Introduction to Computer Networks

CS640

Infrastructure Services

https://pages.cs.wisc.edu/~mgliu/CS640/F22/

Ming Liu
mgliu@cs.wisc.edu

Today

Last lecture

- How to take advantage of in-network support for TCP efficiency improvement?

Today

- What are infrastructure services used for?

Announcements

- Lab4 is due 12/02/2022, 11:59 PM
- Lab5 is due 12/14/2022, 11:59 PM
- Final exam: Dec 17, 2022 5:05 PM – 7:05 PM @Engineering Hall 1800

Q: What are infrastructure services used for?

Q: What are infrastructure services used for?

A: Provide network-assisted functionalities

- #1: Domain Name System (DNS)
- #2: Simple Network Management Protocol (SNMP)

Q: Why DNS?

Motivation: Naming Hosts

Thus far we have identified hosts using IP addresses and MAC addresses

- Hard for humans to remember these identifiers

Want to assign human-friendly names to hosts

- But routing still needs IP addresses
- Need a way to define and lookup the mapping between a hostname and an IP address

Naive Approach

Early Internet: a file mapping IP address to hostnames was manually updated and manually copied to all hosts in the Internet

- Problem: does not scale
- Still useful for small local networks – look at the /etc/hosts file on a CS dept. machine

Domain Name System (DNS) Overview

#1: Distributed name resolution system

- Many name servers (NSs) are distributed throughout the Internet – in ISPs, in Campus/ Enterprise networks, in department networks, etc.

#2: Domain names (DNs) are hierarchical

- A single NS doesn’t need to store the name for every host in the Internet

Domain Name System (DNS) Overview

#3: DNs can be mapped to IPv4 addresses, IPv6 addresses, and other DNs

- Mapping can be changed over time, or based on other factors (e.g., geo location)

#4: Queries are issued to a sequence of NSs

- Each knows about a different part of the DN hierarchy
- Answers can be cached to avoid the overhead of frequent lookups

Domain Name Hierarchy

DNs are processed from right to left, with periods as the separator

- Rightmost name is at the top of the hierarchy, and leftmost is at the bottom

Example: cs.wisc.edu

- Top of hierarchy – edu (rightmost name)
- Bottom of hierarchy – cs (leftmost name)

Domain Name Hierarchy — TLDs

Top-level domain (TLD): rightmost name

- Original TLDs were designed for the US – edu, com, gov, mil, org, net
- Expanded to include TLDs for countries – uk, cn, etc.
- Expanded to address the high demand for .com – .biz, .info, .tv, .ai, etc.
- Recently expanded to include arbitrary TLDs
  - Lots of contention over who should have rights to a specific TLD

Domain Name Hierarchy — Subdomains

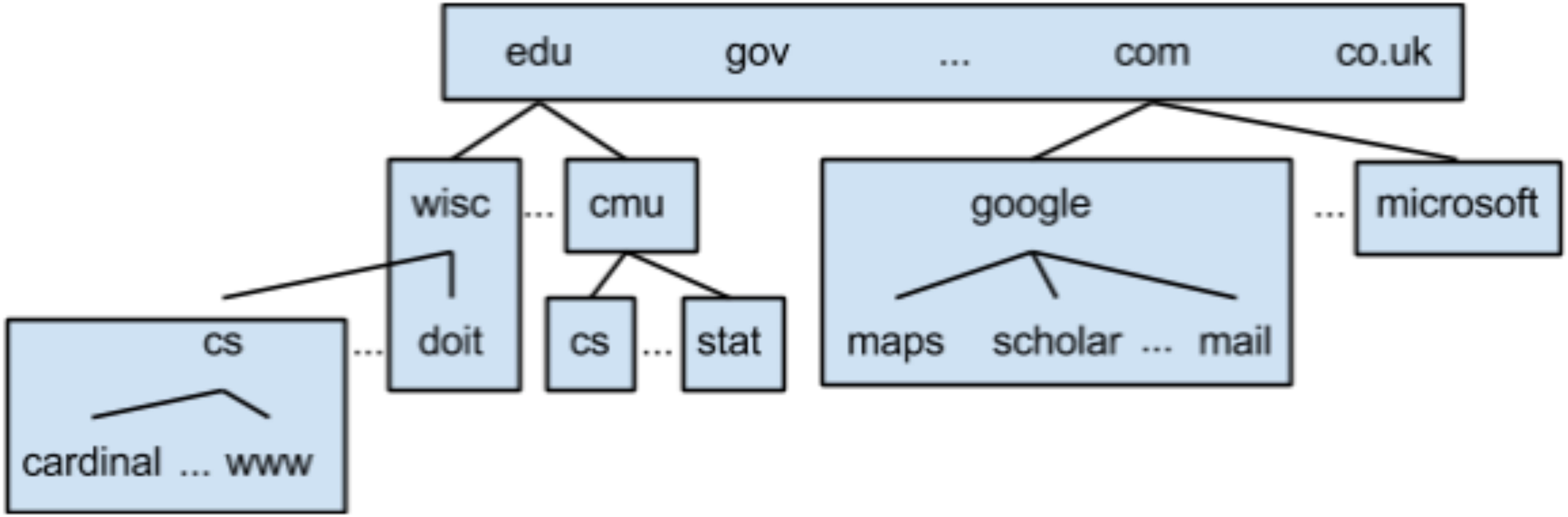
Second-level domain (SLD): Second from right

