Networking Review (Architecture)

UT CS361S – Network Security and Privacy

Spring 2021

Lecture Notes

Computing 1960-1980 (ish)

NETWORK

This Photo by Unknown Author is licensed under CC BY-SA-NC

“DUMB” TERMINAL

MAINFRAME

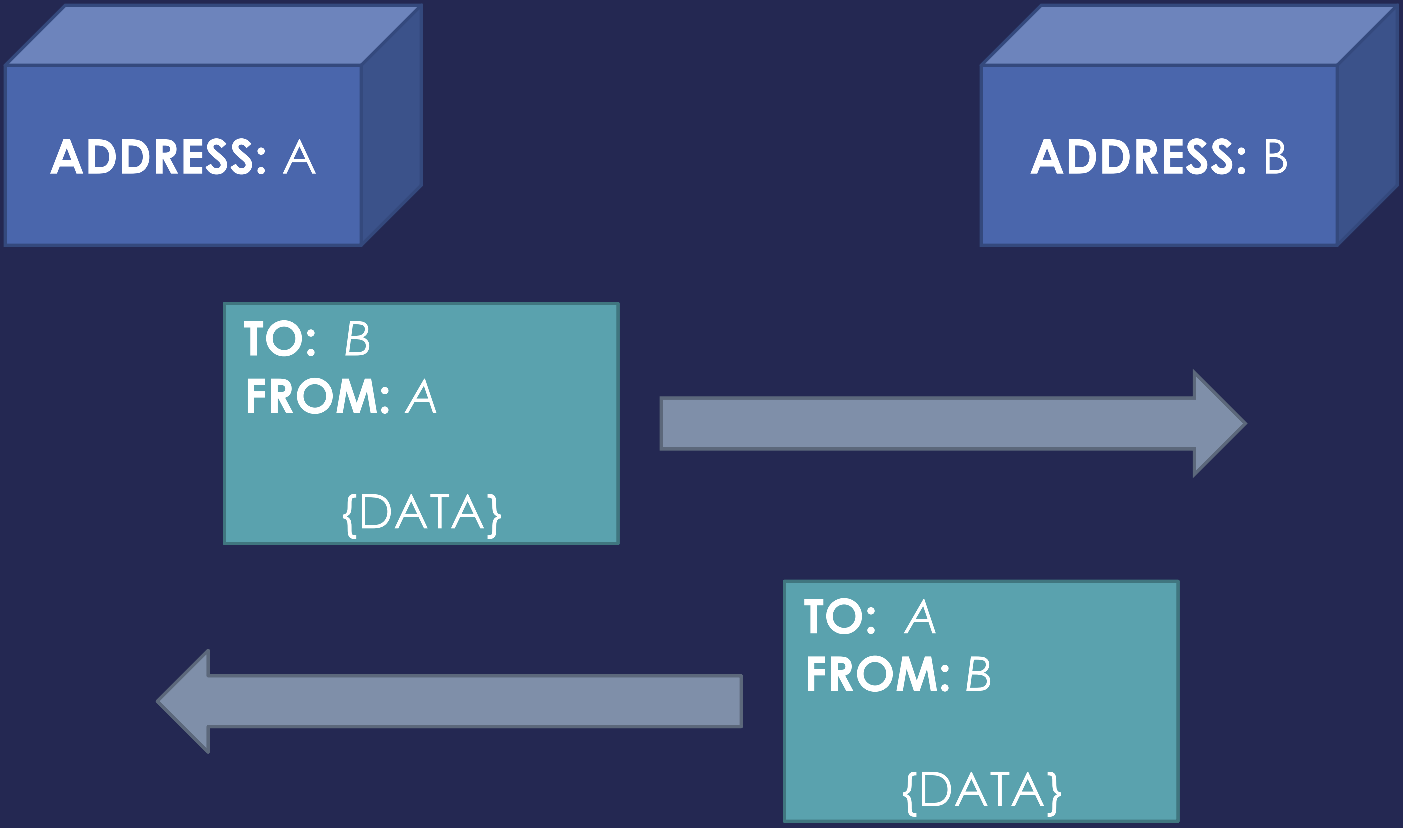
Computing 1980-2000 (ish)

Computing 2000 – Present

NETWORK

Simplified Network Communication

- Two participants; we will call each one ***a "node"***
- Data is transmitted between participants in ***packets*** (chunks)
- Packet includes metadata, usually prepended as a ***header***
- Header includes ***sender's address*** *and* **receiver's address**
- Receiver can "reply" to sender's packets by reversing addresses



