may fail even if end systems are alive

# Implication of Violation of E2E Principle by DNS

- violation of E2E principle means
  - vulnerable to server down
    - DNS specifies zones **must** have multiple servers
      - end to end multihoming
  - load concentration on name servers
    - especially on root NSes
    - should use light weight protocol of UDP
    - anycast root server?
      - have **a lot of** root servers sharing same IP address and choose the nearest one by routing system

# Extensions to DNS

- DNSIND
  - IXFR (Incremental Zone Transfer)
  - Notify
  - DYNUPD (Dynamic Update)
- Secure DNS
- AROOT (Anycast Root Servers)

# IXFR (Incremental Zone Transfer) (rfc1995)

- when a zone is updated
  - so far, all the zone information is transferred
  - want to transfer difference to previous versions only
- extend zone transfer format
  - separate differences by SOAs
- may use UDP, if difference is small

# Example of IXFR

JAIN.AD.JP. IN SOA serial=3  
JAIN.AD.JP. IN SOA serial=1  
NEZU.JAIN.AD.JP. IN A 133.69.136.5  
JAIN.AD.JP. IN SOA serial=2  
JAIN-BB.JAIN.AD.JP. IN A 133.69.136.4  
JAIN-BB.JAIN.AD.JP. IN A 192.41.197.2  
JAIN.AD.JP. IN SOA serial=2  
JAIN-BB.JAIN.AD.JP. IN A 133.69.136.4  
JAIN.AD.JP. IN SOA serial=3  
JAIN-BB.JAIN.AD.JP. IN A 133.69.136.3  
JAIN.AD.JP. IN SOA serial=3

# Notify (rfc1996)

- notify zone update to other servers
- so far, secondaries periodically (with SOA refresh interval) check other servers
- notify is sent from primary side to secondaries

# DYNUPD (Dynamic Update)

- protocol to modify zone information
  - so far, zone files are manually edited
- unnecessarily complicated
- security!
  - DYNUPD with secure DNS?
- DNSIND
  - DYNUPD->Notify->IXFR

# AROOT (Anycast Root Servers)

- a lot of load concentrate on root servers
- root servers are the key to DNS security
- operate root servers everywhere
  - the number of root server addresses is limited below 13 (with IPv4 only), because of maximum message size of 512B
  - with anycast, a single IP address is shared by many servers
    - the nearest (w.r.t. routing metric) server will be used



# Security of Plain DNS

- plain DNS is weakly secure
  - secure if infrastructure (related NSes and ISPs) is trustworthy
    - if root NS and ISPs to/from it is trustworthy, reply from the root NS is trustworthy
    - if some NS and ISPs to/from it is trustworthy, reply from the NS (answer or referral to child NS) is trustworthy
  - though 16bit ID to match queries and replies is very weak

# Attack on Plain DNS

- plain DNS is very weakly secure
  - ID to match request and reply is 16bit long
    - attacker causes a resolver issue repeat same queries
    - attacker send false reply to the resolver
      - attacker may also perform DOS attack to suppress legitimate replies
  - because of birthday problem,  $2^{16/2}$  attacks have good chance for ID of some query and false reply matches

# Secure DNS (rfc2535)

- true (?) security to DNS by PKI
- zone hierarchy of DNS as CA hierarchy
  - everyone share public key of root zone
  - public key of child zone is signed by private key of parent zone, recursively
- not really deployed at all
  - operation too complicated
    - why?

# What is Security?

- secrecy
  - hide information from third parties
  - encryption
- authentication
  - certify permission
- mutually related
  - information for authentication must be secret
  - secret information may be offered to authenticated person

# Methods of Encryption

- by shared secret key
  - encrypted text= $E(\text{plain text}, \text{secret key})$
  - plain text= $D(\text{encrypted text}, \text{secret key})$
- by public key
  - encrypted text= $E(\text{plain text}, \text{public key})$
  - plain text= $D(\text{encrypted text}, \text{secret key (private key)})$

# Methods of Authentication

- use hash function (pseudo random number)
- by shared secret key
  - authentication info= $H(\text{plain text}, \text{secret key})$
- by public key
  - authentication info= $A(H(\text{plain text}), \text{private key})$
  - $D(\text{authentication info}, H(\text{plain text}), \text{public key})$

# Is Public Key Cryptography Better than Shared Key One?

- different secret key must be shared with different peers
  - $N^2$  keys necessary for  $N$  parties
- a public key may be announced to all peers
  - only  $N$  keys necessary
- what if, the announcement is forged?
  - introduce CA (Certificate Authority) to sign public keys of parties and let all the parties share public key of the CA

# Security of Phone Network

- assume phone companies are trustworthy
  - call is connected to peer specified by phone number
  - no wiretapping en route
  - can identify peer by phone number
    - rely phone number of peer provided by telco
    - phone number told be peer is not reliable
      - can be confirmed by calling back
- trustworthiness is just an assumption

