Network Management

Robert Szabo

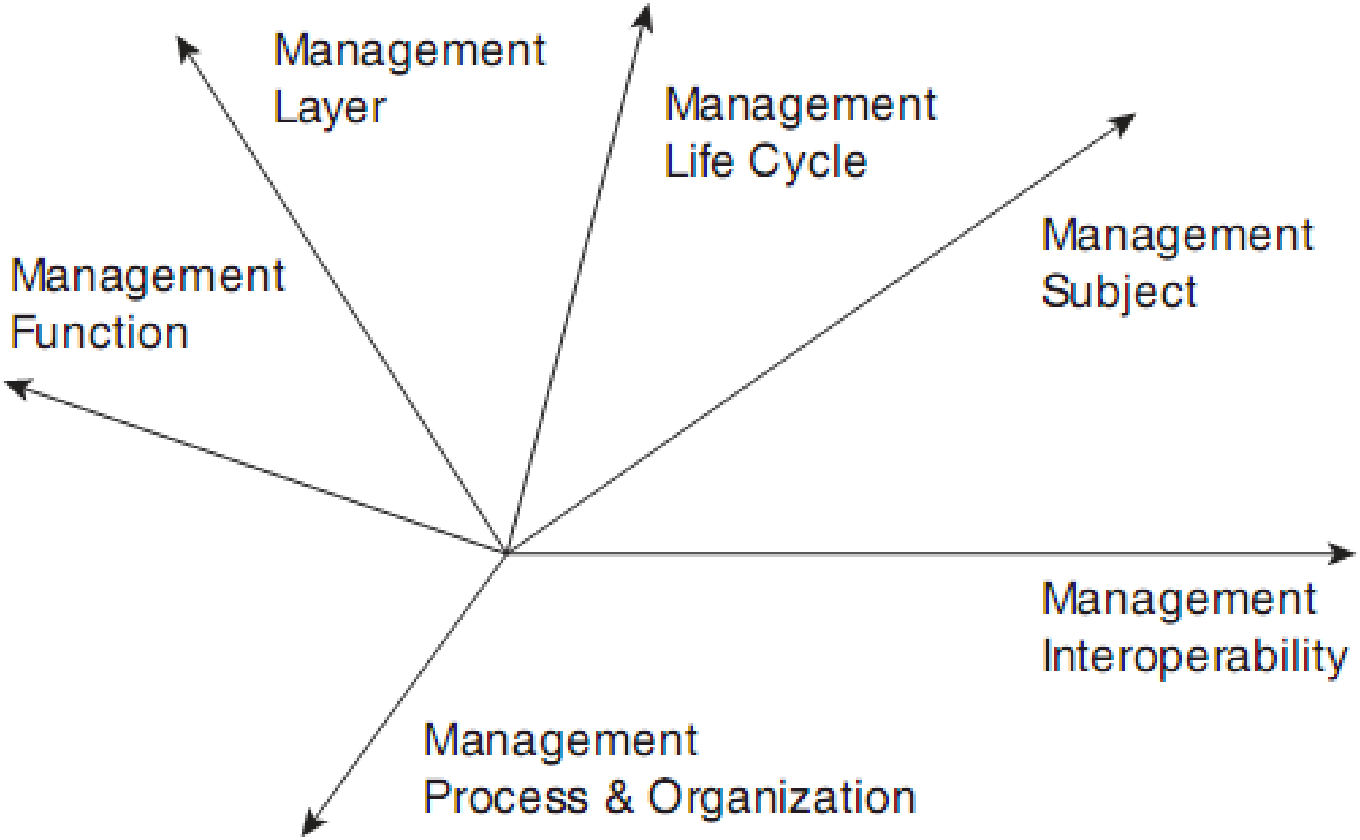
Outline

- ❑Introduction
- ❑SNMP
- ❑Command Line Interface (CLI)
- ❑syslog
- ❑Netconf
- ❑Netflow
- ❑Summary

Network management

- ❑Network management refers to the activities associated with running a network, along with the technology required to support those activities.
- ❑Analogies
  - ▪ Health care system
  - ▪ Throwing a party

MW Mgmt Dimensions



Frequently Used Acronyms

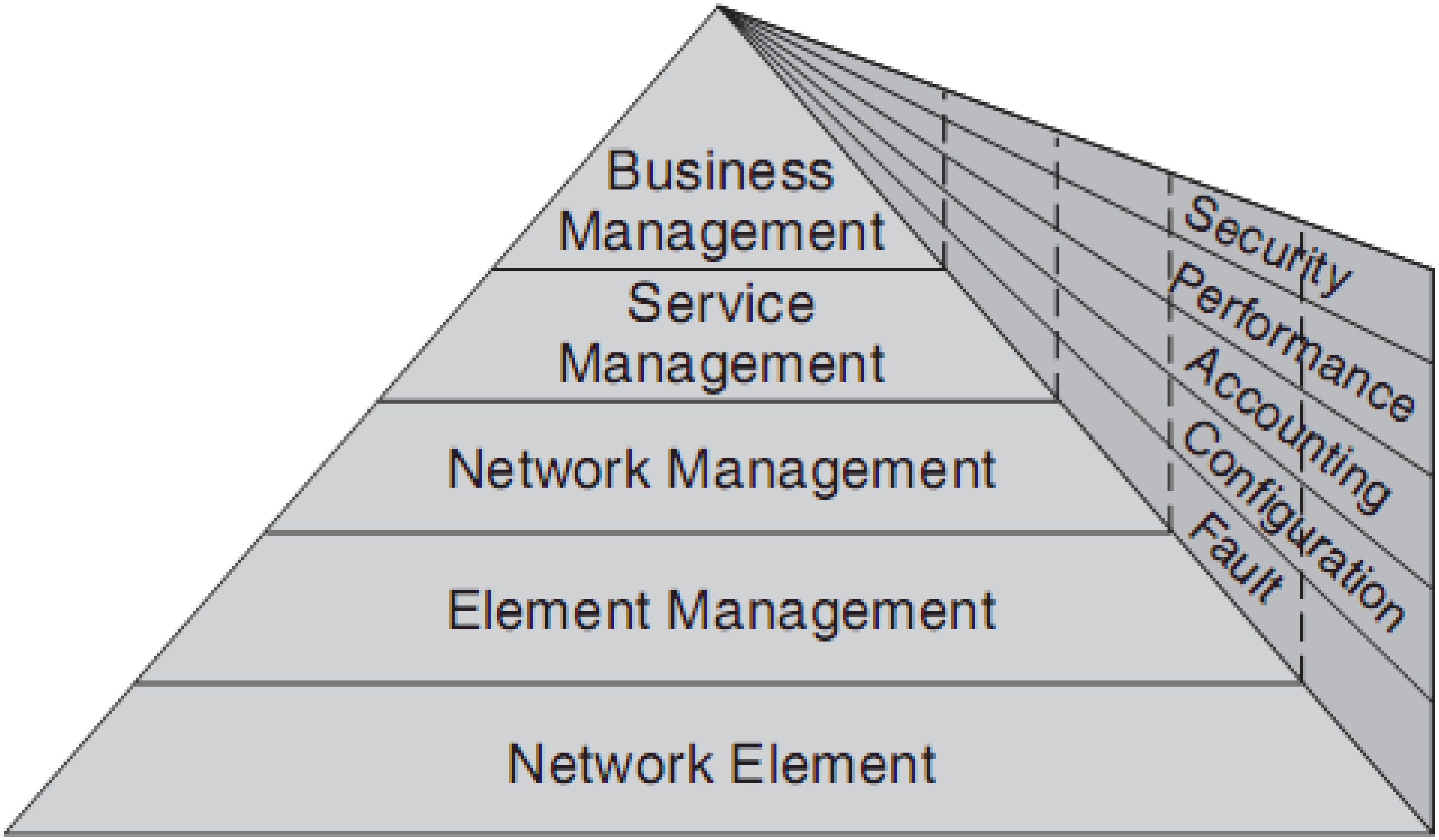
- ❑**Operation, Administration, Maintenance, Provision (OAMP or O&M)**
- ❑TOM, eTOM:
  Fulfillment -> Assurance -> Billing
- ❑International Telecommunications Union (ITU-T) Telecommunications Management Network (TMN)

Network Operation Center (NOC)

- □ Physical location?
- □ Might house the equipment itself
- □ Cabling and passive components
  - ▪ Need facilities to keep track of
- □ Several NOC acting as peers
  - ▪ Follow the sun
- □ Regional NOC

- □ Central Office (CO) vs NOC
  - ▪ CO terminates local lines!

TMN: Mgmt Layer Hierarchy & Functions



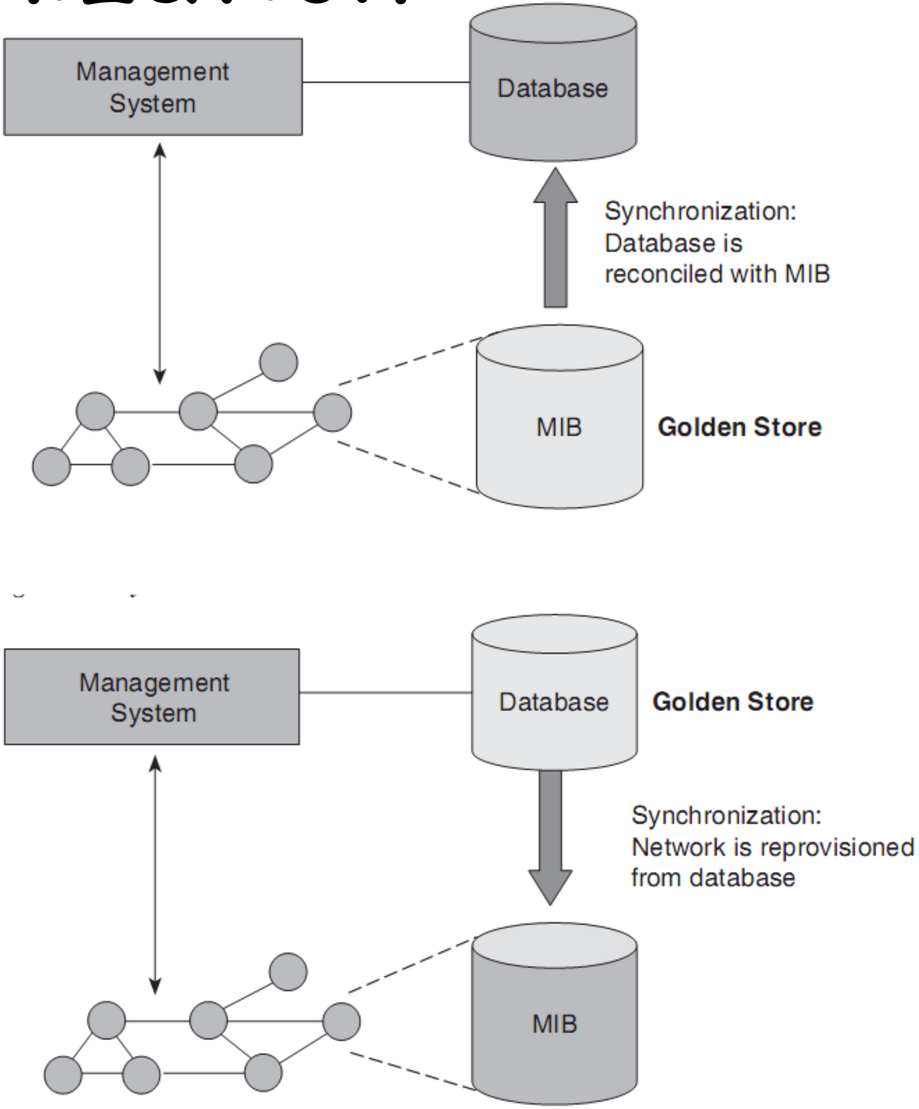
Auditing, Discovery, and Autodiscovery

- ❑ Auditing: find out **what actually has been configured**
  - ▪ non-configuration data with other FCAPS functions
- ❑ Discovery: to find out **what's in the network**
  - ▪ Inaccurate inventory records
  - ▪ Unrecorded changes in the network
  - ▪ Might be more efficient to discover the network instead of **entering data** into the mgmt system
  - ▪ **Mobile or roaming devices**

- ❑ (sometimes auditing is used for discovery, and autodiscovery for discovery ☺)

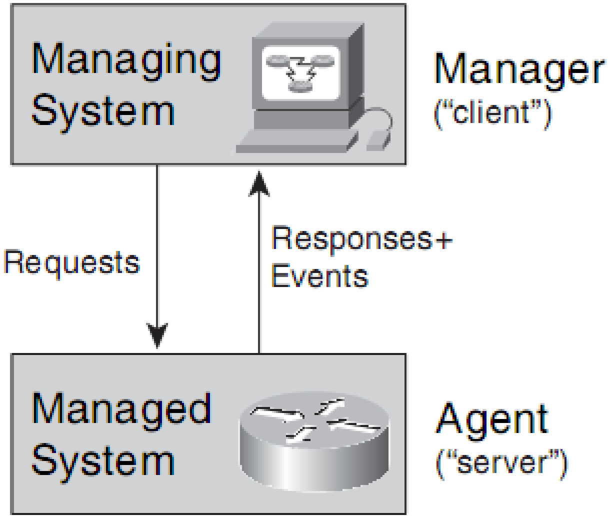
Synchronization

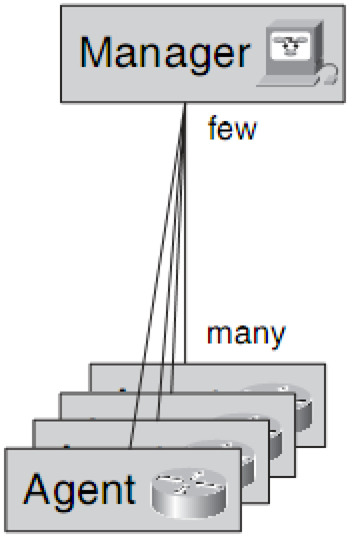
- Auditing is slow and inefficient
  - Must do it sometimes though
- Mgmt system **to cache information** about the network
  - Information that is relatively slow to change
- Problem
  - Two representations: network vs. mgmt system
    - **Network is the master** – enterprises' view
    - **Mgmt system is the master** – telco's view
    - Discrepancy reporting