ADDRESS: A

ADDRESS: B

TO: *B*
FROM: *A*

{DATA}

TO: *A*
FROM: *B*

{DATA}

Also, Network Discovery

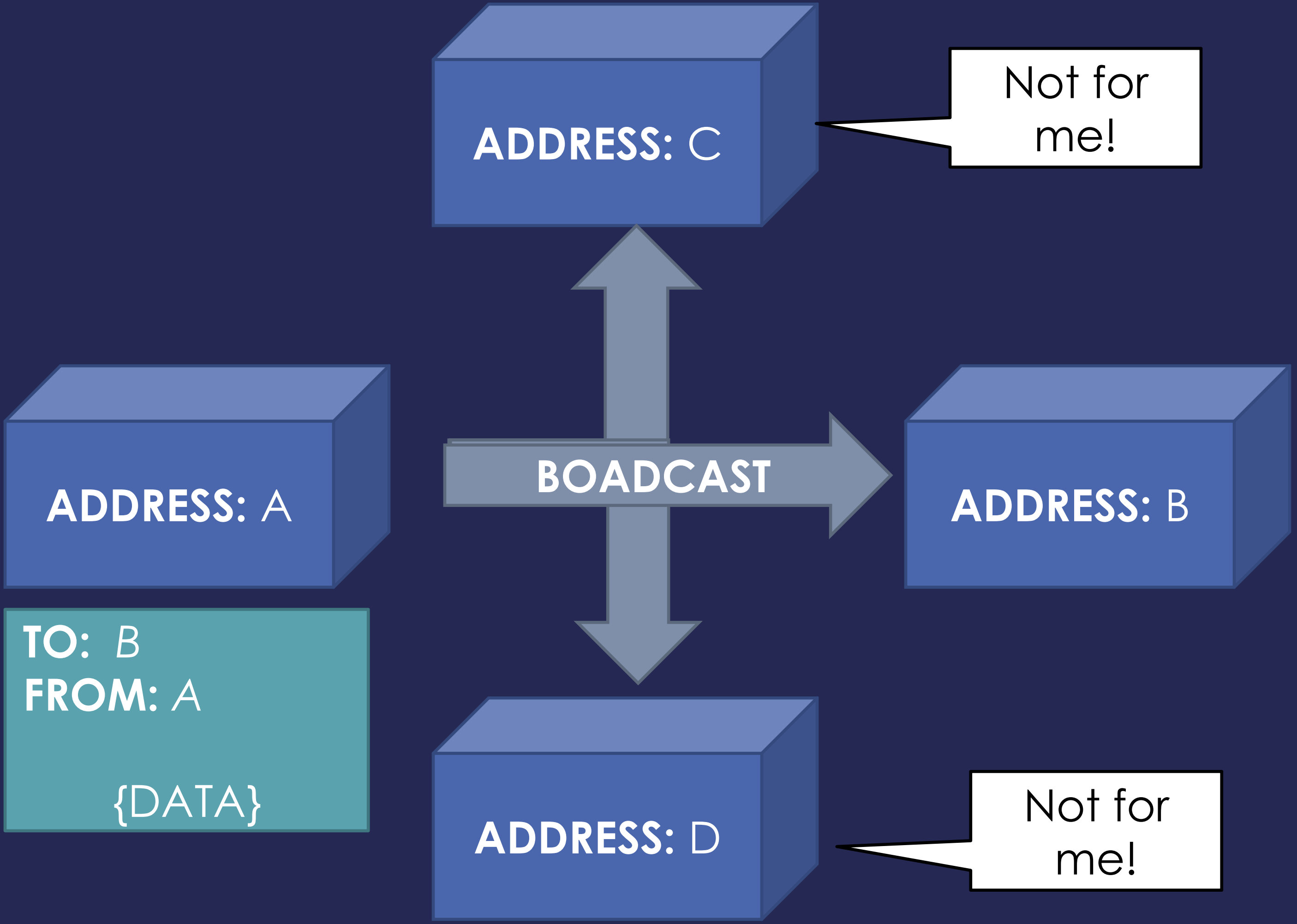
- Many communication patterns require ***discovery***
- Either, the discovery of the existence of nodes
- And/or discovery of how to communicate with a specific node