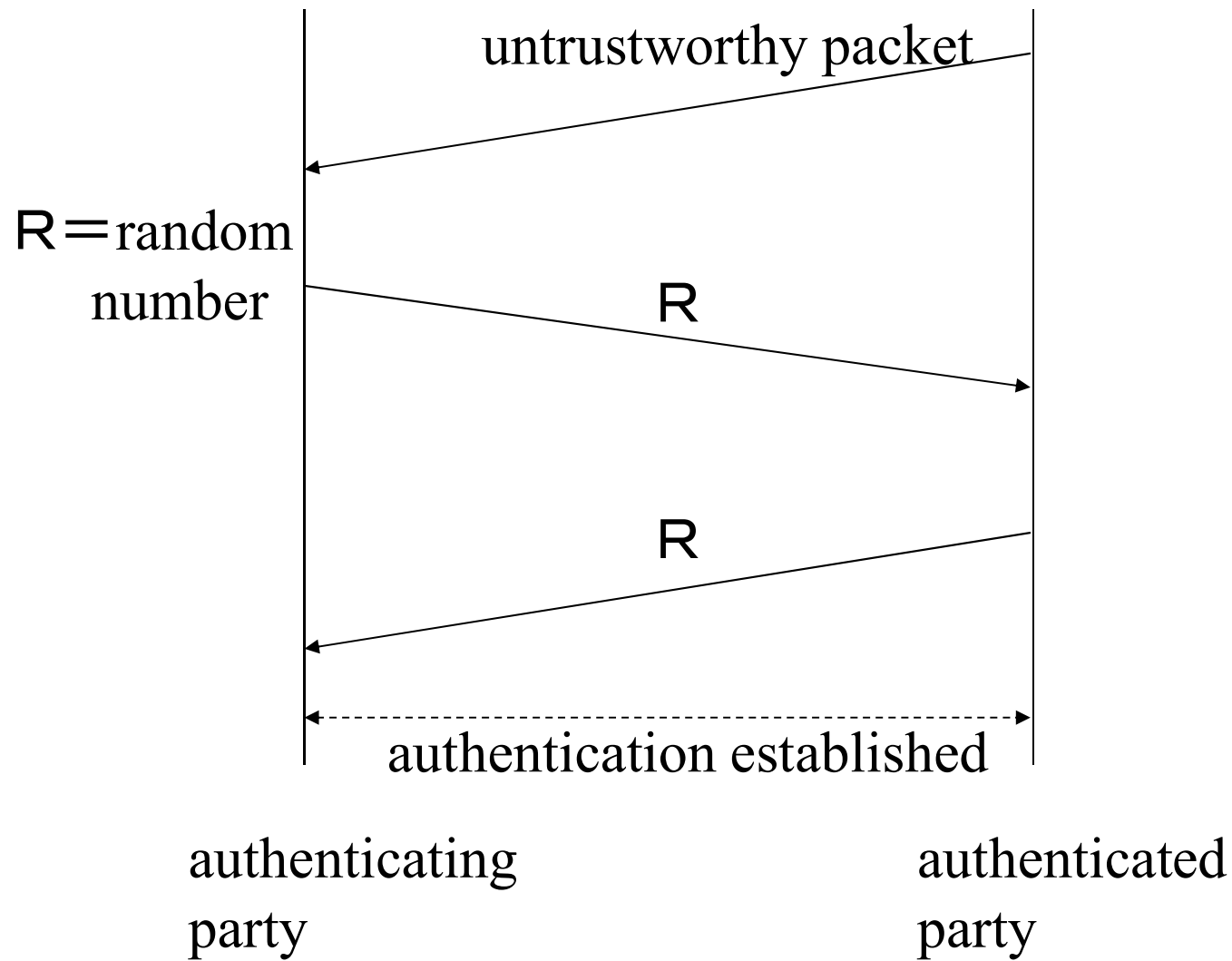


# Weak Security

- security relying on infrastructure offered by third parties
  - if ISPs are reliable, packet is delivered to a host with the destination address, without being tapped or modified
  - ISPs may tap or modify packets
- similar to security of phone network
  - telcos may tap or modify conversation
- similar to security of PKI

# Weak Security of the Internet

- assume ISPs are trustworthy
  - packet is sent to peer specified by IP address
  - no wiretapping en route
  - can identify peer by IP address
    - source IP address in IP header is not reliable
      - can be confirmed by handshaking (random number sent to peer's IP address is sent back)
- trustworthiness is just an assumption