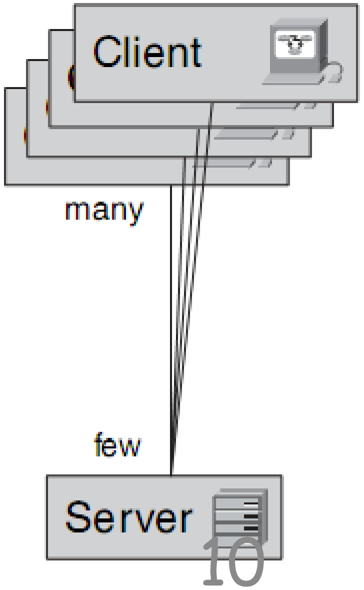

Management System
Database
Synchronization: Database is reconciled with MIB
MIB
Golden Store
Management System
Database
Golden Store
Synchronization: Network is reprovisioned from database
MIB

Management Agent 1/2

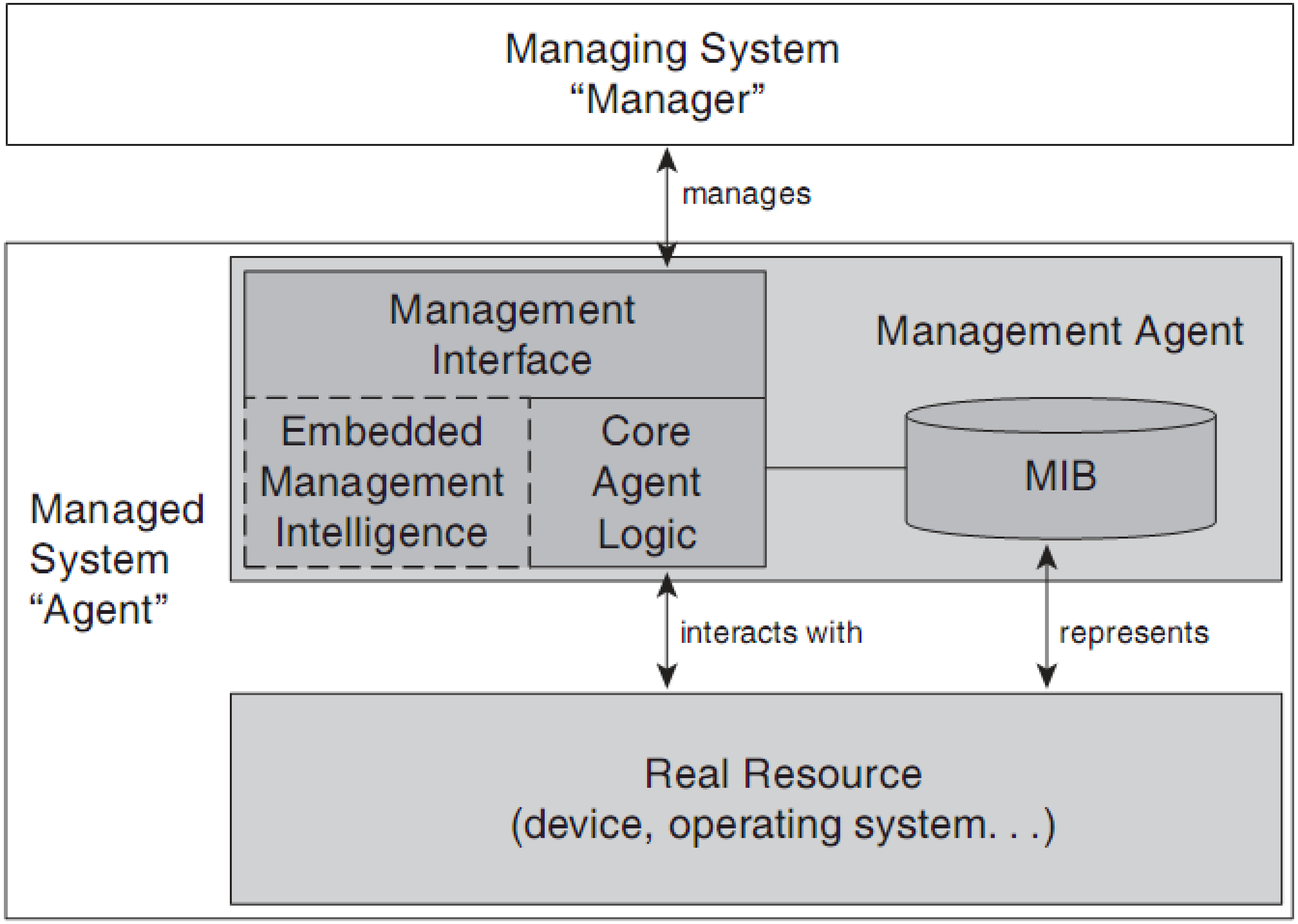
- Asymmetrical mgmt communication
- **Manager - agent**
  - Client - server
- Software (*manager agent*)
- Agent = agent role and sw
- Mgmt interfaces
  - To handle mgmt comm.
- Management Information Base (MIB) is a conceptual data store
  - ➔ mgmt information
- Core agent logic (mapping)
- Added mgmt functions (intelligence)
  - offload managers





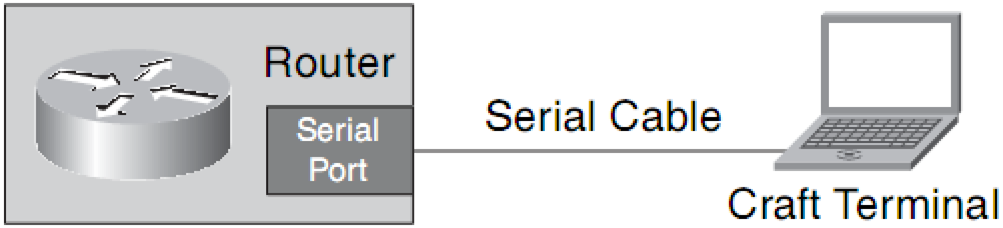


Management Agent 2/2



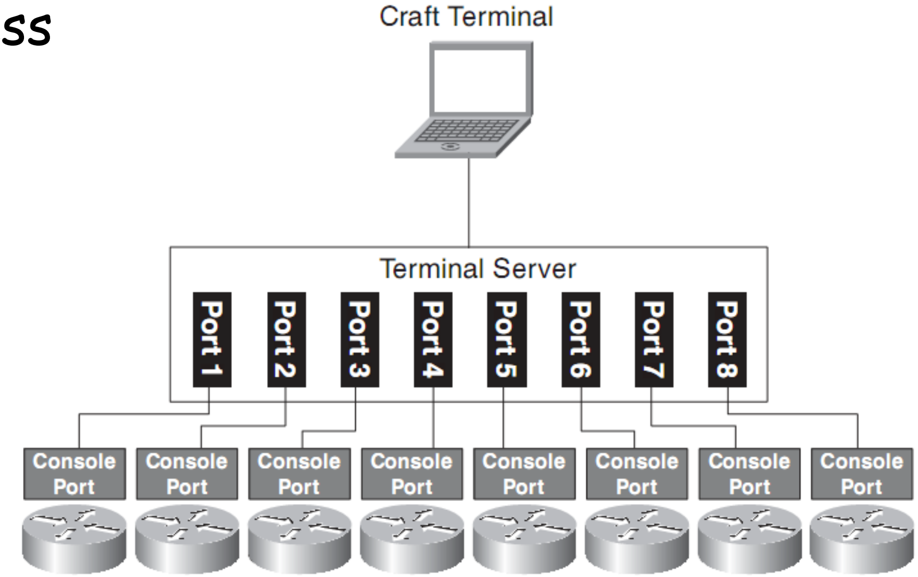
Networking for Management

- Serial interface / command line interface (CLI)
  - USB!
- Terminal server

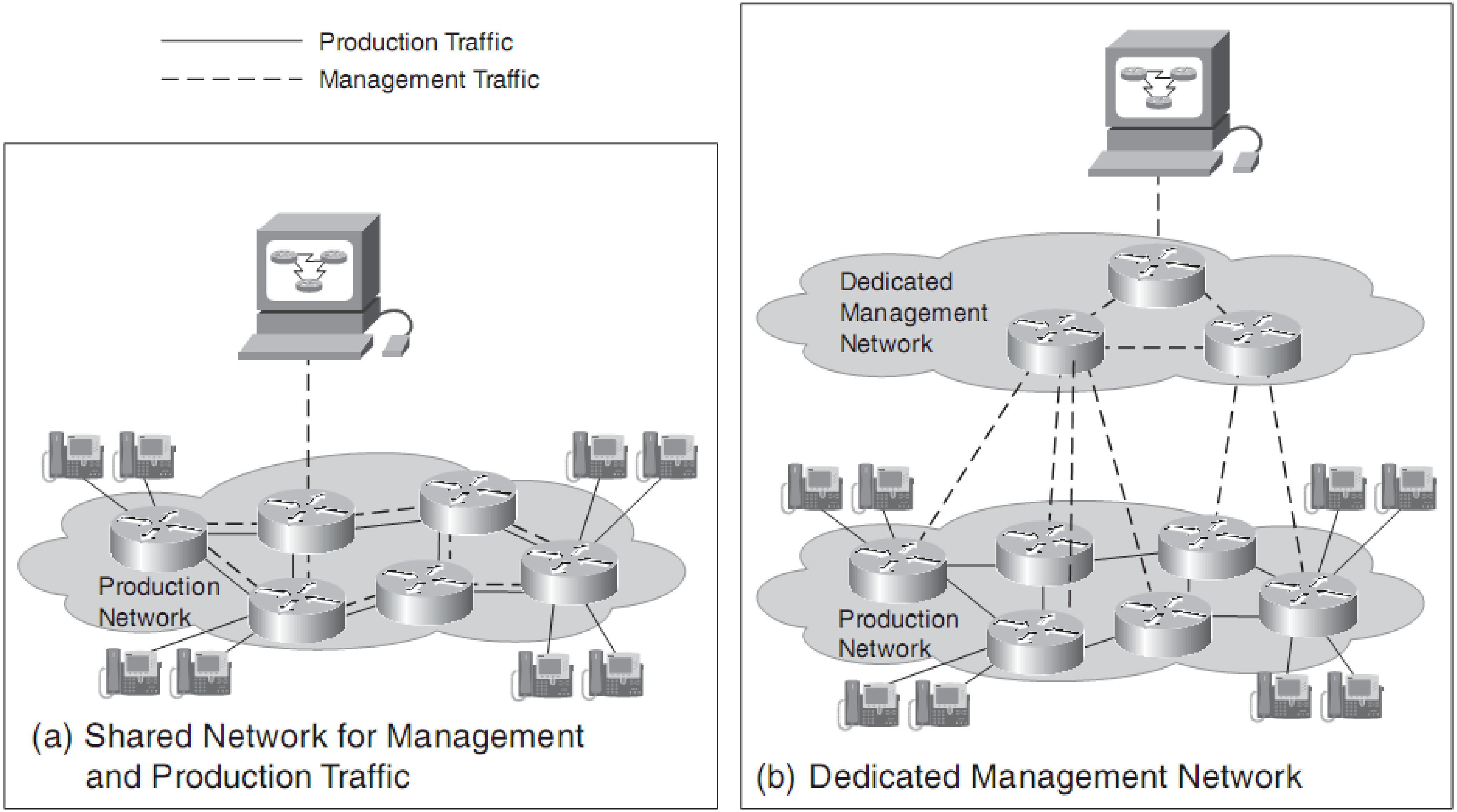


- Terminal server with IP address
- NE with Ethernet port
  - Dedicated
  - Shared with prod. Traffic

- "In band"
  vs.
  "out of band"



Dedicated Mgmt Network 1/2



(a) Shared Network for Management and Production Traffic

(b) Dedicated Management Network

Dedicated Mgmt Network 2/2

- ❑ Reliability
- ❑ Interference avoidance
- ❑ Ease of network planning
- ❑ Security
- ❑ Cost and overhead
- ❑ No reasonable alternative
  - ▪ E.g. DSL lines

➔ dedicated management network has undeniable advantages!