Local Network Concepts

- Local Area Network (LAN) – Direct node-to-node connection
- Usually involves a “medium”, hence Medium Access Control
- Nodes can send or receive MAC packets w/ MAC addresses
- ***Broadcast*** typically used for discovery

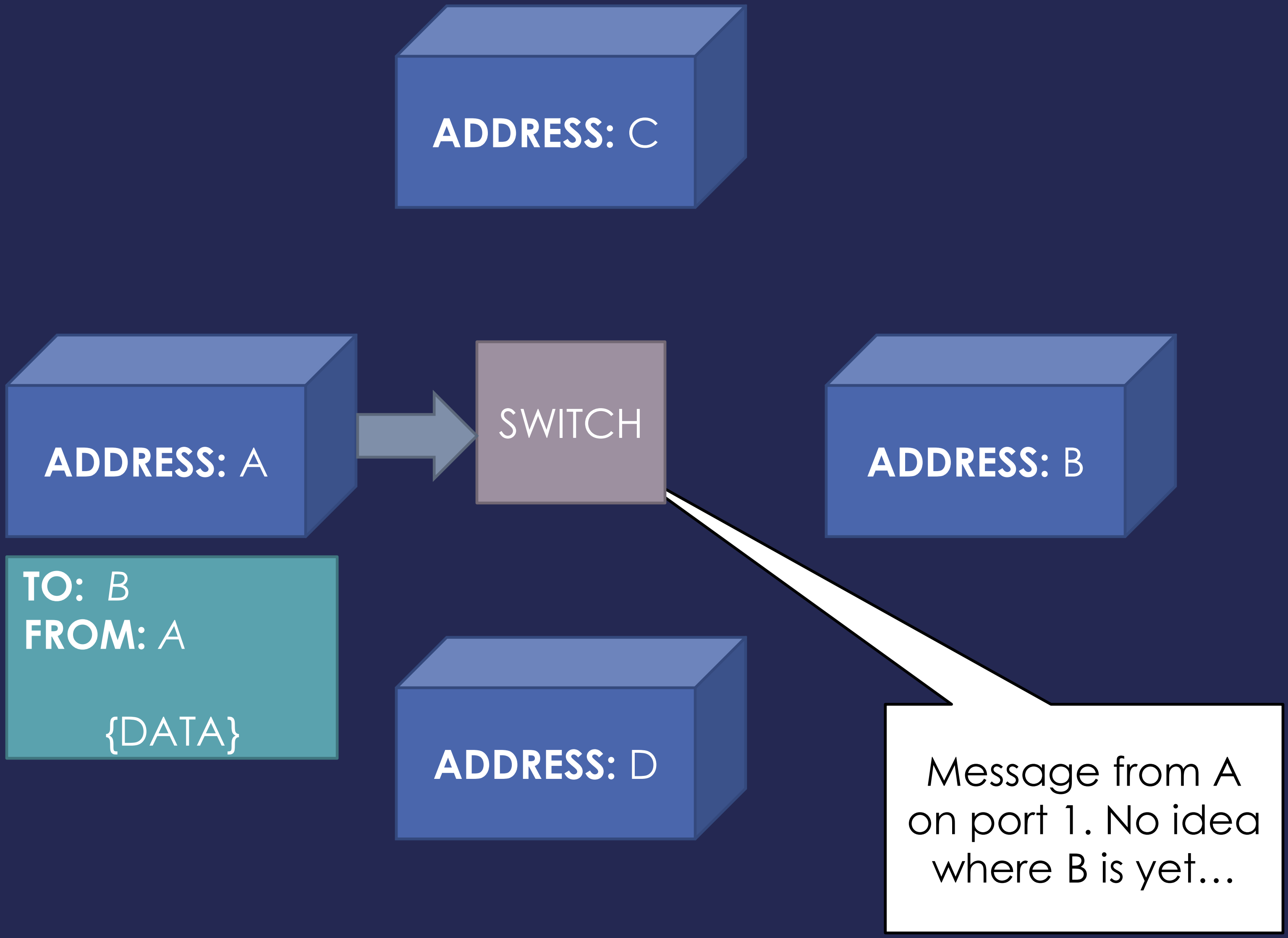
“Old” Ethernet (< 10mbps)

- Broadcast used for ***everything***
- Each node would only respond to packets addressed to it
- Upon detecting a collision, node would randomly back-off


ADDRESS: C
Not for me!
ADDRESS: A
BOADCAST
ADDRESS: B
TO: *B*
FROM: *A*
{DATA}
ADDRESS: D
Not for me!