DN consisting of 3+ names is often referred to as a subdomain

- E.g., cs.wisc.edu is a subdomain of wise.edu

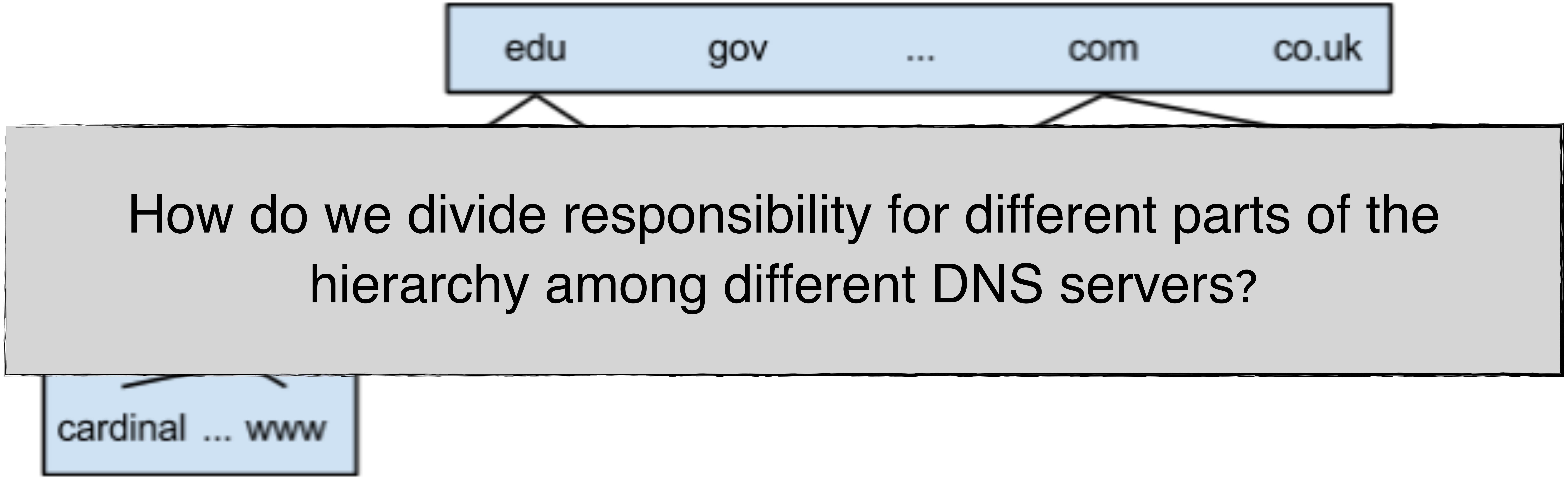
Domain Name Hierarchy — Example

Complete hierarchy only exists conceptually — no single DNS server stores the entire hierarchy



Domain Name Hierarchy — Example

Complete hierarchy only exists conceptually — no single DNS server stores the entire hierarchy



How do we divide responsibility for different parts of the hierarchy among different DNS servers?

Zones

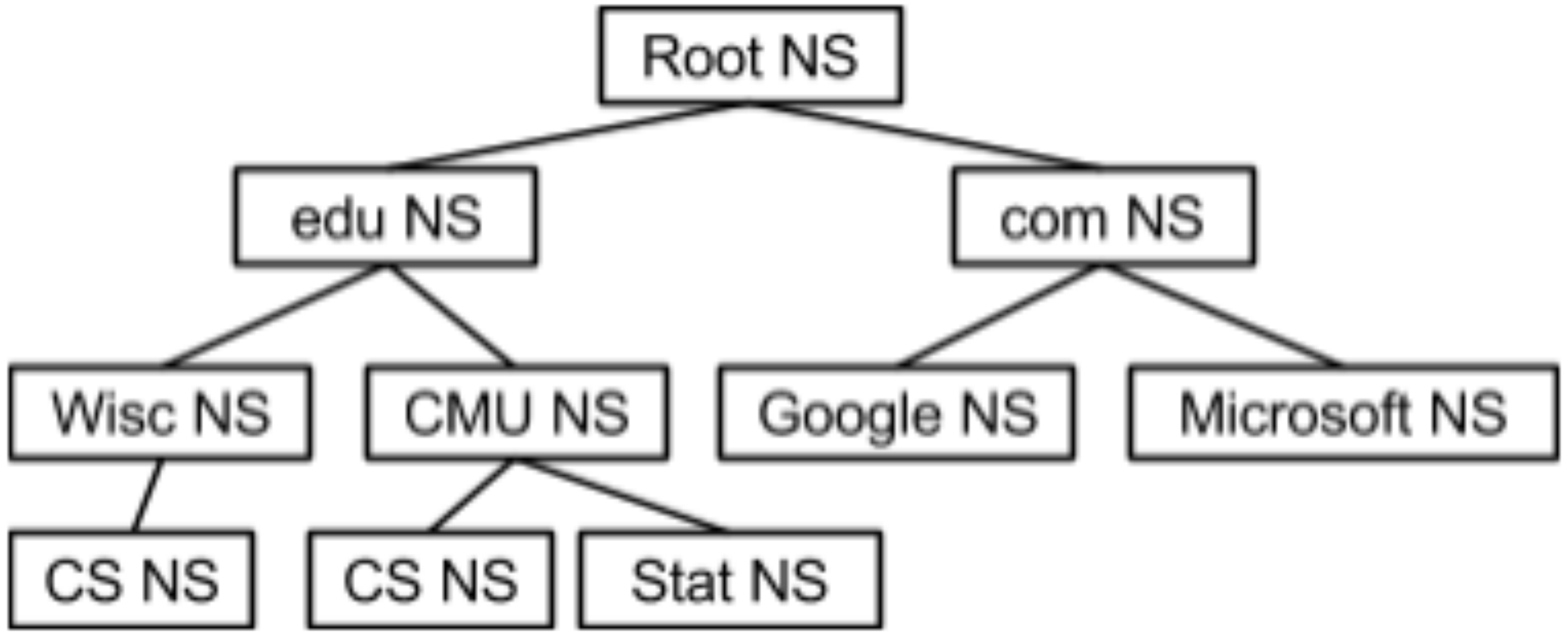
Zone: a portion of the hierarchy that is managed by an administrative entity