Outline

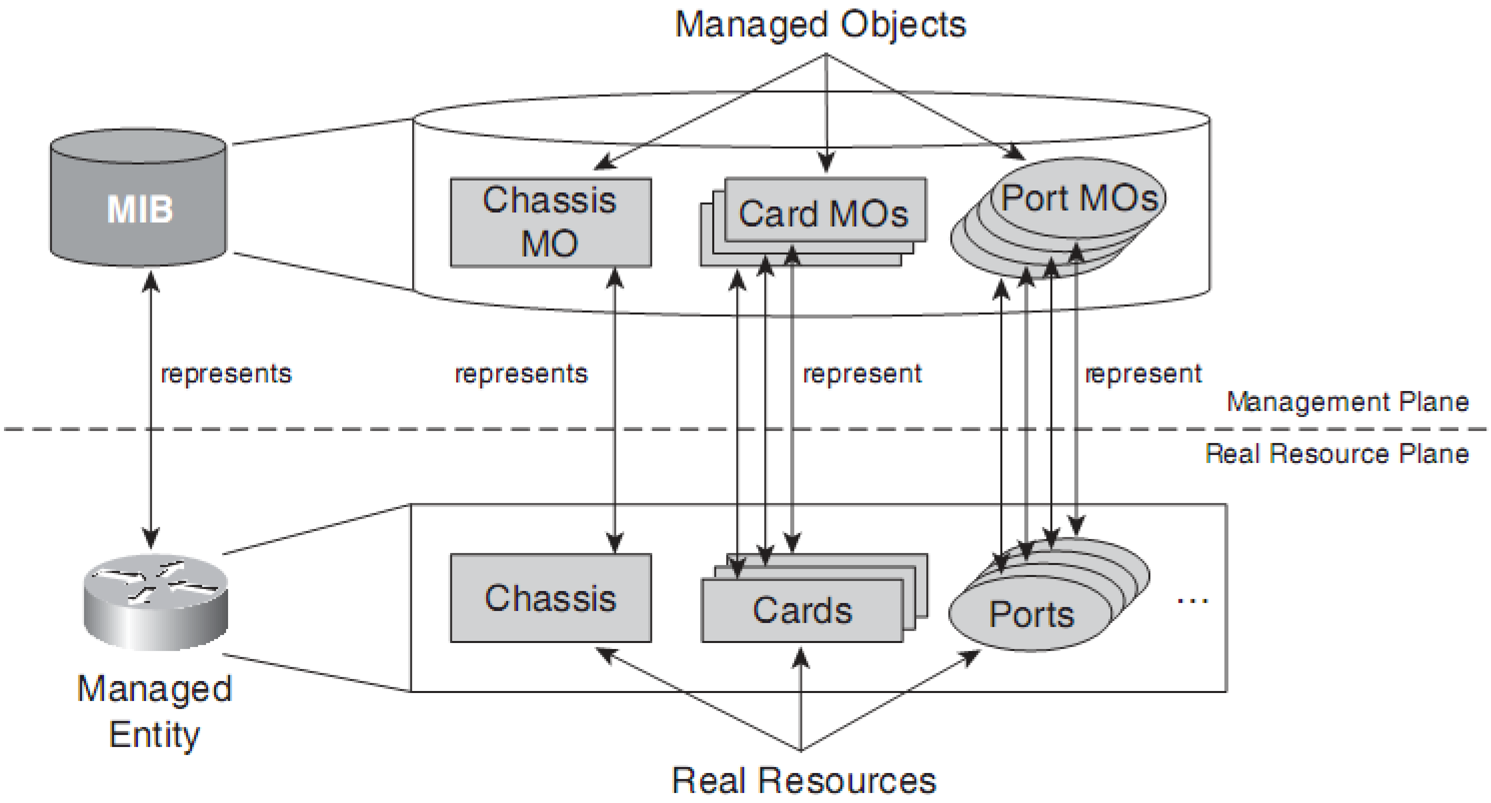
- ❑ SNMP
- ❑ Command Line Interface (CLI)
- ❑ syslog
- ❑ Netconf
- ❑ Netflow
- ❑ Summary

Introduction

- Everything that managers need to know about the entity that is being managed constitutes **management information**

- *A* central aspect of management information is that it establishes a common and mutually understood way by which agents and managers can refer to various aspects of the managed device.

MIB and MOs, Managed Entity and Real Resources



Categories of Management Information 1/2

- ❑ **State information**
  - Physical and logical resources
  - Operational data
  - **May change frequently and rapidly**
    - **Not to cache!!!**
  - Management applications **cannot modify** state information but can only **retrieve it**, i.e. state information is effectively “owned” by the device
- ❑ **Physical configuration information**
  - “owned” by the device
  - Manager cannot modify it
  - Information changes only rarely
  - Best **to cache** it at the managers

Categories of Management Information 2/2

- Logical configuration information
  - typically controlled and can be changed by management
  - **Cache it** (it changes only when *a manager* changes it)
  - Types: startup and transient
- Historical information
  - does not reflect actual managed resources
    - (it should not be in the MIB at all)

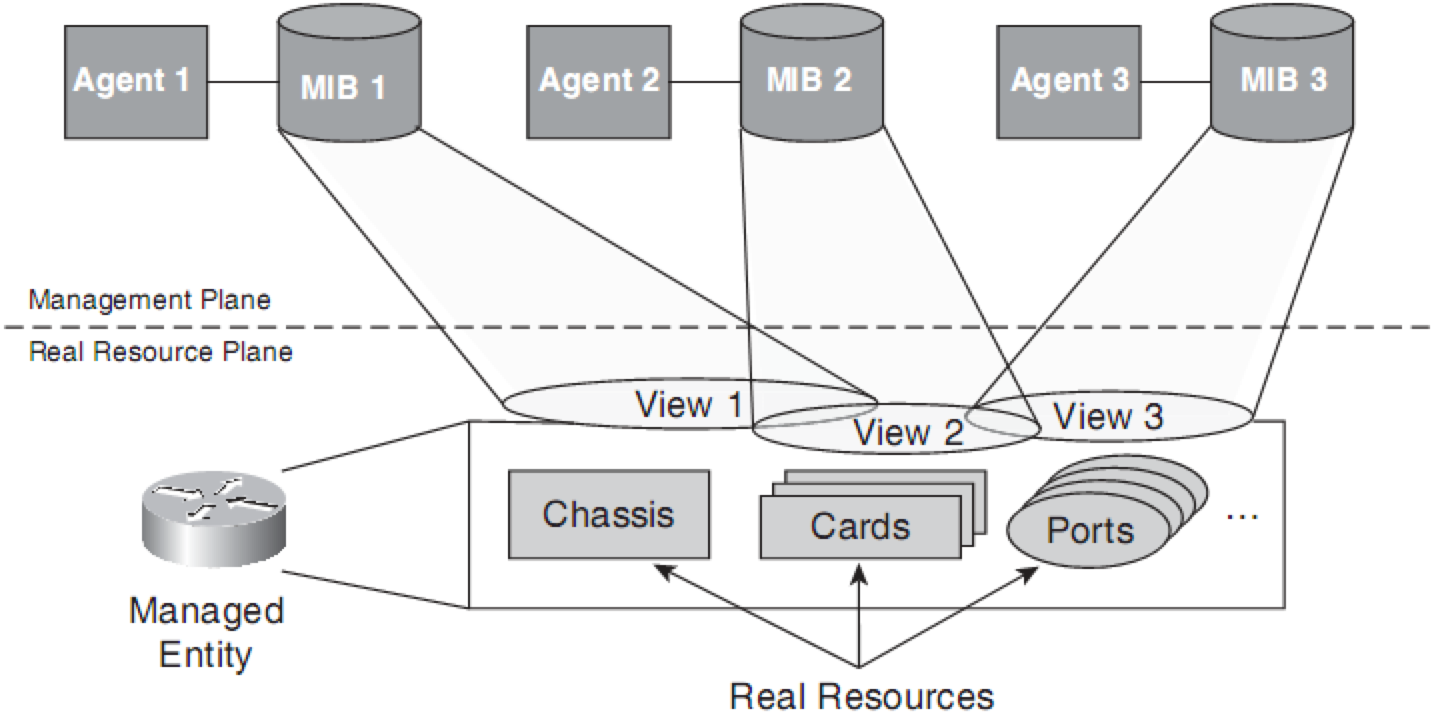
- **Sometimes actions to be performed on the device are also stored in MIBs.**

MIB vs Databases

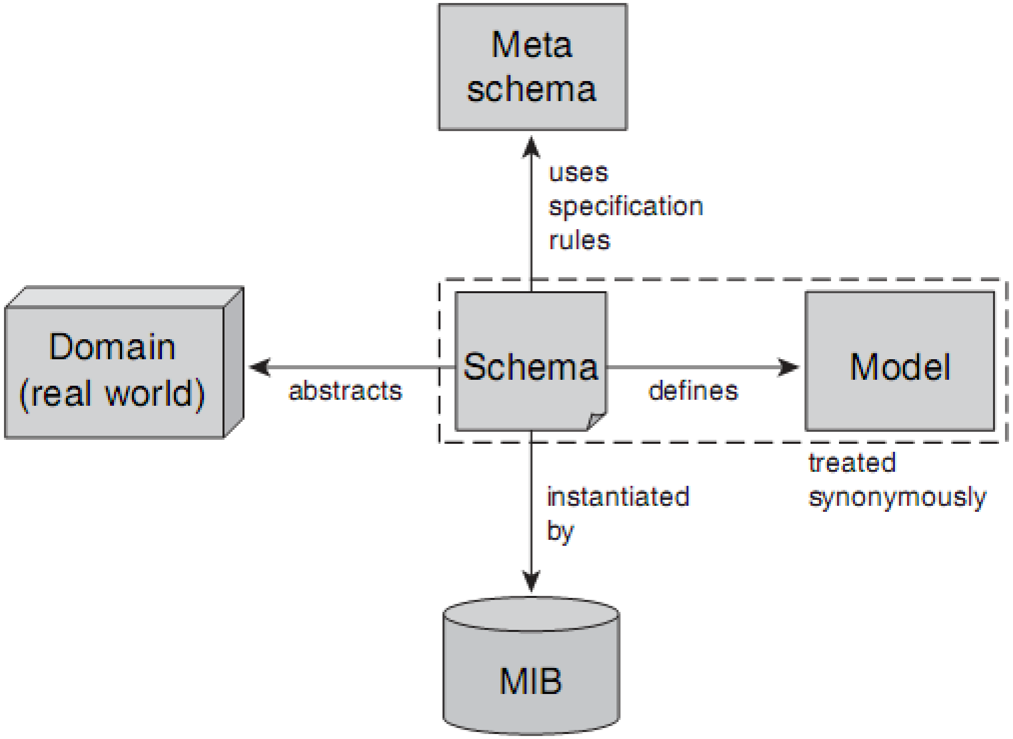
- Why not use SQL and database mgmt systems (DBMS)?
  - Footprint at agent
  - Specialties
    - Hierarchical information
    - Some information is maintained by the agent some by the manager
    - No need for joins between tables and filtering
  - Real effects
    - Mgmt information is accessed by control plane protocols, the device, users in CLI
  - Data characteristics
    - DBMS: few tables with many entries
    - MIB: many different types with relatively few instances of each
- Though, managers store data in DBMS'

MIB and Mgmt Protocols

- MIB does not depend on any particular mgmt protocol

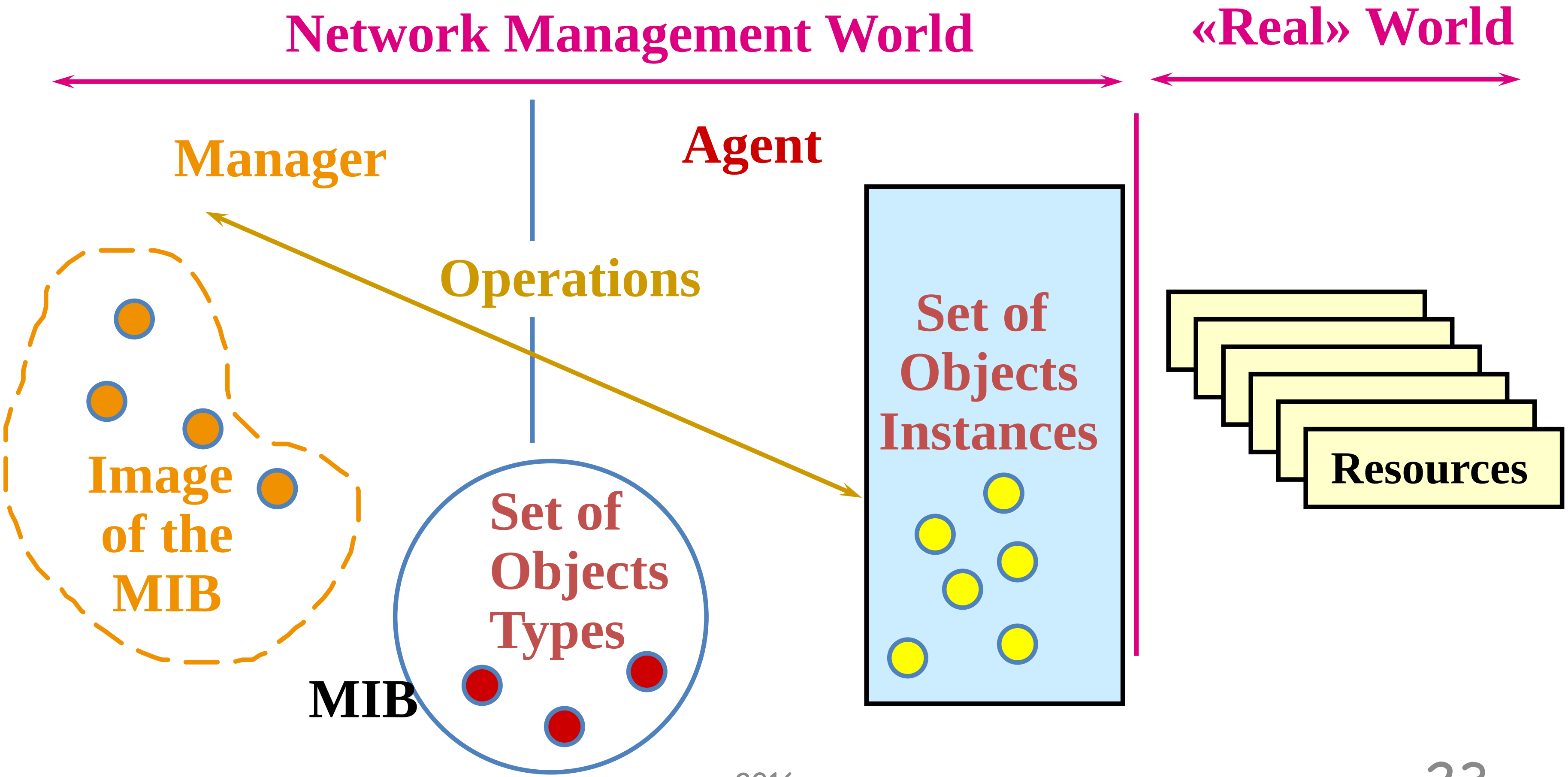


Schema, Metaschema, Model, Domain, and MIB



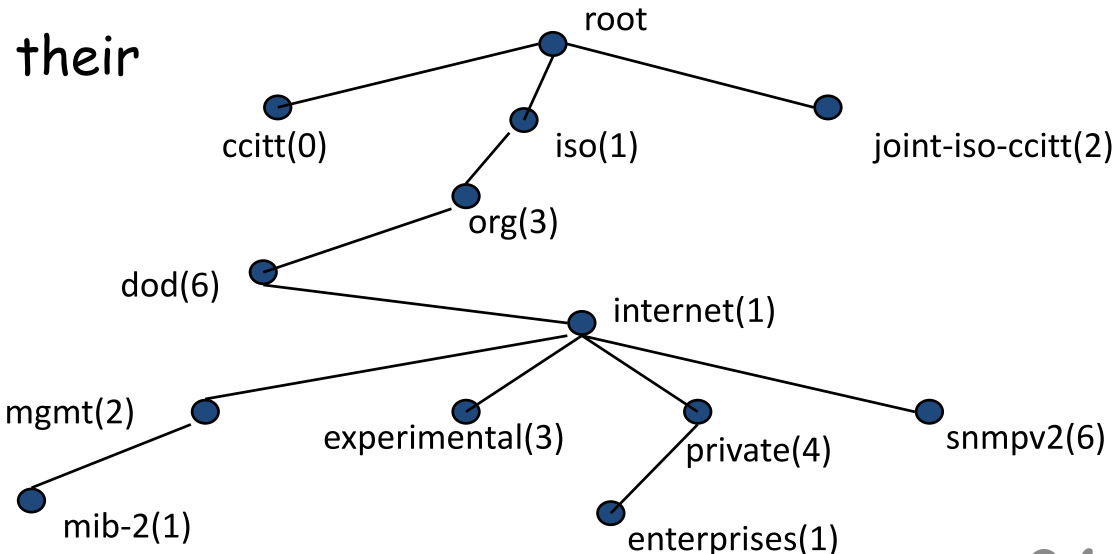
Resource, Managed Objects and MIB

How to model management information?