Modern Ethernet

- Switches ensure the data is delivered to the right node
- Broadcast (flooding) used to find a node the first time
- Much more efficient; enables 100Mbs+



ADDRESS: C

ADDRESS: A

SWITCH

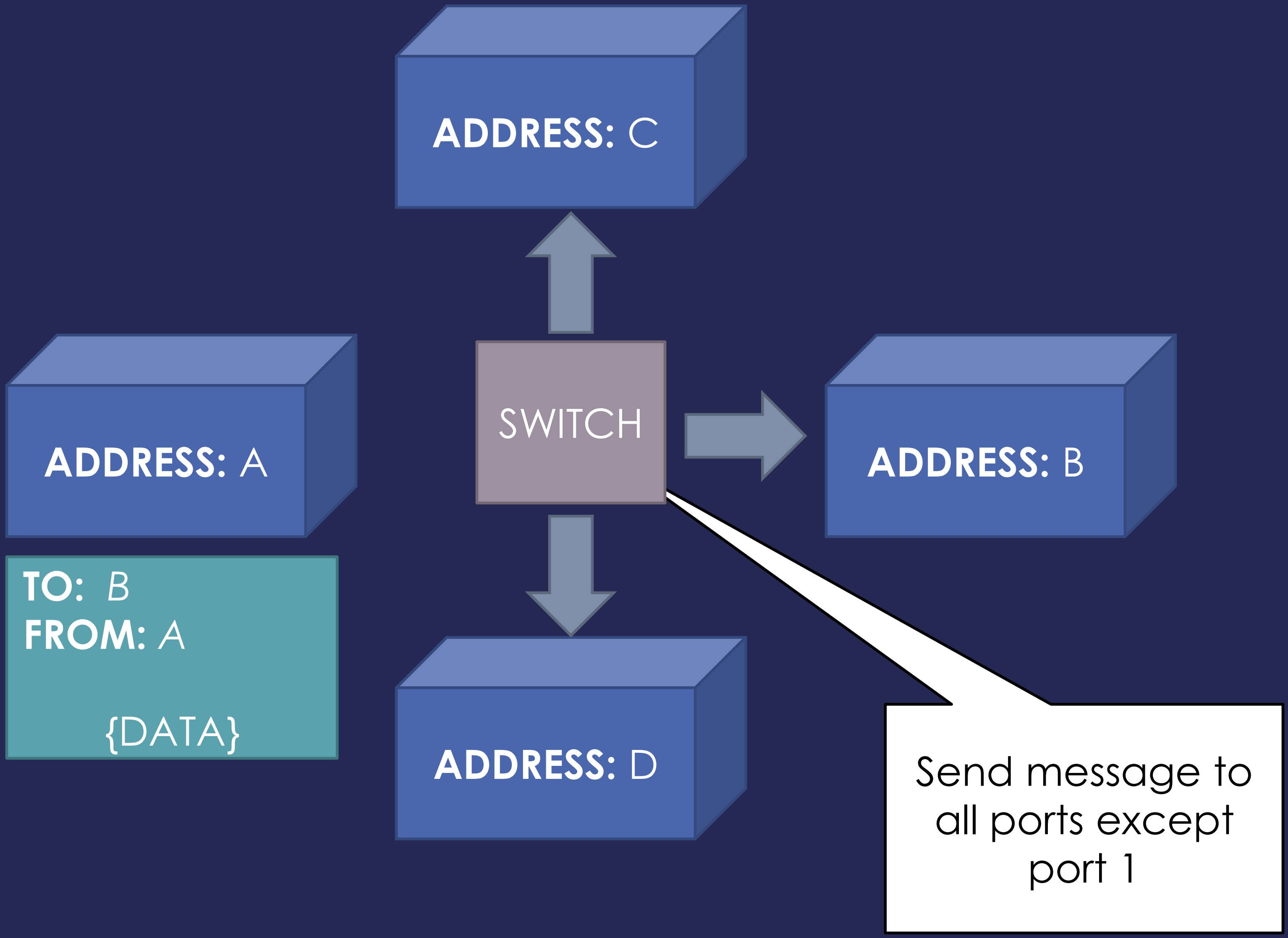
ADDRESS: B

TO: *B*
FROM: *A*

{DATA}

ADDRESS: D

Message from A on port 1. No idea where B is yet…