- Internet Corporation for Assigned Names and Numbers (ICANN) is responsible for the zone containing all TLDs
- CS department is responsible for the zone containing cs.wisc.edu and all subdomains (e.g., netlab-gatway.cs.wisc.edu, www.cs.wisc.edu, etc.)
- Google is responsible for the zone containing google.com and all subdomains (e.g., maps.google.com, scholar.google.com, mail.google.com, etc.)

Zones — Name Servers

Two or more name servers (NSs) are responsible for each zone

- Multiple NSs per zone to ensure availability in case of a failure
- Each store information for all domain names in the zone as records



DNS Records

Record has: name, type, value, and TTL

#1: Name = DN

#2: TTL specifies how long another DNS server can cache the record

DNS Records

#3: Types

- A: value is an IPv4 address
- AAAA: value is an IPv6 address
- NS: value is the domain name for a DNS server that is responsible for the zone containing the DN
- CNAME – value is another domain name for a particular host
- MX – value is domain name for a mail server that accepts message for the DN

DNS Records Example: Root NS

Root NS

- edu, NS, a.eduservers.net
- edu, NS, c.eduservers.net
- a.eduservers.net, A, 192.5.6.30
- c.eduservers.net, A, 192.26.92.30
- …

DNS Records Example: edu NS

edu NS

- wisc.edu, NS, adns1.doit.wisc.edu
- wisc.edu, NS, adns3.doit.wisc.edu
- adns1.doit.wisc.edu, A, 144.92.9.21
- adns3.doit.wisc.edu, A, 144.92.104.21
- adns3.doit.wisc.edu, AAAA, 2607:f388::a53:3
- cmu.edu, NS, NSAUTH1.net.cmu.edu
- NSAUTH1.net.cmu.edu, A, 128.2.1.8
- NSAUTH1.net.cmu.edu, AAAA, 2607:fb28::4

DNS Records Example: wise NS

wisc NS

- cs.wisc.edu, NS, dns.cs.wisc.edu
- cs.wisc.edu, NS, dns2.cs.wisc.edu
- dns2.cs.wisc.edu, A, 128.105.2.10
- dns2.cs.wisc.edu, A, 128.105.6.12
- …

DNS Records Example: cs NS

cs NS

- cardinal.cs.wisc.edu, A, 128.105.14.122
- www.cs.wisc.edu, A, 128.105.7.31
- cs.wisc.edu, MX, granite.cs.wisc.edu
- cs.wisc.edu, MX, obsidian.cs.wisc.edu
- granite.cs.wisc.edu, A, 128.105.6.24
- obsidian.cs.wisc.edu, A, 128.105.6.13
- netlab-gateway.cs.wisc.edu, A, 128.105.214.163

Name Resolution Algorithm

Step #1: Client contacts Local Name Server

- Local NS is provided to the client by DHCP or set in the client configuration
- Local NS contacts root name server

Step #2: Root NS provided NS & A record for NS that can resolve TLD

Step #3: Local NS contacts NS for TLD

Name Resolution Algorithm

Step #4: NS for TLD provides NS & A record for NS that can resolve SLD

Step #5: Local NS contacts NS for SLD

Step #6: NS for SLD provides A record for domain, or NS & A record for NS that can resolve domain

Name Resolution Algorithm

Step #4: NS for TLD provides NS & A record for NS that can resolve SLD

Step #5: Local NS contacts NS for SLD

Local DNS server will cache any records it receives

Name Resolution Example

Resolve <u>netlab-gateway.cs.wisc.edu</u> ?

Name Resolution Example

Resolve netlab-gateway.cs.wisc.edu ?

Resolve mail.google.com

- Google NS
  - mail.google.com, CNAME, googlemail.1.google.com
  - googlemail.l.google.com, A, 74,125.225.53
  - googlemail.l.google.com, A, 74.125.225.54

Name Resolution Discussion

Interactions# with NSs depending on

- How many parts there are to the DN (e.g., wisc.edu v.s. netlab-gateway.cs.wisc.edu)?
- How many levels in the name hierarchy are in the same zone (e.g., wisc.edu and doit.wisc.edu are in the same zone, while wisc.edu and cs.wisc.edu are not)?
- Whether there are CNAMEs that require contacting a different NS?
- What records the local NS has already cached?
- A single local name server is shared by many clients. Leverages benefits of caching and reusing DNS replies

Name Resolution Optimization

#1: NS can return different sets of records for different queries

- Use for load balancing or geo-based server selection

#2: Load balancing

- Assign short TTL to records
- Return different A records for each query for a DN – cycle through A records in weighted round-robin order; weight is based on server load