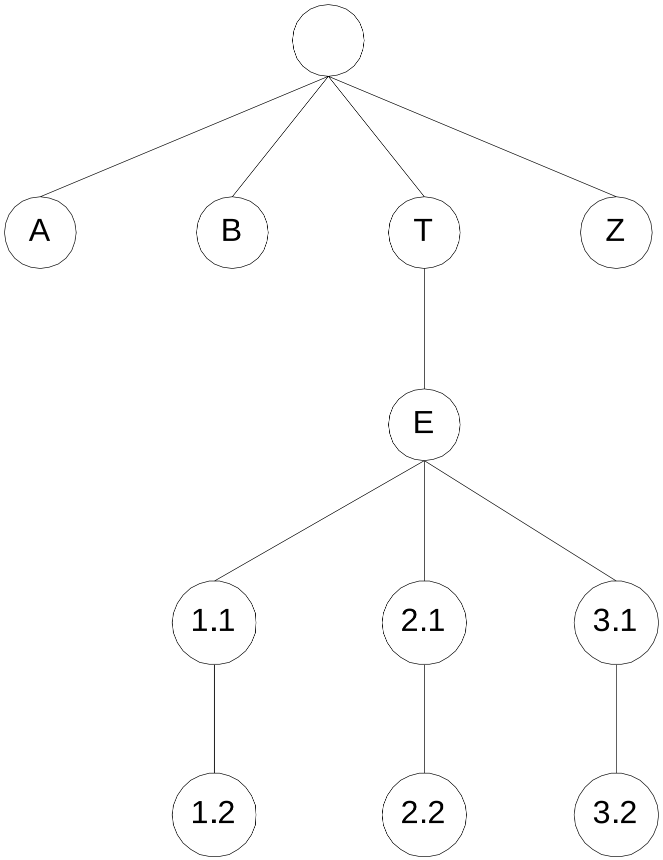


MIB as a Conceptual Tree

- Each node is **named relative to** a containing node
  - == Object Identifier (OID)
  - Top node of a MIB module is the def. of the MIB module
    - and is registered into the (global) Internet object identifier tree
- Enterprises node
  - Companies to add their own proprietary MIB modules
  - No need to ask for permissions



Lexicographic Ordering



A
B
T
E
1.1
1.2
2.1
2.2
3.1
3.2
Z

Outline

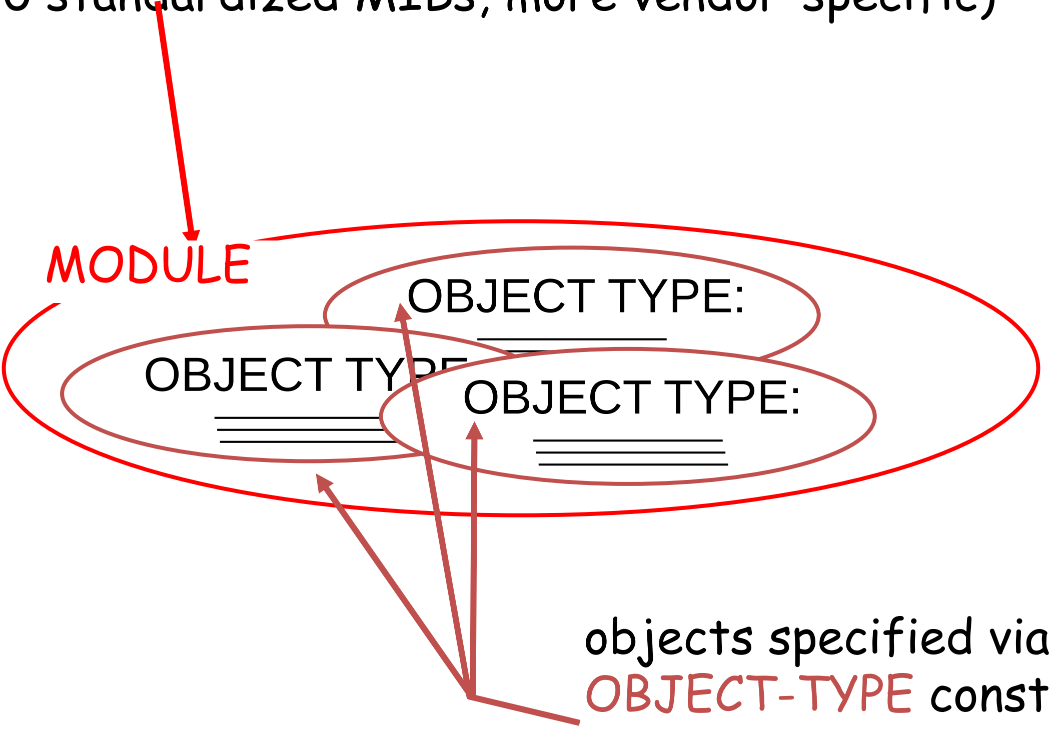
- ❑SNMP
- ❑Command Line Interface (CLI)
- ❑syslog
- ❑Netconf
- ❑Netflow
- ❑Summary

SNMP komponensek

- Management Information Base (MIB):
    - hálózatmenedzsment információk elosztott tárolása
- Structure of Management Information (SMI):
    - adatdefiníciós nyelv a MIB-ek számára
- SNMP protokoll
    - információ szállítására a menedzser és ügynök között
- Biztonság és adminisztráció
    - főleg SNMPv3-ban

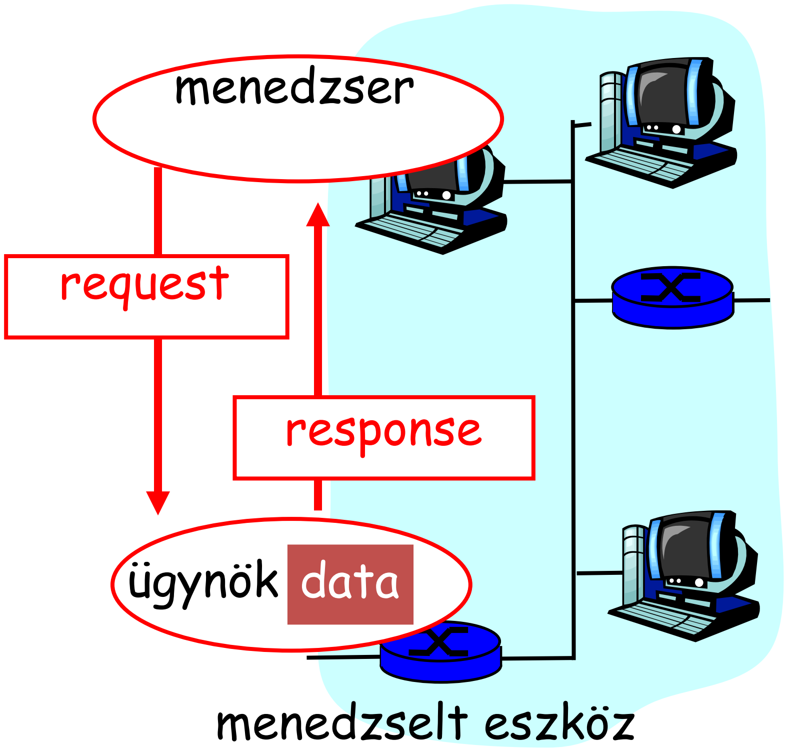
SNMP MIB

MIB module specified via SMI
MODULE-IDENTITY
(100 standardized MIBs, more vendor-specific)



objects specified via SMI
OBJECT-TYPE construct

SNMP: kommunikációs módok (1/2)

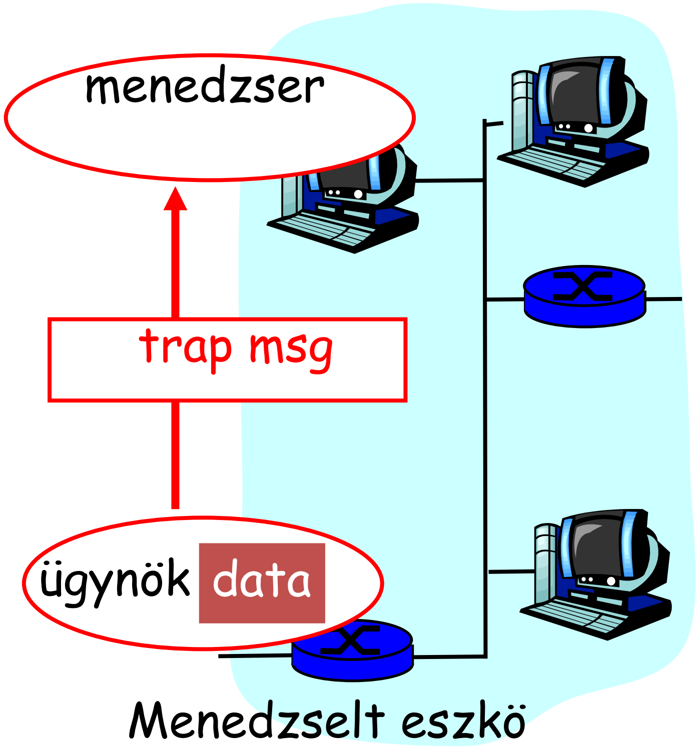


request/response mode

- ❑ Menedzser kérést intéz az ügynökhöz
- ❑ Az ügynök fogadja a kérést, végrehajtja majd válaszol
- ❑ Felhasználás
  - MIB objektumok lekérdezésére
  - MIB objektumok megváltoztatására
- ❑ **Client pull**
  - menedzsment *kliens* (menedzser) kihúzza a adatot a *szerverből* (ügynök)

SNMP: kommunikációs módok (2/2)

- ❑ az ügynök egy „felszólítás nélküli" (unsolicited) üzenetet küld a menedzsernek – trap üzenet
- ❑ Felhasználás
  - jelentés rendkívüli szituációról amely valamely MIB objektum értékének változásaként állt be
- ❑ **Server push**
  - a szerver (ügynök) kilöki a trap üzenetet a kliens (menedzser) felé



trap mode

Client pull és server push módok (1/2)

- ❑Overhead
  - pull módnak két üzenetre van szüksége míg a push módnak csak egyre
  - push módnál csak kivételes helyzetek vannak jelentve, remélhetőleg kevesebb üzenet.
    - Pull módnál akár sok-sok üzenetváltás, hogy kiderüljön minden rendben
  - ➔ Mindkét esetben a push mód kevesebb hálózati erőforrást használ ami előnyös
    - „ha a hálózatmenedzsment elárasztja a hálózatot akkor minek menedzselni?"

Client pull és server push módok (2/2)

- ❑ *Push* módnál az abnormális helyzet azonnal jelezésre kerül (időtényező)
  - ▪ pull mód megvárja a következő lekérdezési időt, késleltetett probléma-felderítés
  - ➔ ezen idő alatt a hiba továbbgyűrűzhet

Robosztusság

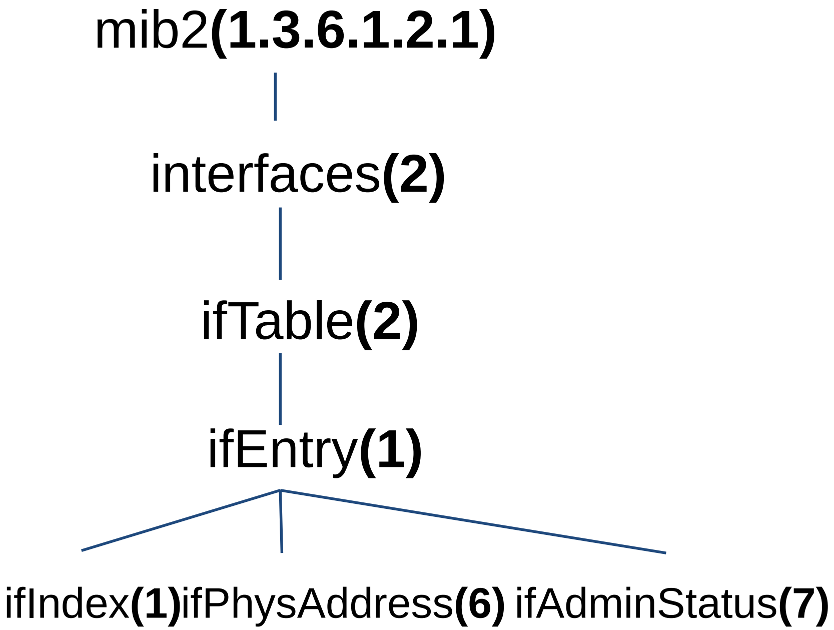
- ❑ A *pull* lekérdezésből kinyert többlet információ komplex analízist támogathat
- ❑ Ha az üzenetek elvesznek az ügynök és a menedzser között a *pull* mód ezt észreveszi
  - ▪ nincs válasz a kérésre
  - ▪ *push* módszer képtelen ezt kezelni!