ADDRESS: C

ADDRESS: A

SWITCH

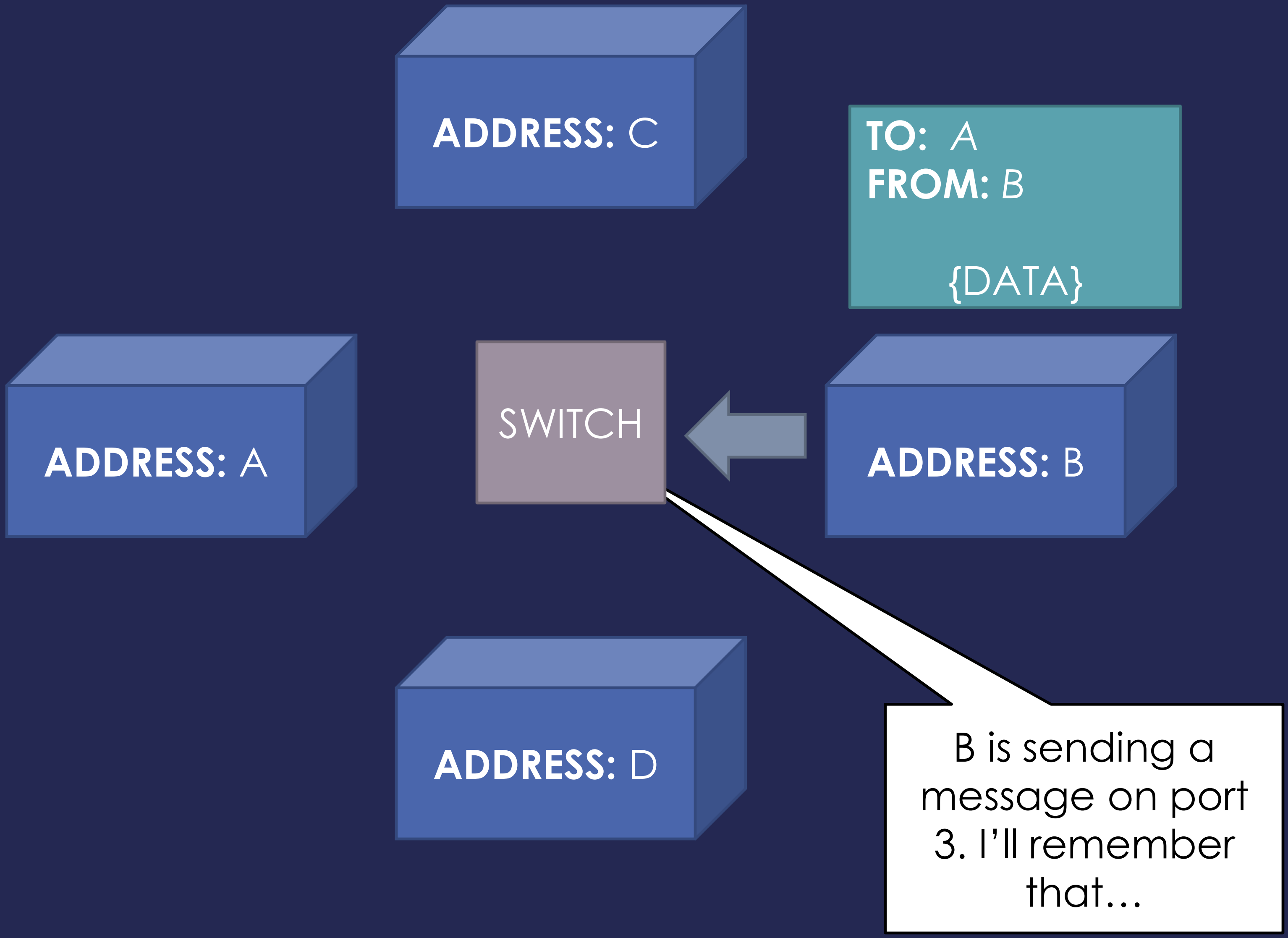
ADDRESS: B

TO: *B*
FROM: *A*

{DATA}

ADDRESS: D

Send message to all ports except port 1



ADDRESS: C

TO: *A*
FROM: *B*

{DATA}

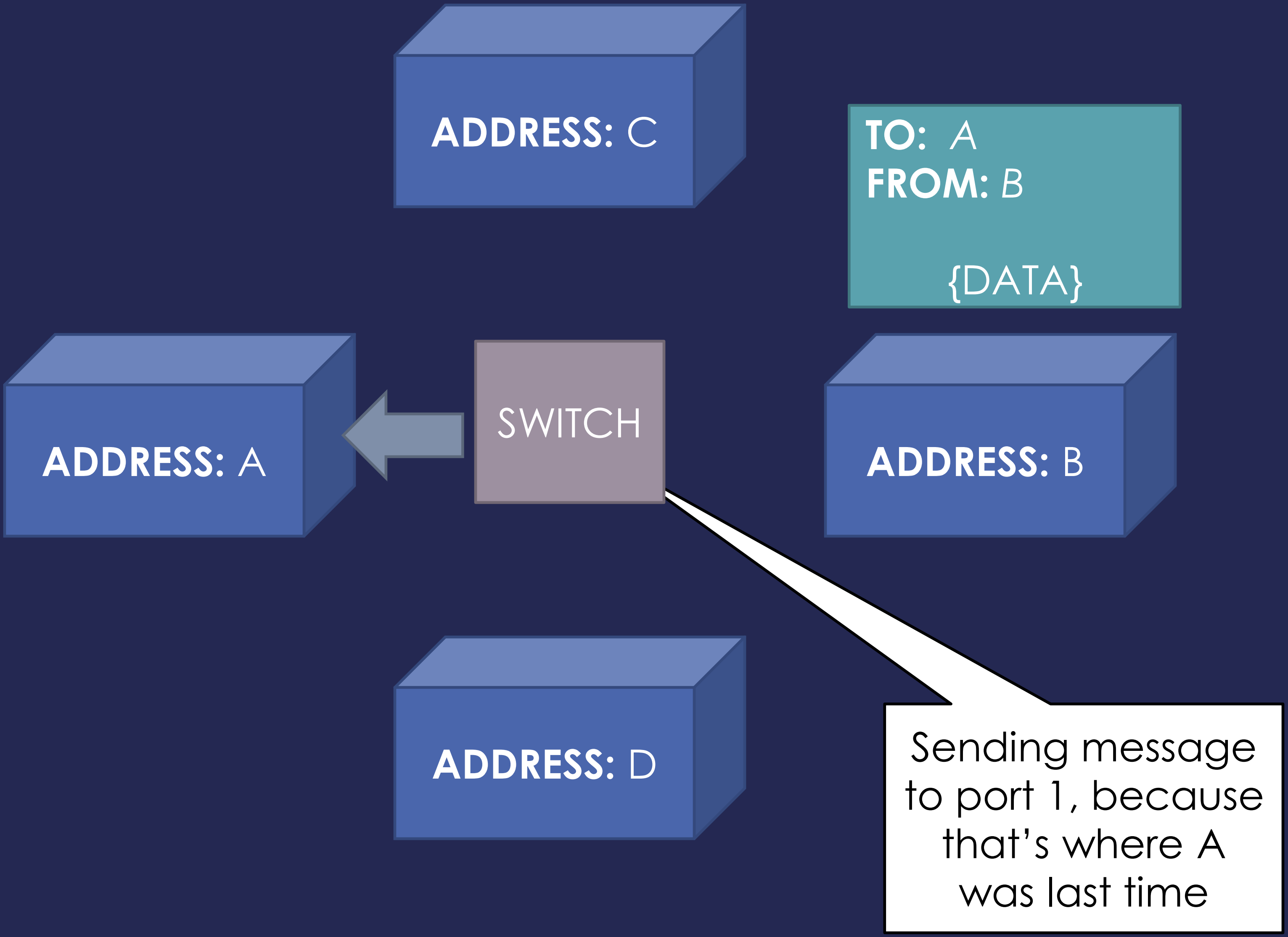
ADDRESS: A

SWITCH

ADDRESS: B

ADDRESS: D

B is sending a message on port 3. I'll remember that…