Name Resolution Optimization

#3: Geo-based server selection

- Used by content distribution networks (CDNs)
- CND's NS is configured with approximate geo-location of certain IP blocks
- The address of NS that issued the query is compared against IP blocks to determine the rough location of the client that issued the queries
- Based on location, CDN's NS returns a CNAME record whose value is DN for the server close to client

Name Resolution Optimization

Example: google.com

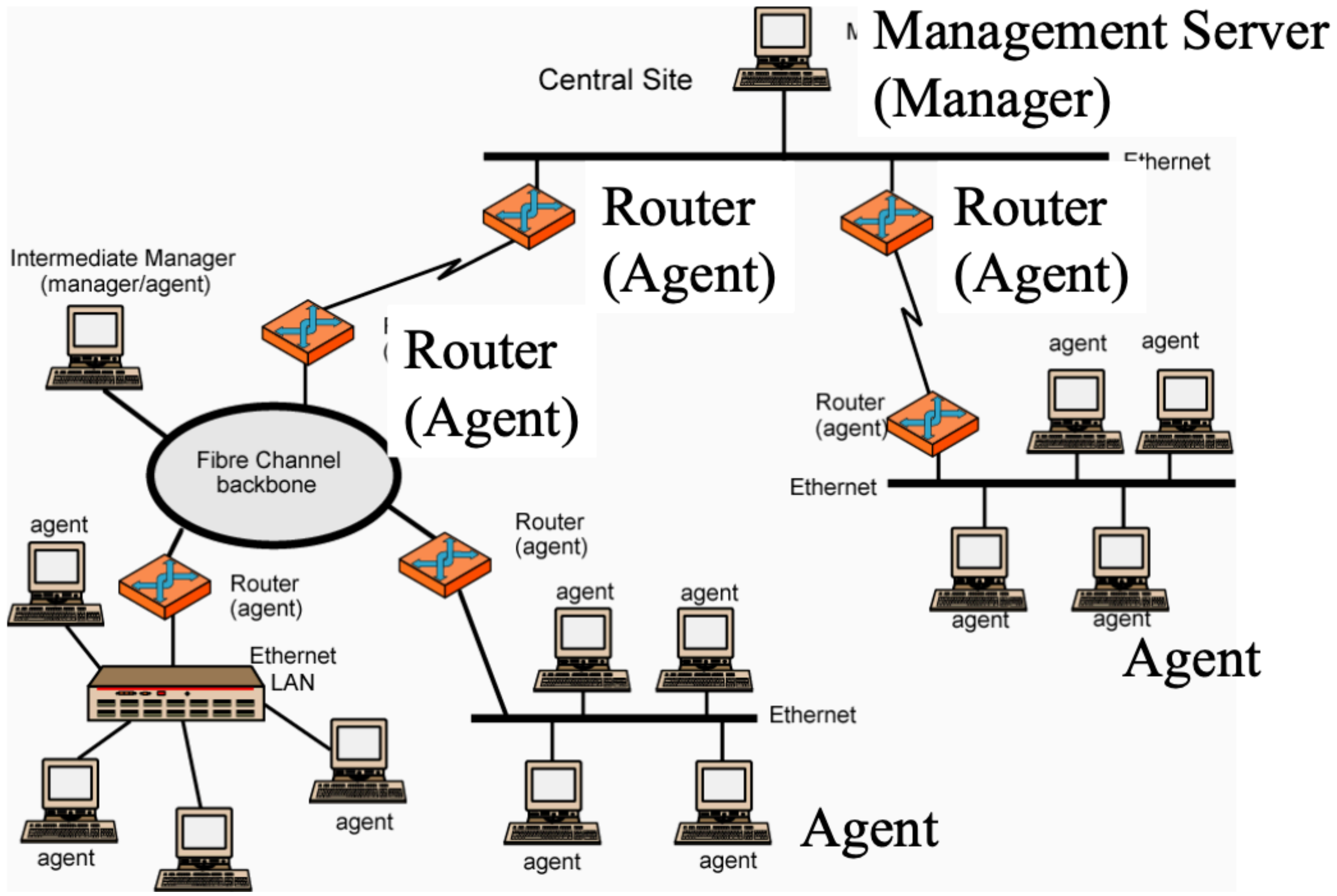
- #1: Local NS contacts root NS
- #2: Root NS provides NS & A records for com NS
- #3: Local NS contacts com NS
- #4: Com NS provides NS & A record for Google NS
  - E.g., ns1.google.com, 216.239.32.10
- #5: Local NS contacts Google NS
- #6: Google NS looks at the source IP for query and provides different addresses based on the estimated location of the source IP
  - From home (Charter): 64.15.120.52
  - From Milwaukee (AT&T DSL): 74.125.225.32
  - From Los Angeles: 74.125.239.161

Q: Why DNS?

A: Simplifying naming and addressing

Q: Why SNMP?

Q: Why SNMP?


Central Site
Management Server (Manager)
Ethernet
Router (Agent)
Router (Agent)
Intermediate Manager (manager/agent)
Router (Agent)
agent
agent
Router (agent)
Ethernet
Fibre Channel backbone
agent
Router (agent)
Router (agent)
agent
agent
agent
agent
Agent
Ethernet LAN
agent
agent
Ethernet
agent
agent
Agent

Q: Why SNMP?