SNMP üzenet típusok

Üzenet típus	Funkció
GetRequest GetNextRequest GetBulkRequest	Menedzser-ügynök “kérek adatot”
InformRequest	Menedzser-menedzser: MIB érték
SetRequest	Menedzser-ügynök: érték állítás
Response	Ügynök-menedzser: érték, válasz a kérésre
Trap	Ügynök-menedzser: értesítés kivételes helyzetről

Get Request működés

- csak a levél objektumokat érheti el!



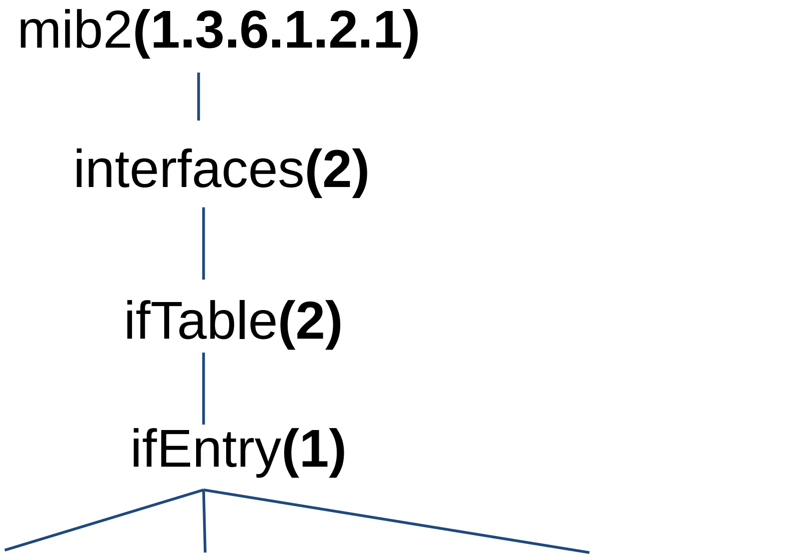
1	**00:00:39:20:04**	**1 (up)**
2	**08:00:56:16:11**	**3 (testing)**
8	**00:00:b4:02:33**	**2 (down)**

GetRequest (ifPhysAddress.2)

↓

Response (ifPhysAddress.2 = 08:00:56:16:11)

Get Request táblázatos objektumok esetén

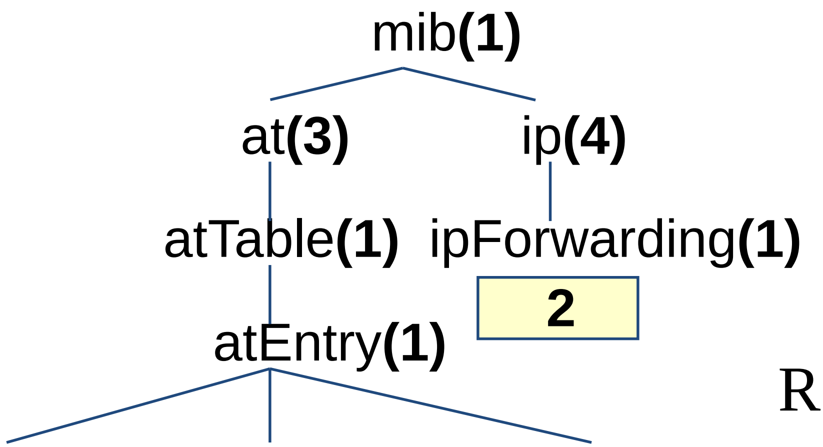


ifIndex(1)	ifPhysAddress(6)	ifAdminStatus(7)
1	00:00:39:20:04	1 (up)
2	08:00:56:16:11	3 (testing)
8	00:00:b4:02:33	2 (down)

- ❑ mivel csak a levél objektumokat érheti el, így nem lehet egész sort/oszlopot lekérdezni:
  - sor lekérdezése: minden egyes oszlop objektumra referálni

GetRequest (ifIndex.2,
ifPhysAddress.2,
ifAdminStatus.2)

Ismeretlen táblázat lekérdezése 1/4

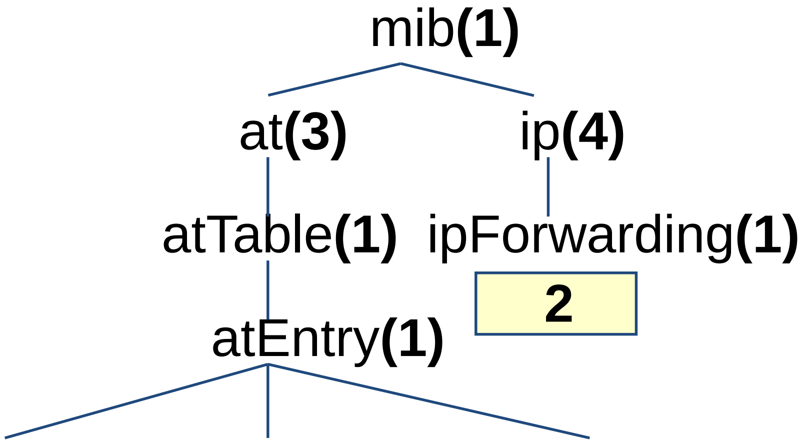


atIfIndex	atPhysAddr.	atNetAddr.
1	00:00:39:20:04	194.2.6.10
4	08:00:56:16:11	194.22.67.45
5	00:00:b4:02:33	194.7.53.11

GetNextRequest (
atIfIndex, atPhys, atNet)

Response (atIfIndex.1 = 1,
atPhys.1 = 00:00:39:20:04,
atNet.1 = 194.2.6.10)

Ismeretlen táblázat lekérdezése 2/4

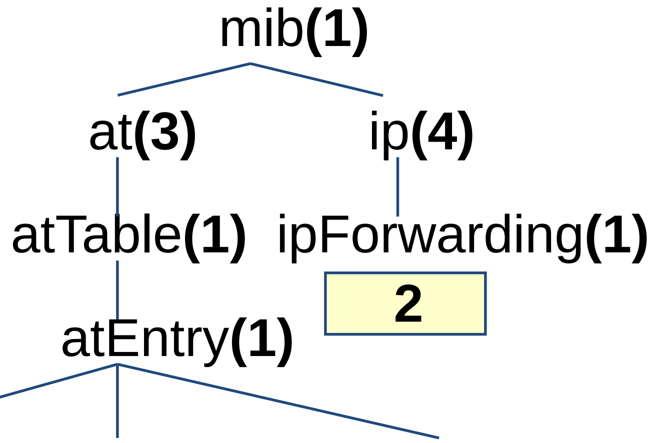


atIfIndex	atPhysAddr.	atNetAddr.
1	00:00:39:20:04	194.2.6.10
4	08:00:56:16:11	194.22.67.45
5	00:00:b4:02:33	194.7.53.11

GetNextRequest
(atIfIndex.1, atPhys.1,
atNet.1)

Response (atIfIndex.4 = 4,
atPhys.4 = 08:00:56:16:11,
atNet.4 = 194.22.67.45)

Ismeretlen táblázat lekérdezése 3/4

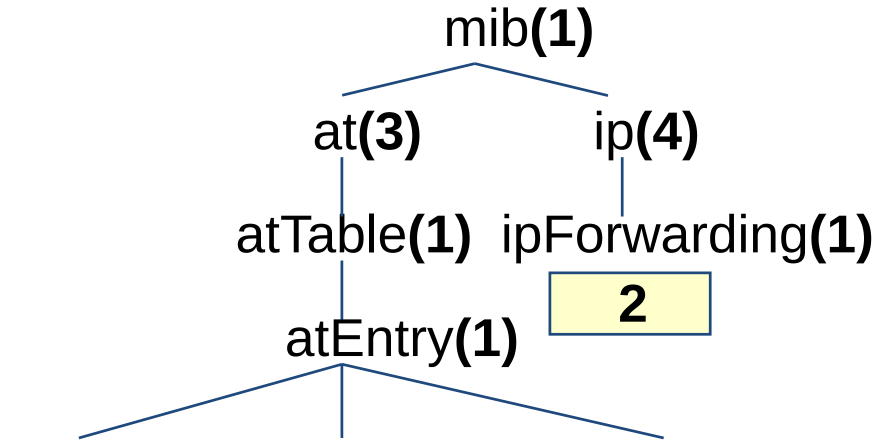


atIfIndex	atPhysAddr.	atNetAddr.
1	**00:00:39:20:04**	**194.2.6.10**
4	**08:00:56:16:11**	**194.22.67.45**
5	**00:00:b4:02:33**	**194.7.53.11**

GetNextRequest
(atIfIndex.4, atPhys.4,
atNet.4)

Response (atIfIndex.5 = 5,
atPhys.5 = 00:00:b4:02:33,
atNet.5 = 194.7.53.11)

Ismeretlen táblázat lekérdezése 4/4



atIfIndex	atPhysAddr.	atNetAddr.
1	**00:00:39:20:04**	**194.2.6.10**
4	**08:00:56:16:11**	**194.22.67.45**
5	**00:00:b4:02:33**	**194.7.53.11**

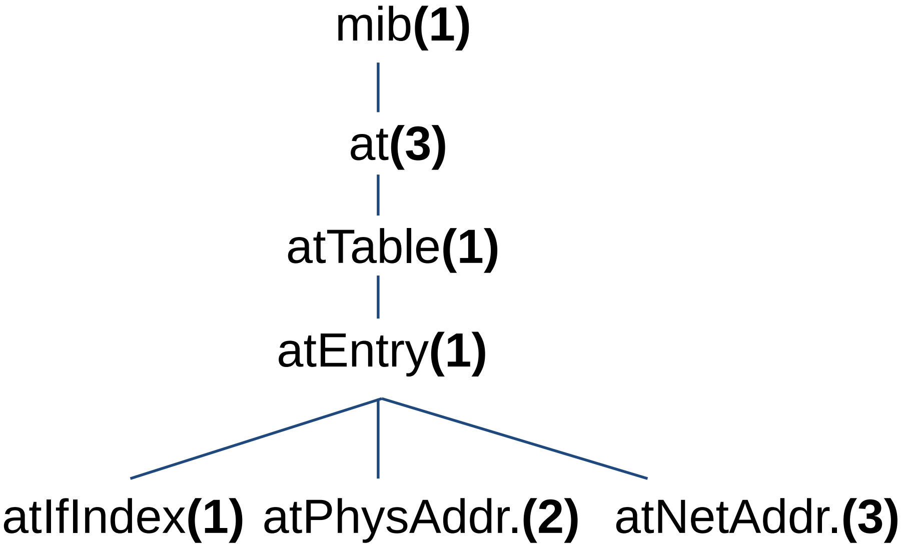
- ❑ a kérés és válaszban lévő objektum nevek nem egyeznek meg
  - ▪ a menedzser a táblázat végére ért

GetNextRequest (atIfIndex.5, atPhys.5, atNet.5)

Response (
atPhys.1 = 00:00:39:20:04,
atNet.1 = 194.2.6.10,
ipForwarding.0 = 2)

Set Request működés

- ❑ csak a levél objektumokhoz

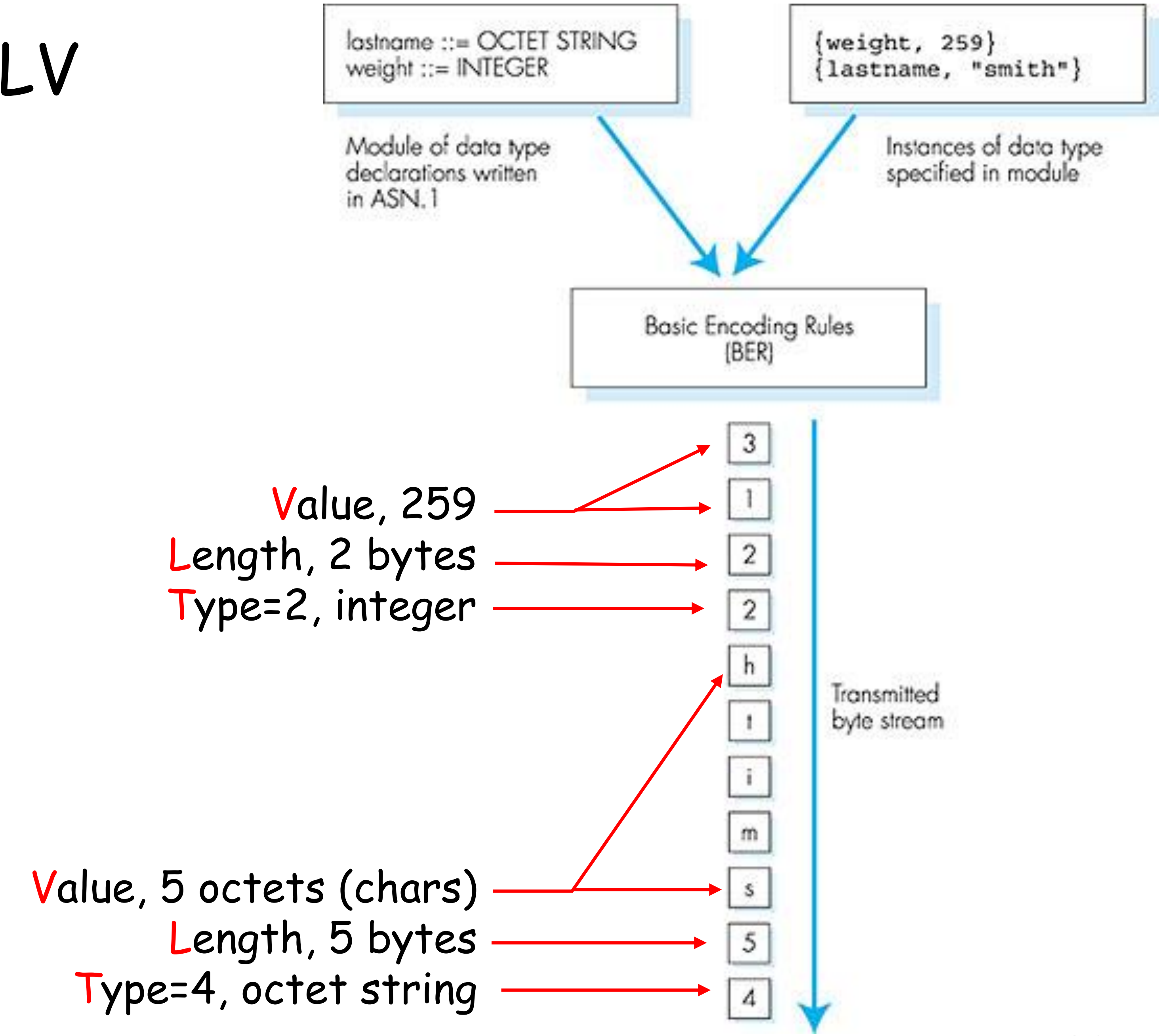


1	**00:00:39:20:04**	**194.2.6.10**
4	**00:00:77:b1:45**	**194.22.67.45**
5	**00:00:b4:02:33**	**194.7.53.11**