ADDRESS: C

TO: *A*
FROM: *B*
{DATA}

ADDRESS: A

SWITCH

ADDRESS: B

ADDRESS: D

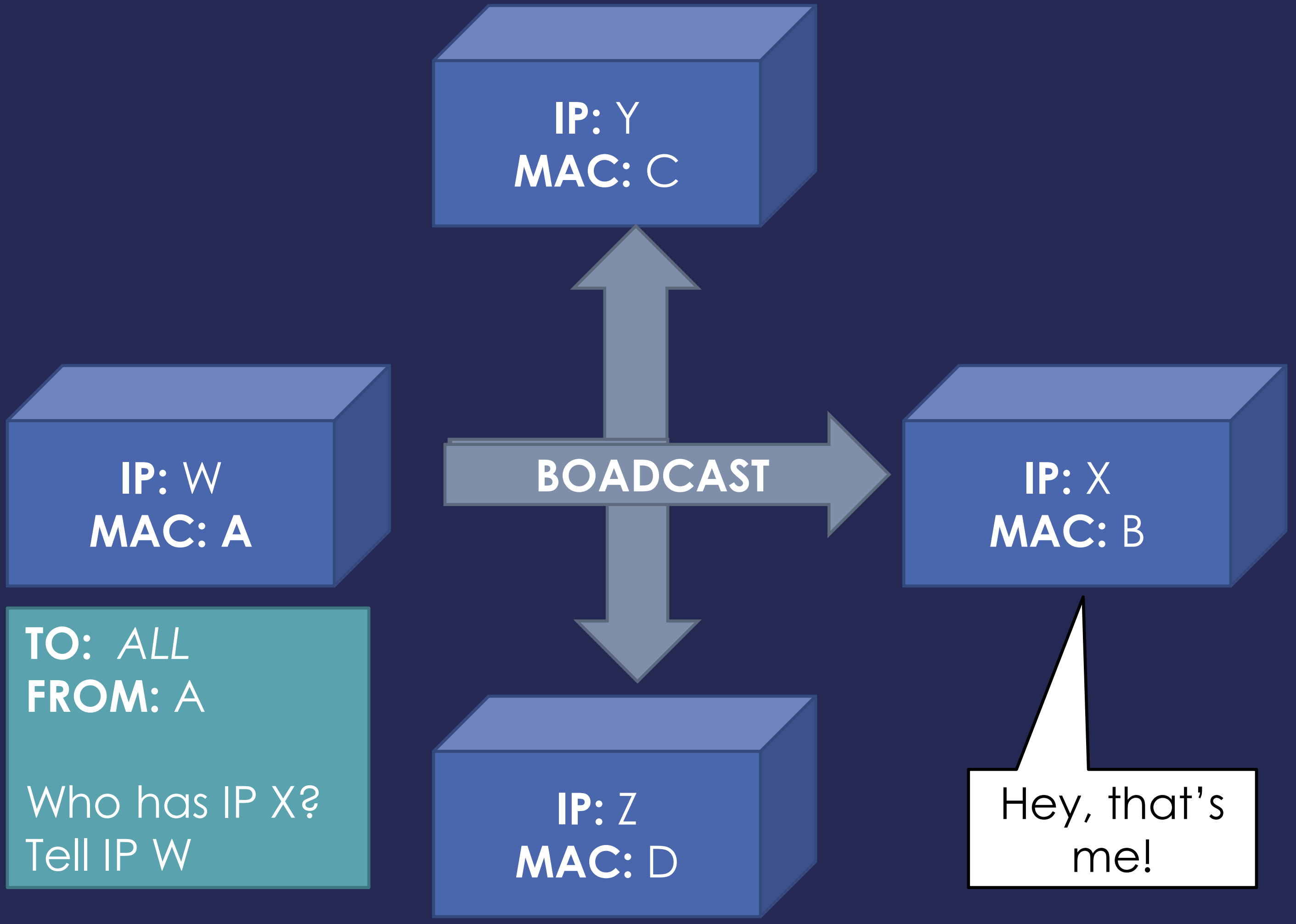
Sending message to port 1, because that's where A was last time

IP Address Mapping

- MAC addresses are great, but app layer doesn’t use them
- Even when communicating on your LAN, you use IP addrs
- To communicate, IP-to-MAC address mapping required
- Enter the Address Resolution Protocol (ARP)

ARP Broadcast Discovery

- Every LAN has an explicit broadcast mechanism
- Ethernet typically uses broadcast address ***FF:FF:FF:FF:FF:FF***
- When IP with unknown MAC mapping needed, broadcast
- Ask all nodes on the LAN "who has IP addr X, tell IP addr W"