Ticket #1: I cannot access the printer;
Ticket #2: I cannot access the file server;
Ticket #3: Why it took me 10mins to print a 1-page document?
Ticket #4: I sent an email to my colleague on another floor, which takes half an hour;

What is Network Management?

#1: Configuration management

- Keep track of device settings and how they function

#2: Fault management

- Dealing with problems and emergencies in the network

#3: Performance management

- How smoothly is the network running?
- Can it handle the workload it currently has?

Network Management Goals

Management interface must be

- Standardized
- Extendible
- Portable

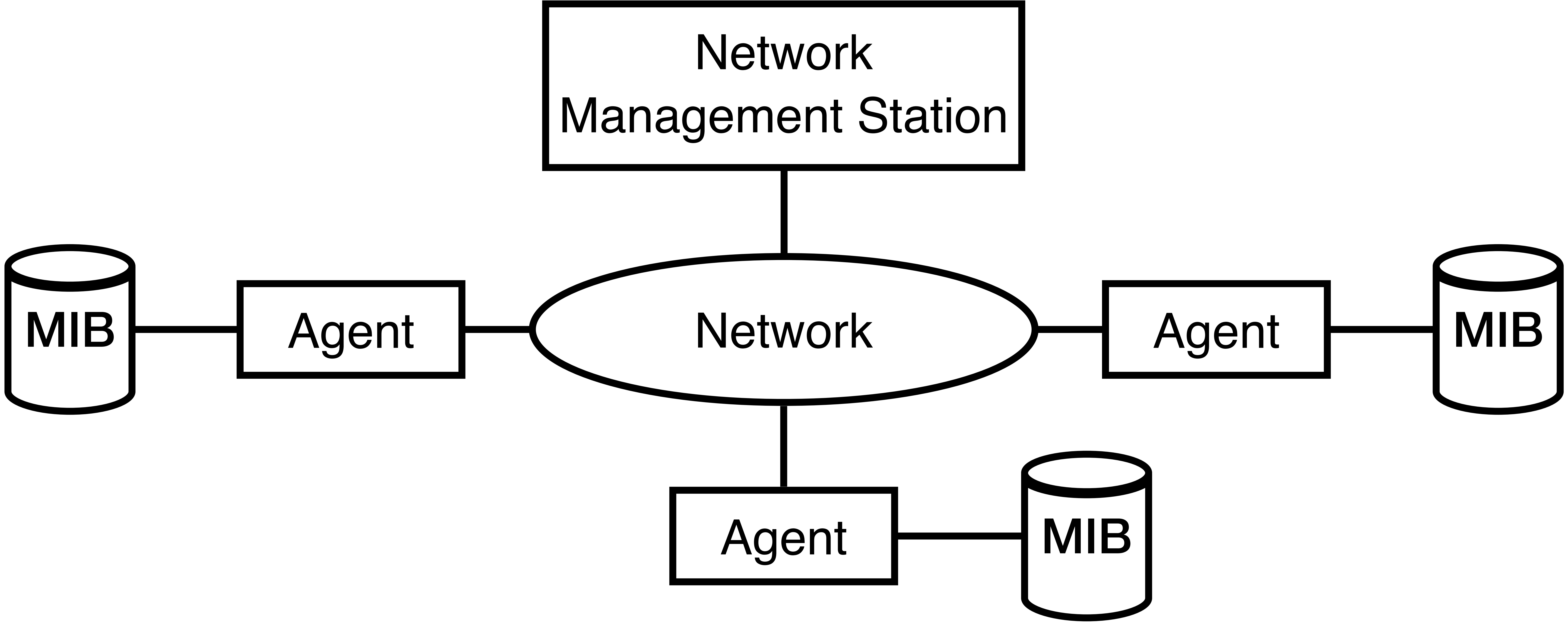
Management mechanism must be

- Inexpensive
- Implemented as software only

Network Management Overview

Management = Initialization, Monitoring, Control

Manager, Agents, and Management Information Base (MIB)



SNMP

Based on Simple Gateway Management Protocol (SGMP) — RFC 1028 — Nov. 1987

SNMP: Simple Network Management Protocol

- A tool (protocol) that allows for remote and local management of items on the network including servers, workstations, routers, switches and other managed devices

RFC 1058, April 1988

SNMP

Only five commands

Command	Meaning
get-request	Fetch a value
get-next-request	Fetch the next value (in a tree)
get-response	Reply to fetch operation
set-request	Store a value
trap	An event

Advantages of Using SNMP

#1: Standardized

#2: Universally supported

#3: Extendible

#4: Portable

#5: Allows distributed management access

#6: Light-weighted

Client Pull & Server Push

SNMP is a “client pull” model

- The management system (client) “pulls” data from the agent (server)

SNMP is a “server push” model

- The agent (server) “pushes” out a trap message to a (client) management system

Ports & UDP

SNMP uses User Datagram Protocol (UDP) as the transport mechanism for SNMP messages

Like FTP, SNMP uses two well-known ports to operate:

- UDP Port 161: SNMP messages
- UDP Port 152: SNMP trap messages

SNMP Details

SNMP Protocol

- Defines the format of messages exchanged by management systems and agents
- Specifies the GET, GetNext, Set, and Trap operations

Structure of Management Information (SMI)

- Rules specifying the format used to define objects managed on the network that the SNMP protocol accesses

Management Information Base (MIB)

- A map of the hierarchical order of all managed objects and how they are accessed

Nodes

Items in an SNMP Network are called nodes. There are different types of of nodes.