SetRequest (
atPhysAddress.4 = 00:00:77:b1:45)

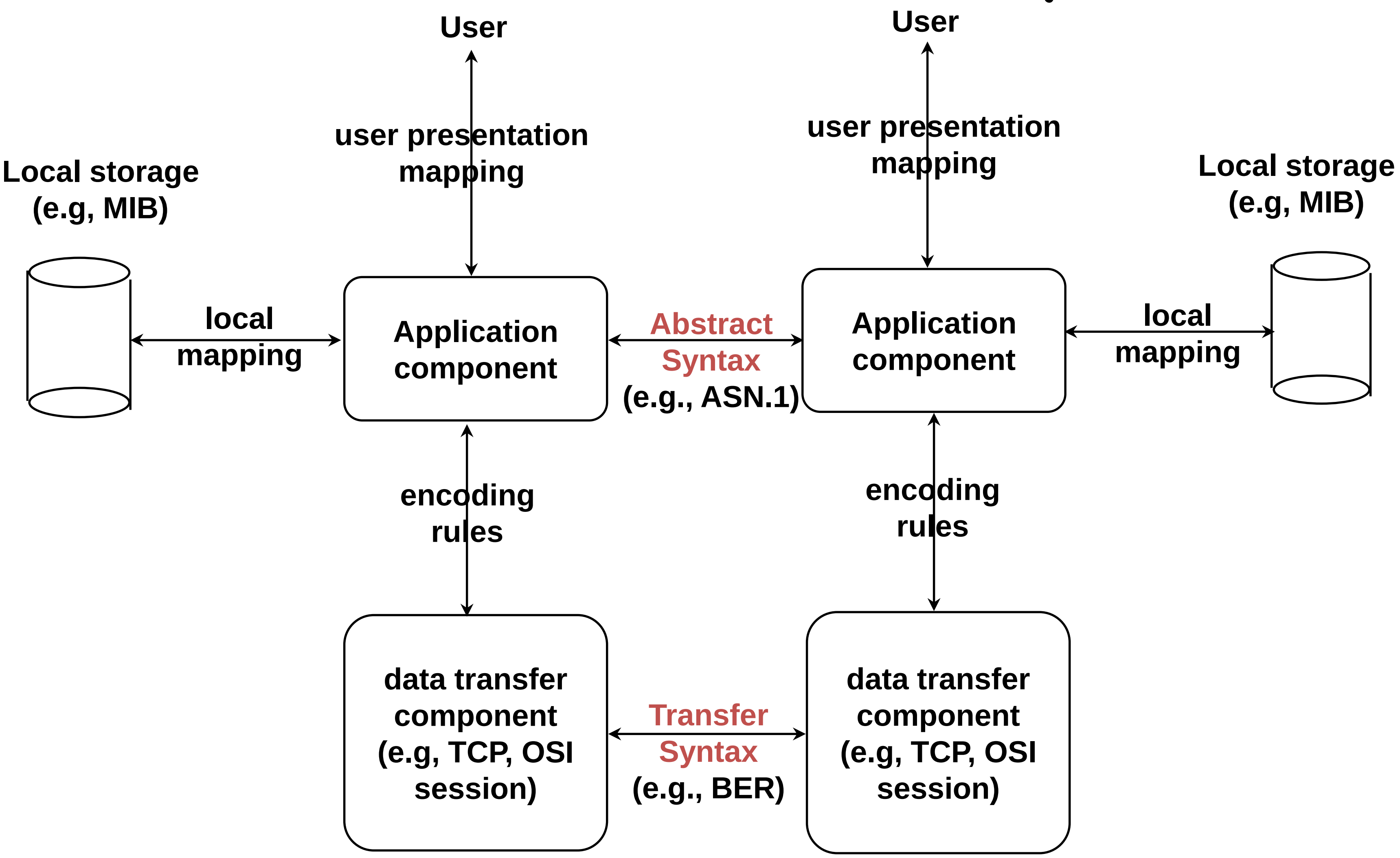
Response (
atPhysAddress.4 = 00:00:77:b1:45)

ASN+TLV kódolás


lastname ::= OCTET STRING
weight ::= INTEGER
{weight, 259}
{lastname, "smith"}
Module of data type declarations written in ASN.1
Instances of data type specified in module
Basic Encoding Rules (BER)
3
1
2
2
h
t
i
m
s
5
4
Transmitted byte stream

Value, 259
Length, 2 bytes
Type=2, integer
Value, 5 octets (chars)
Length, 5 bytes
Type=4, octet string

Abstract & Transfer Syntaxes



SNMPv2 & v3

- V2: Funkcionális kiterjesztés
- V3: teljes újratervezésé az architektúrának, biztonsági kiterjesztések

Outline

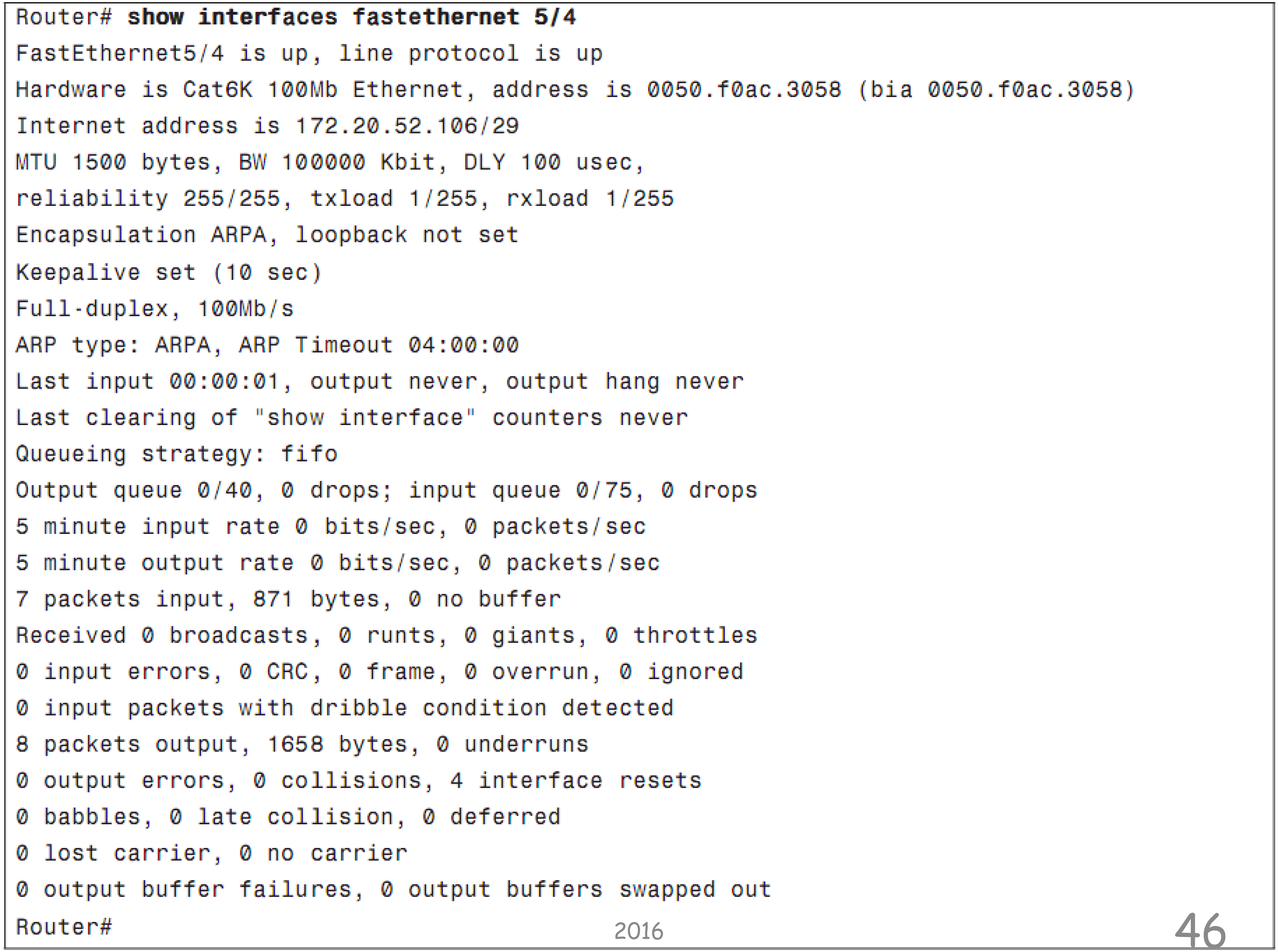
- SNMP
- Command Line Interface (CLI)
- syslog
- Netconf
- Netflow
- Summary

Command Line Interface (CLI)

- ❑ First routers were just UNIX machines
- ❑ CLIs
  - ▪ On Juniper: JunOS
  - ▪ On Cicso: Cisco's Internet Operating System (IOS)

- ❑ CLIs are for human interactions
  - ▪ With help functions (?)
  - ▪ Auto completion
  - ▪ Prompts for different modes

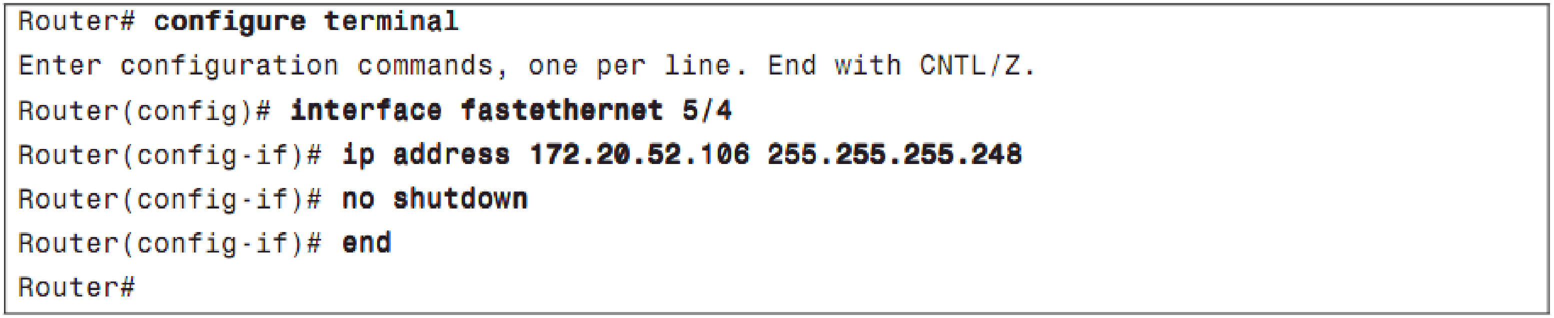
- ❑ Modes and submodes is an interesting property of CLI
  - ▪ It offers security levels
  - ▪ Less typing



```
Router# show interfaces fastethernet 5/4
FastEthernet5/4 is up, line protocol is up
Hardware is Cat6K 100Mb Ethernet, address is 0050.f0ac.3058 (bia 0050.f0ac.3058)
Internet address is 172.20.52.106/29
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Full-duplex, 100Mb/s
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:01, output never, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
7 packets input, 871 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 input packets with dribble condition detected
8 packets output, 1658 bytes, 0 underruns
0 output errors, 0 collisions, 4 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
Router#
```