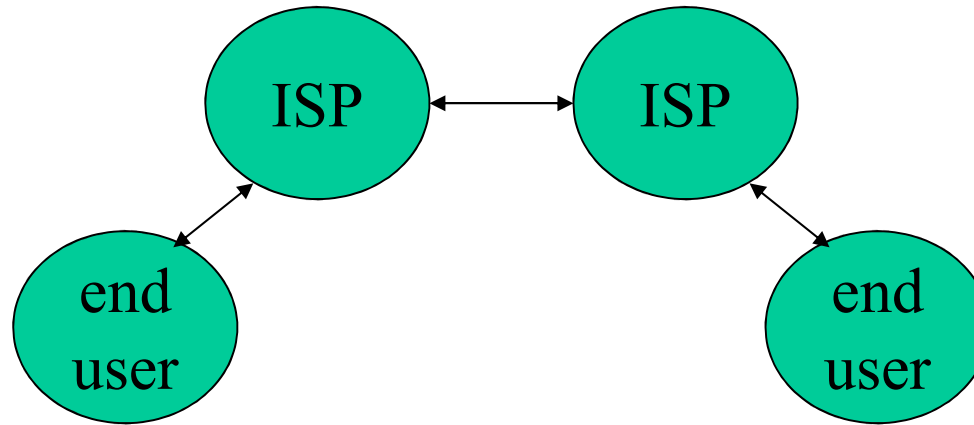
authentication of IP address with weak security

# True (Strong) Security?

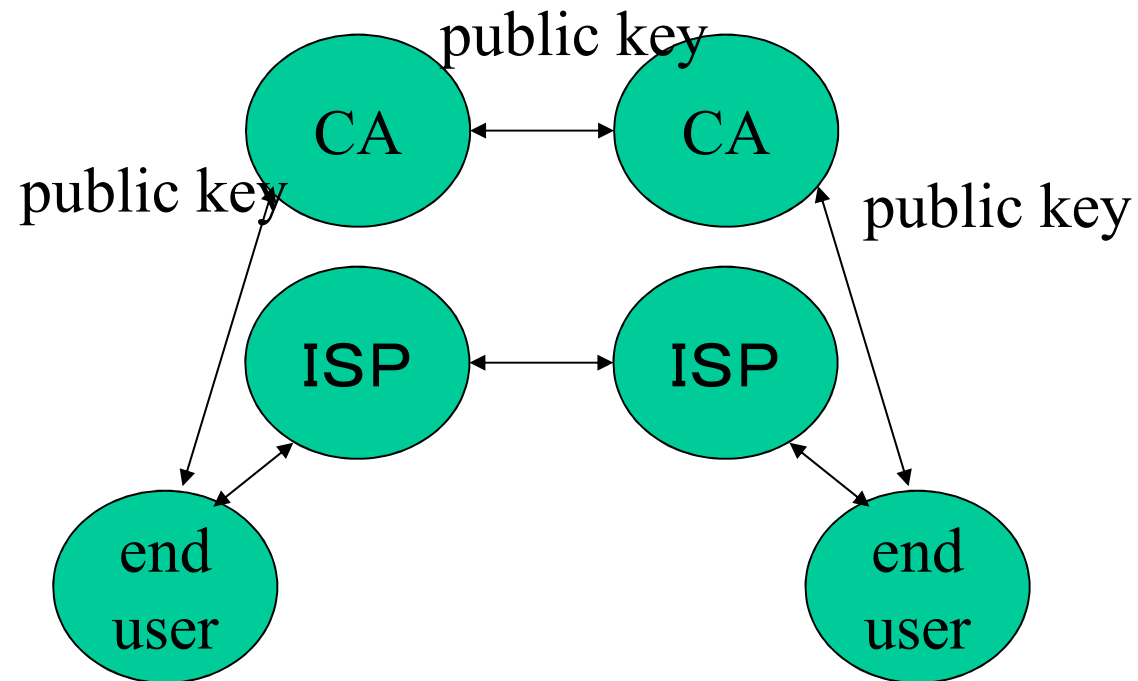
- securely share secret information (key)
- how to share the key?
  - PKI?
    - public key infrastructure using tree of CAs (Certificate Authorities) as trusted third parties
      - CA authenticate public key of end users and child CAs
  - by DNS?
    - to announce public key reliably by DNS
      - DNS information must be authenticated

# Isn't Public Key Cryptography Meaningless?

- the number of shared key ( $N^2$ ) is not a problem
- shared key cryptography scales with KDCs (Key Distribution Centers) just as CAs
  - though communication with KDC is necessary for every transaction
    - communication over the Internet is free
- CA is assumed to be a Trusted Third Party
  - weak security no different from ISPs as TTPs



weakly secure communication trusting ISPs



weakly secure communication trusting not ISPs but CAs

# Credit and the End to End Principle

- cryptography is a tool to carry credit
- credit is formed by direct (face to face, end to end) communication
  - secrete key may be shared at the same time
  - CAs do not offer credit
  - CAs are intermediate intelligent entities and is useless
- the E2E security needs direct credit formation and key sharing between related parties

# Wrap Up

- the Internet is information/communication infrastructure
  - not applications (not web or e-mail)
- the E2E principle is basis of the Internet
- IP, **the datagram**, is the protocol of the Internet
- UDP is light weight
- DNS is basically over LW UDP
- public key cryptography is merely weakly secure