#1: Managed nodes

- Typically runs an agent process that services requests from a management node

#2: Management nodes

- Typically a workstation running some network management & monitoring software

#3: Nodes that are not manageable by SNMP

- A node may not support SNMP but may be manageable by SNMP through a proxy agent running on another machine

Community Names

Community names are used to define where an SNMP message is destined for

They mirror the same concept as a Windows NT or Unix domain

- Set up your agents to belong to certain communities
- Set up your management applications to monitor and receive traps from certain community names

SNMP Agents

Extendible agents

- Open, modular design allows for adaptations to new management data and operational requirements

Monolithic agents

- Not extendible
- Optimized for specific hardware platform and OS

SNMP Semantics

Structure of Management Information (SMI)

- Specifies the format used for defining managed objects that are accessed via the SNMP protocol

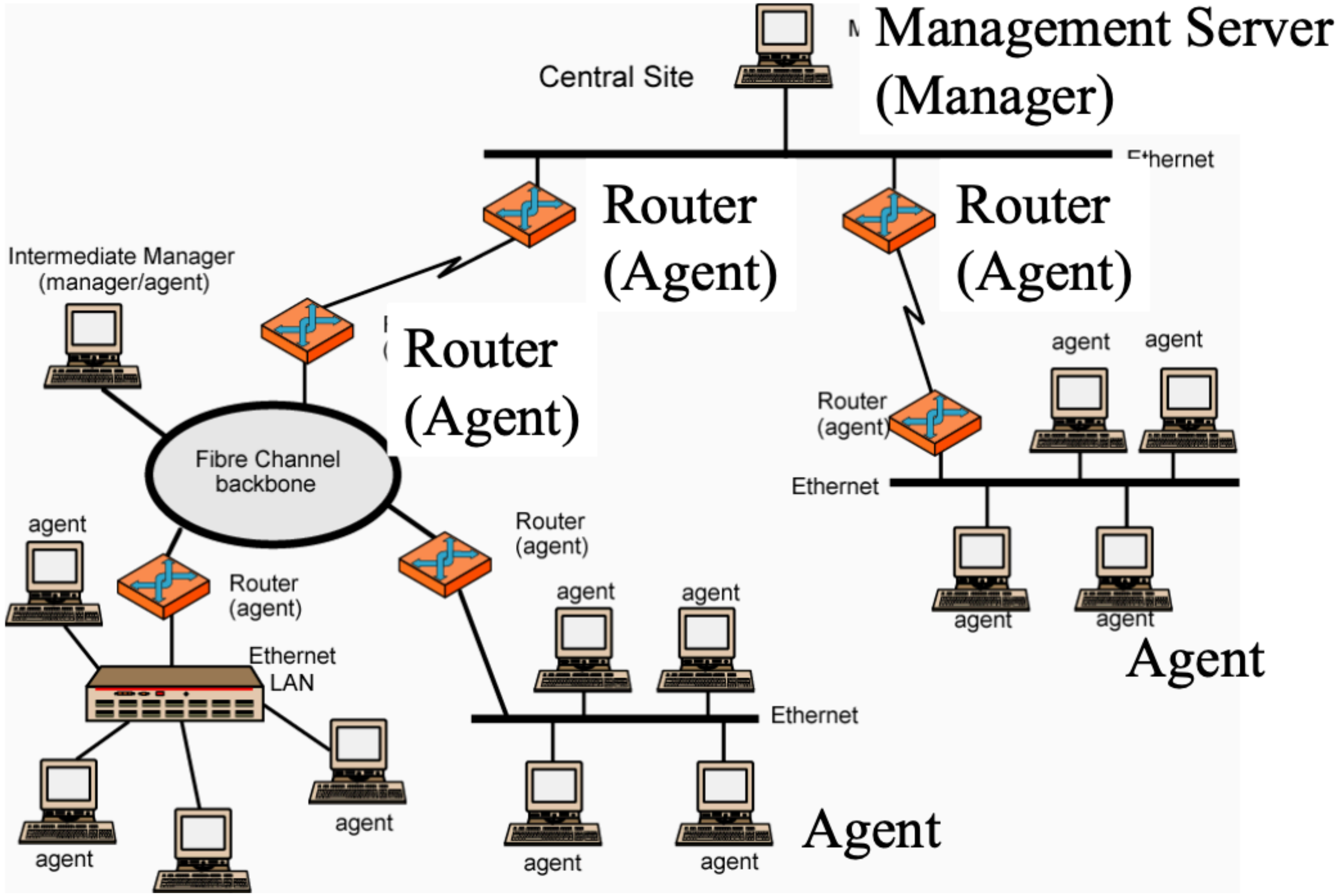
Abstract Syntax Notation One (ASN.1)

- Used to define the format of SNMP messages and managed objects (MIB modules) using an unambiguous data description format

Basic Encoding Rules (BER)

- Used to encode the SNMP messages into a format suitable for transmission across a network