Observations

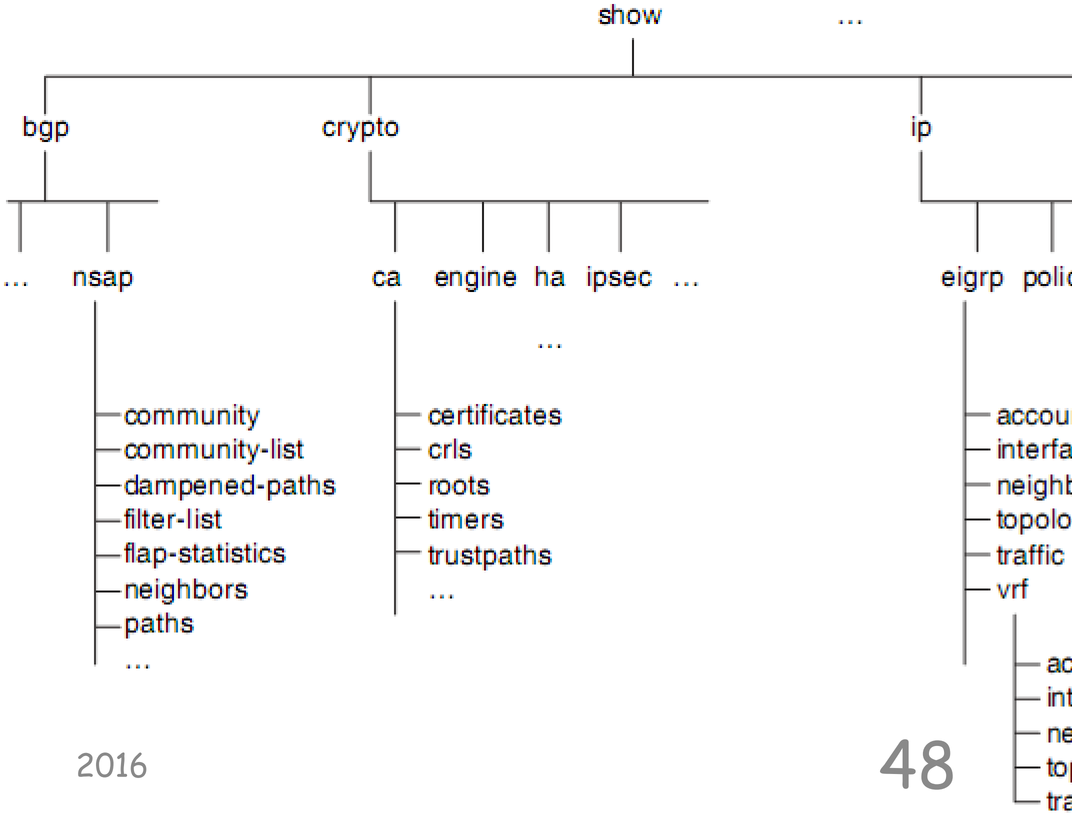
- Content of lines are different
  - A single MIB variable
  - Several MIB variables
- Different delimiters are used
- ➔ **CLI is relatively difficult to use for scripts and mgmt applications**
  - "Screen scarping"



```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface fastethernet 5/4
Router(config-if)# ip address 172.20.52.106 255.255.255.248
Router(config-if)# no shutdown
Router(config-if)# end
Router#
```

CLI commands

- Organized in hierarchies
  - E.g.: show ip policy-list, show ip ospf, show ip rip
  - Makes it human friendly with autocompletion



CLI as Management Protocol

- The concern is **to interpret the results** of CLI commands
  - No clear return code
  - Config and show statements are not symmetric
    - Unlike with Get and Set
  - Scrape the screen
  - E.g. show commands outputs are best suited for humans
    - Tables or multi line output with special structure
      - In tables entries are not even tagged (they belong to a column)
- **Custom code** to support each individual outputs
- Since CLIs are command, they only allow **request and response communication patterns**
  - So, besides the command line, one needs additional mechanism to handle events

Outline

- ❑(SNMP)
- ❑Command Line Interface (CLI)
- ❑syslog
- ❑Netconf
- ❑Netflow
- ❑Summary

syslog

- (written in lowercase)
- From UNIX systems
- Mechanism for managed devices **to emit event messages**

Overview 1/2

- System messages written into a file for further analysis
- Each syslog message is one entry in the log file
  - N.B.: Management applications can receive the entry without retrieving the log files
- syslog messages may contain
  - From critical alarm conditions to
  - debug statements everything
    - "I think I may have just dropped the tenth packet in a row,"
    - "I'm experiencing good utilization on my link,"
    - "Look, I'm currently in this new branch of code,"
    - "Strange— someone just tried to log into me a hundred different times, trying a different password each time."
- ➔ **general trail of the activity of the device**

Overview 2/2

- ❑Messages for humans
  - ▪ Like CLI
- ❑Mostly from print statements in the code
- ❑**The weakness is in how to parse the syslog**

syslog Messages

- Message header and body
- Body
  - Informal part
    - Plain English text
- Header
  - Minimal information in a very structured way
    - Time
    - Name of the host
    - Severity of the message
    - Subsystem
    - mnemonic (name for the type of message)
  - **least common denominator of information that should be present in every event message**

syslog examples

- 172.19.209.130 000024: *Apr 12 18:01:55.643: % ENV_MON-1-SHUTDOWN: Environmental Monitor initiated shutdown

- 01:14:11: %IPPHONE-6-REG_ALARM: 25: Name=SEP003094C38724 Load=3.2(2.9) Last=Initialized

  - Note no IP address is included
    - From file
    - Over TCP or UDP transport protocol

Outline

- ❑(SNMP)
- ❑Command Line Interface (CLI)
- ❑Syslog
- ❑Netconf
- ❑Netflow
- ❑Summary

Netconf Design Goals

- ❑ Robustness
- ❑ Task oriented view
- ❑ Extensibility
- ❑ Standardized error handling
- ❑ Distinction between configuration data and operation state
- ❑ Operation on selected subset of mgmt data
- ❑ Data modeling language /human friendly/
- ❑ Secure transport, auth, and robust access control

Netconf

- R. Enns, Ed., NETCONF Configuration Protocol, RFC 4741, December 2006
- Newer management protocol
- Targeted at managing configuration of data-networking devices
  - Not on monitoring!
- "The Network Configuration Protocol (NETCONF) defined in this document provides mechanisms to **install**, **manipulate**, and **delete** the configuration of network devices. It uses an **Extensible Markup Language (XML)**-based data encoding for the configuration data as well as the protocol messages. The NETCONF protocol operations are realized on top of a simple Remote Procedure Call (RPC) layer."

Netconf position

- ❑Fills what SNMP left out in configuration mgmt area
- ❑More structured than CLI (for humans)

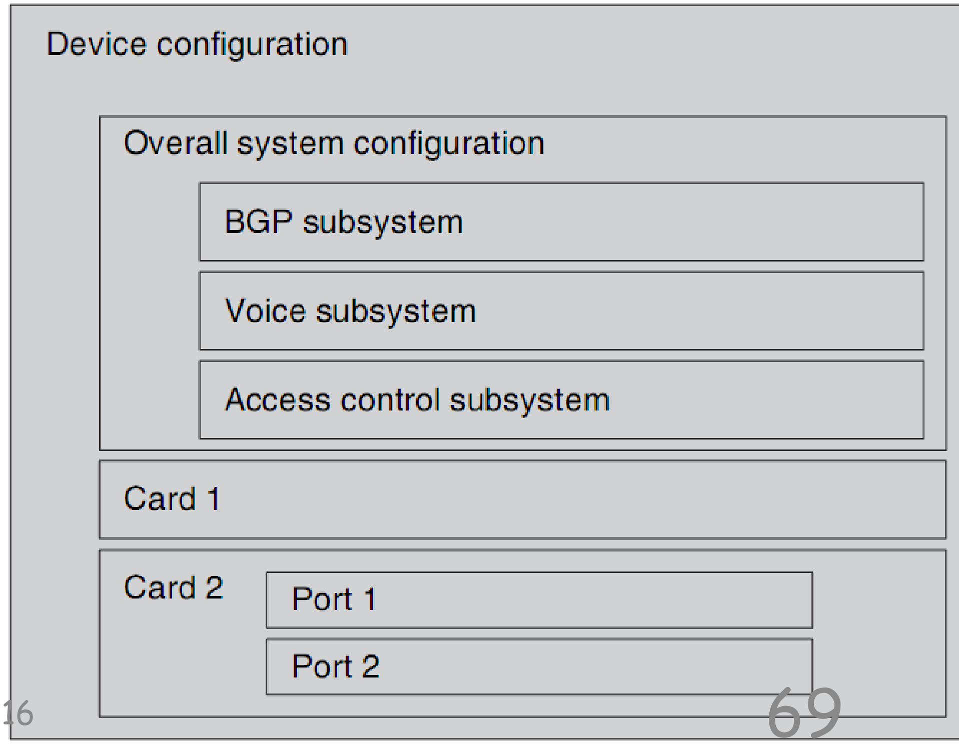
Netconf Datastores

- Configuration information of a device in a datastore
  - Like a file!
  - Resembles a MIB!
- Netconf provides operations to manage **datastores**

- SNMP
  - Operations on individual MO inside the MIB
- Netconf
  - Operation on the MIB or portions as a whole

Hierarchical Datastore

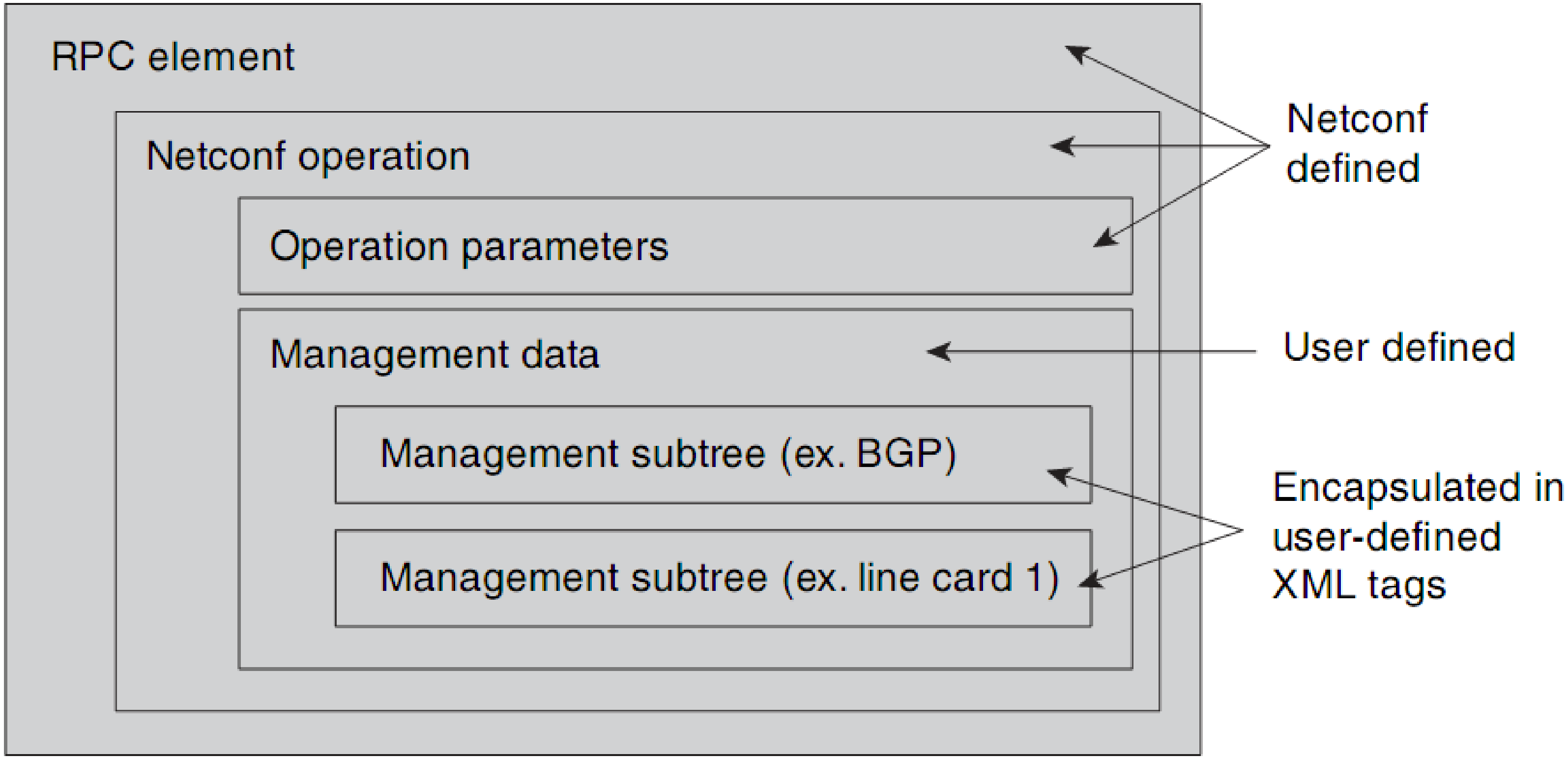
- ❑ Management data in the datastore in hierarchical treelike fashion ➔ **scopes**
- ❑ Management information can be **grouped**
  - **Container within a container**
  - Physical and logical **subsystems**
- ❑ Mgmt operations
  - Individual subtrees (subconfigurations)
  - Aka **Subtree filtering**



Netconf & XML

- No MIB specification language
- Netconf only provides **facilities to navigate** a datastore, where wrappers are using XML structure
  - Inside the wrappers any model the device supports
- XML for encoding management operations
- XML
  - Tags, e.g.: <email>me@foo.com</email>
  - The tags themselves and the semantics associated with them are not part of XML
  - XML Schema Definition (XSD)
- In addition to the XML operations, netconf assumes that inside the datastore configuration information is itself encoded in XML, i.e., they contain tags

Netconf Message Structure



RPC element
Netconf operation
Operation parameters
Management data
Management subtree (ex. BGP)
Management subtree (ex. line card 1)
Netconf defined
User defined
Encapsulated in user-defined XML tags

Netconf Operations

- Configuration information as a conceptual datastore
  - Configuration file, config
- Datastores
  - Running, startup, backup
  - Configuration for a particular service
  - Candidate config

YANG – Netconf Data Modeling Language

- ❑Hierarchical data models
- ❑Distinction configuration vs. state data
- ❑Modeling for event notifications
- ❑Augmentation (vendor)
- ❑Compact & optimized for human (XML-based)