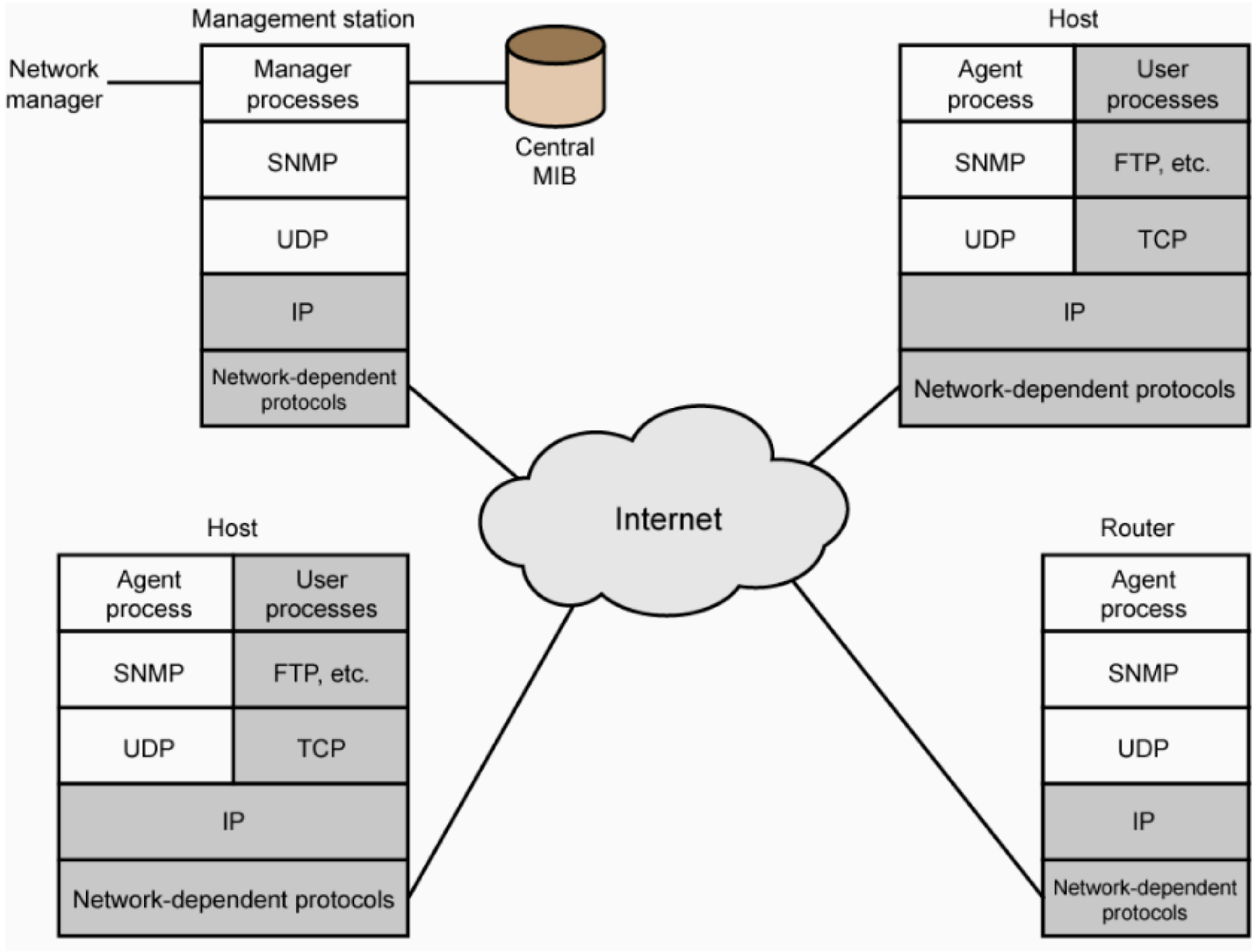
Example of Network Management



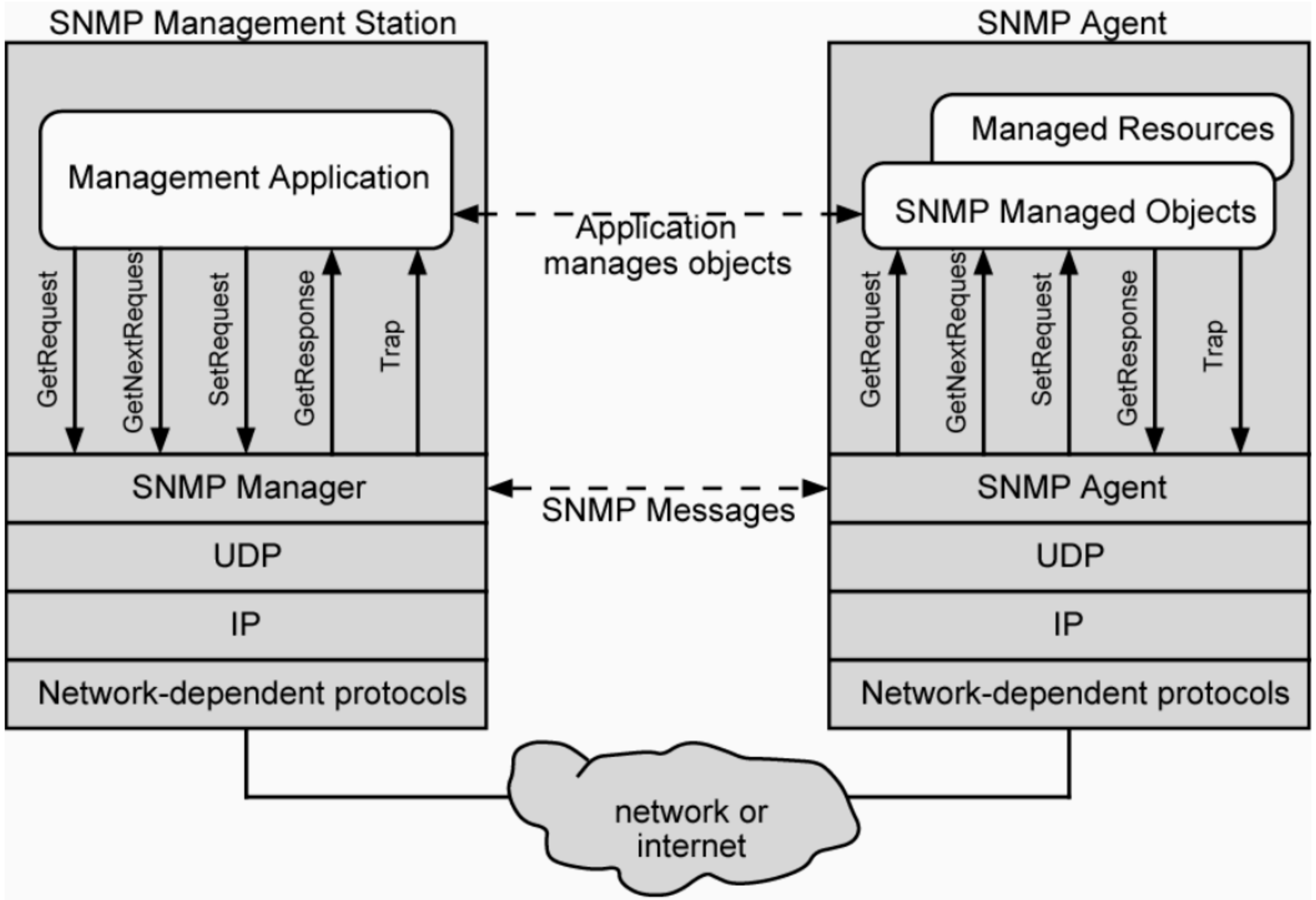
SNMPv1 Configuration

Manager sends request to UDP port 161

Agents send traps to UDP port 162



Role of SNMP v1



SNMPv2 (RFC 1441)

Improved security: authentication and integrity using the Data Encryption Standard (DES)

Information request => Multiple manager coordination

- Locking mechanism prevent multiple managers from writing at the same time

get bulk => Better table handling

Confirmation option for Traps

- Agents can ensure that trap was received correctly

New Error codes: noSuchName, badValue, readOnly

SNMPv3 (RFC 2570)

Security update of SNMPv2

Authentication: message authentication code with a shared secret key

Privacy: Encryption using a shared secret key

Access Control: Each manager can have a different set of read/write permission for various component of MIB

Q: Why SNMP?

A: Simplifying networking monitoring

Terminology

1. Host
2. NIC
3. Multi-port I/O bridge
4. Protocol
5. RTT
6. Packet
7. Header
8. Payload
9. BDP
10. Baud rate
11. Frame/Framing
12. Parity bit
13. Checksum
14. Ethernet
15. MAC
16. (L2) Switch
17. Broadcast
18. Acknowledgement
19. Timeout
20. Datagram
21. TTL
22. MTU
23. Best effort
24. (L3) Router
25. Subnet mask
26. CIDR
27. Converge
28. Count-to-infinity
29. Line card
30. Network processor
31. Gateway
32. Private network
33. IPv6
34. Multicast
35. IGMP
36. SDN
37. (Transport) port
38. Pseudo header
39. SYN/ACK
40. Incarnation
41. Flow
42. SYN flood
43. TCP Segment
44. Window
45. Advertised Window
46. Effective Window
47. TCP Reno
48. Duplicated ACK
49. Congestion Window
50. Congestion Threshold
51. Selective Acknowledgment
52. Active Queue Management (AQM)

Principle

1. Layering
2. Minimal States
3. Hierarchy
4. Mechanism/policy separation

Technique

1. NRZ Encoding
2. NRZI Encoding
3. Manchester Encoding
4. 4B/5B Encoding
5. Byte Stuffing
6. Byte Counting
7. Bit Stuffing
8. 2-D Parity
9. CRC
10. MAC Learning
11. Store-and-Forward
12. Cut-through
13. Spanning Tree
14. CSMA/CD
15. Stop-and-Wait
16. Sliding Window
17. Fragmentation and Reassembly
18. Path MTU discovery
19. DHCP
20. Subnetting
21. Supernetting
22. Longest prefix match
23. Distance vector routing (RIP)
24. Link state routing (OSPF)
25. Boarder gateway protocol (BGP)
26. Network address translation (NAT)
27. User Datagram Protocol (UDP)
28. Transmission Control Protocol (TCP)
29. Three-way Handshake
30. TCP state transition
31. EWMA
32. Sliding window
33. Flow control
34. AIMD
35. Slow start
36. Fast retransmit
37. Fast recovery
38. Nagle’s algorithm
39. Karn/Partridge algorithm
40. TCP Vegas
41. Bit-by-bit Round Robin
42. Fair Queueing (FQ)
43. Random Early Detection (RED)
44. Explicit Congestion Notification (ECN)
45. Domain Name System (DNS)
46. Simple Network Management Protocol (SNMP)

Summary

Today’s takeaways

#1: DNS is a distributed system that provides the mapping between IP addresses and human-friendly domain names

#2: SNMP is a distributed system that maintains execution statistics of in-network devices and endhosts

Next lecture

- Network Application (I)