RFC6020

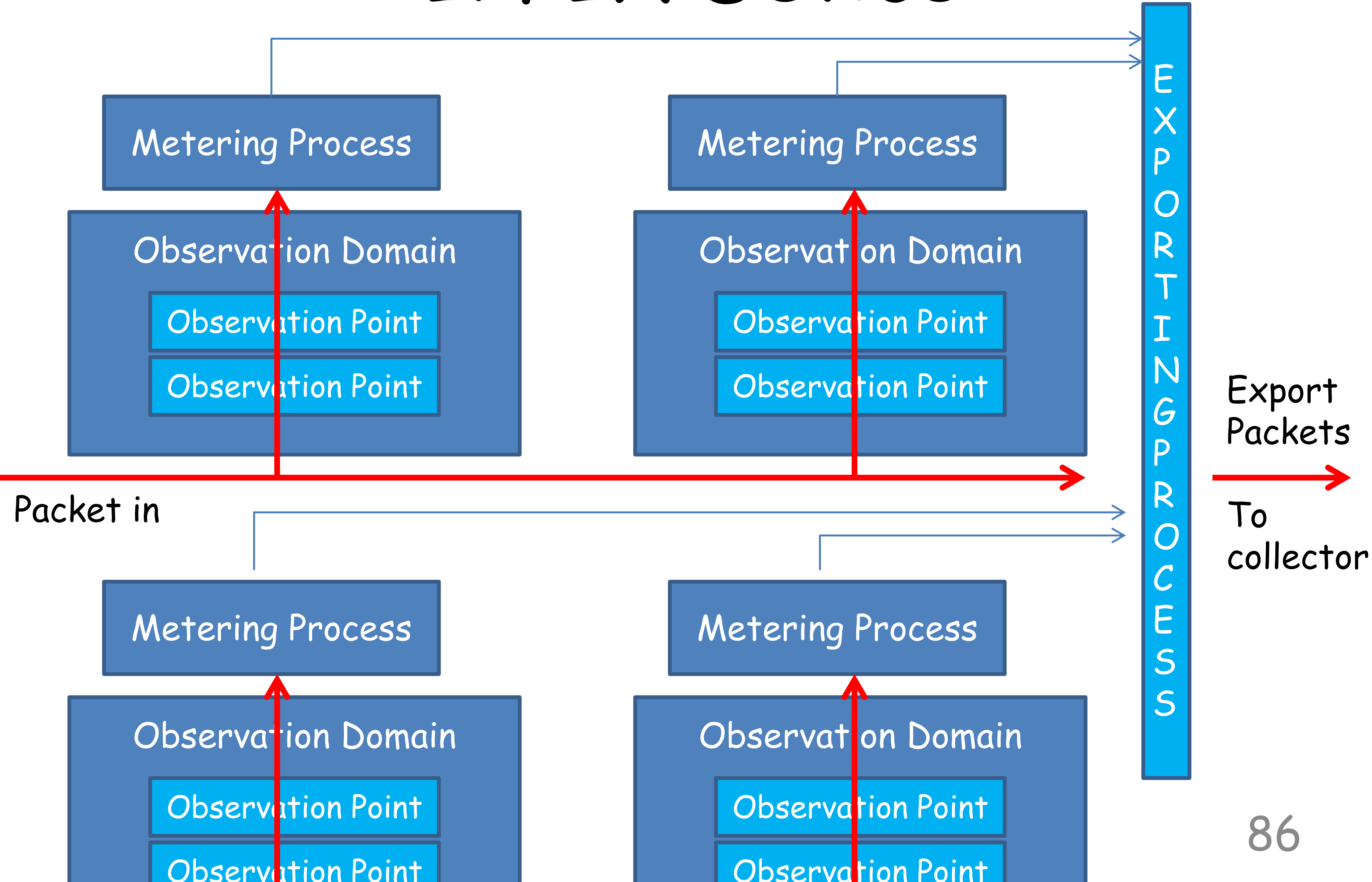
Outline

- ❑(SNMP)
- ❑Command Line Interface (CLI)
- ❑Syslog
- ❑Netconf
- ❑Netflow
- ❑Summary

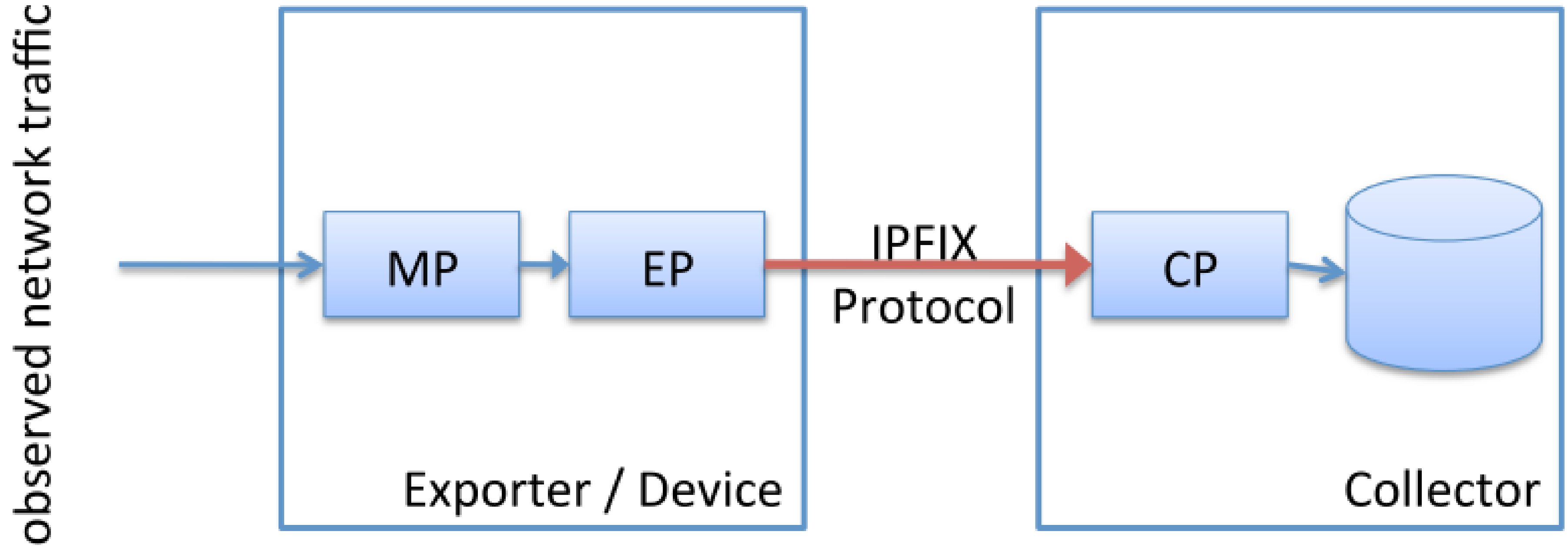
IP Flow Information eXport (IPFIX) / netflow

- ❑**IPFIX and Packet SAMPling (PSAMP) Protocols**
- ❑**push-based** data export mechanism for transferring **IP flow information** in a compact **binary format** from an **Exporter to a Collector**

IPFIX Device



Simple Setup

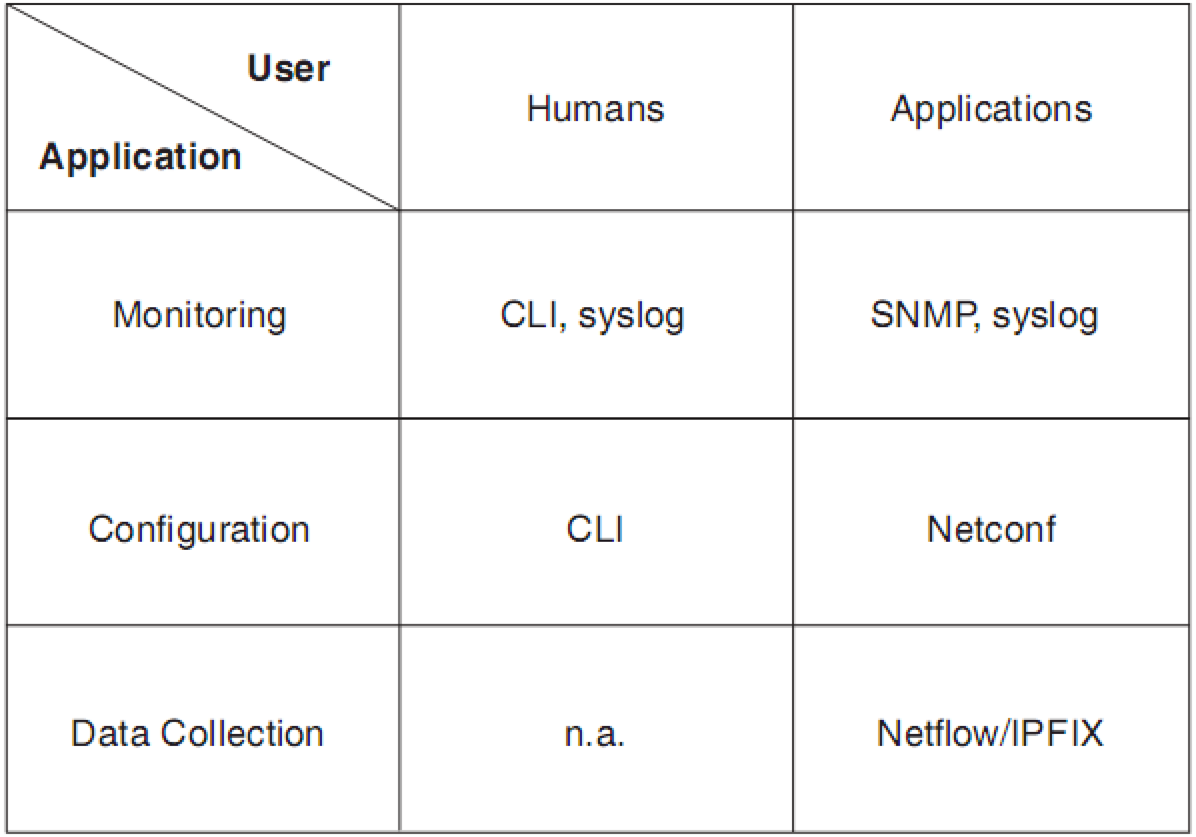


2016

Outline

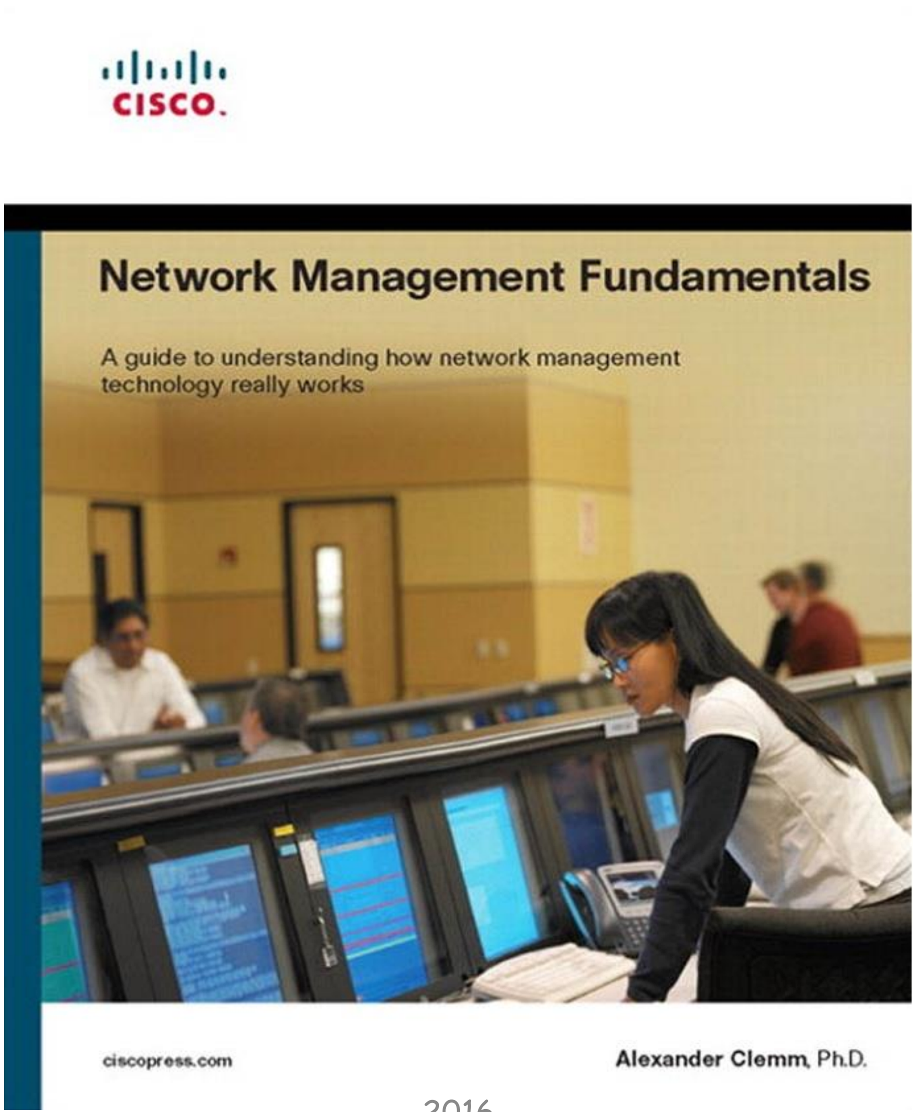
- ❑(SNMP)
- ❑Command Line Interface (CLI)
- ❑Syslog
- ❑Netconf
- ❑Netflow
- ❑Summary

Summary



User Application	Humans	Applications
Monitoring	CLI, syslog	SNMP, syslog
Configuration	CLI	Netconf
Data Collection	n.a.	Netflow/IPFIX

References



CISCO.

Network Management Fundamentals

A guide to understanding how network management technology really works

ciscopress.com

Alexander Clemm, Ph.D.