IP: Y
MAC: C

IP: W
MAC: A

BOADCAST

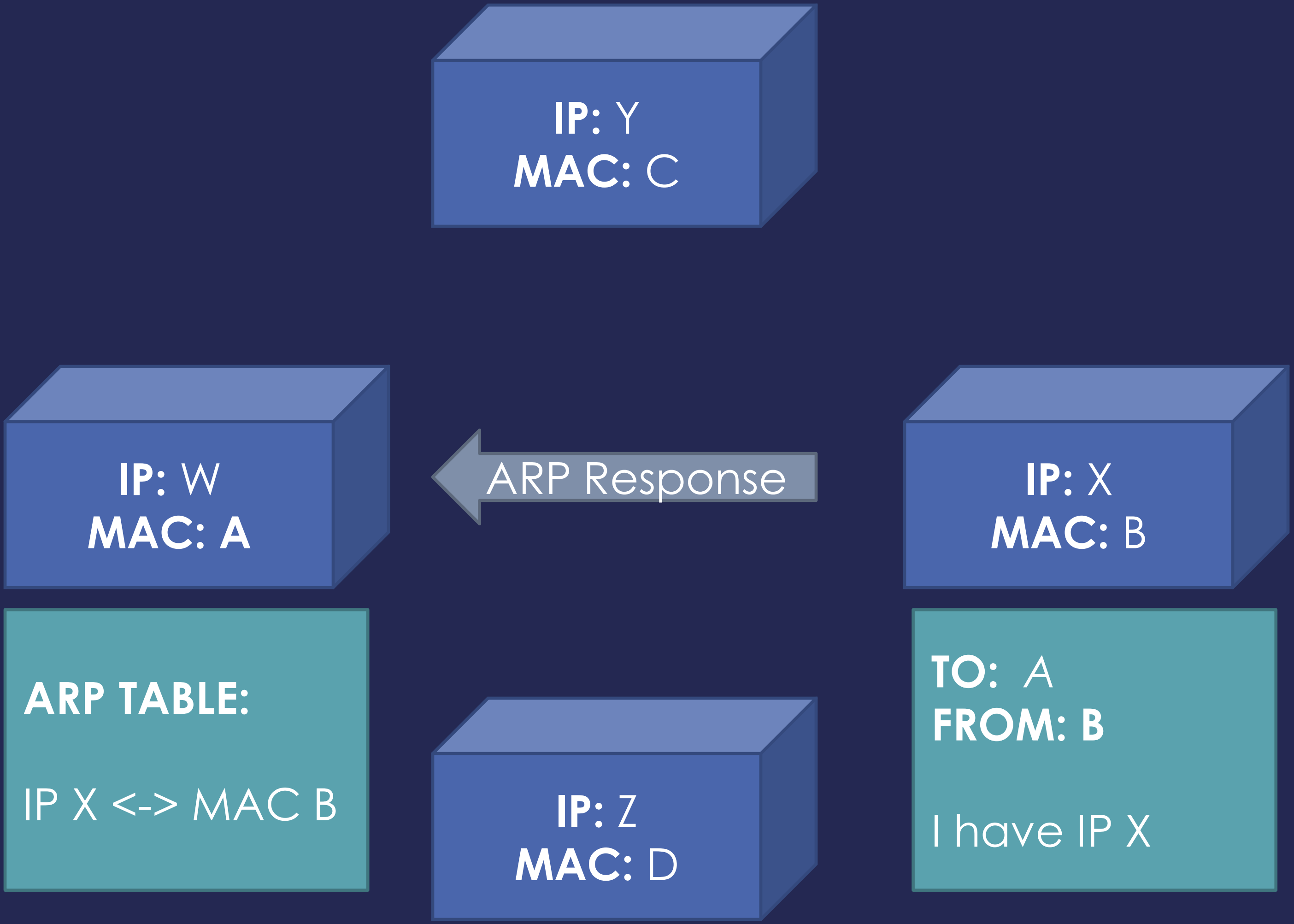
IP: X
MAC: B

TO: *ALL*
FROM: A

Who has IP X?
Tell IP W

IP: Z
MAC: D

Hey, that's me!



IP: Y
MAC: C

IP: W
MAC: A

ARP Response

IP: X
MAC: B

ARP TABLE:

IP X <-> MAC B

IP: Z
MAC: D

TO: *A*
FROM: B

I have IP X

IP to MAC conversion

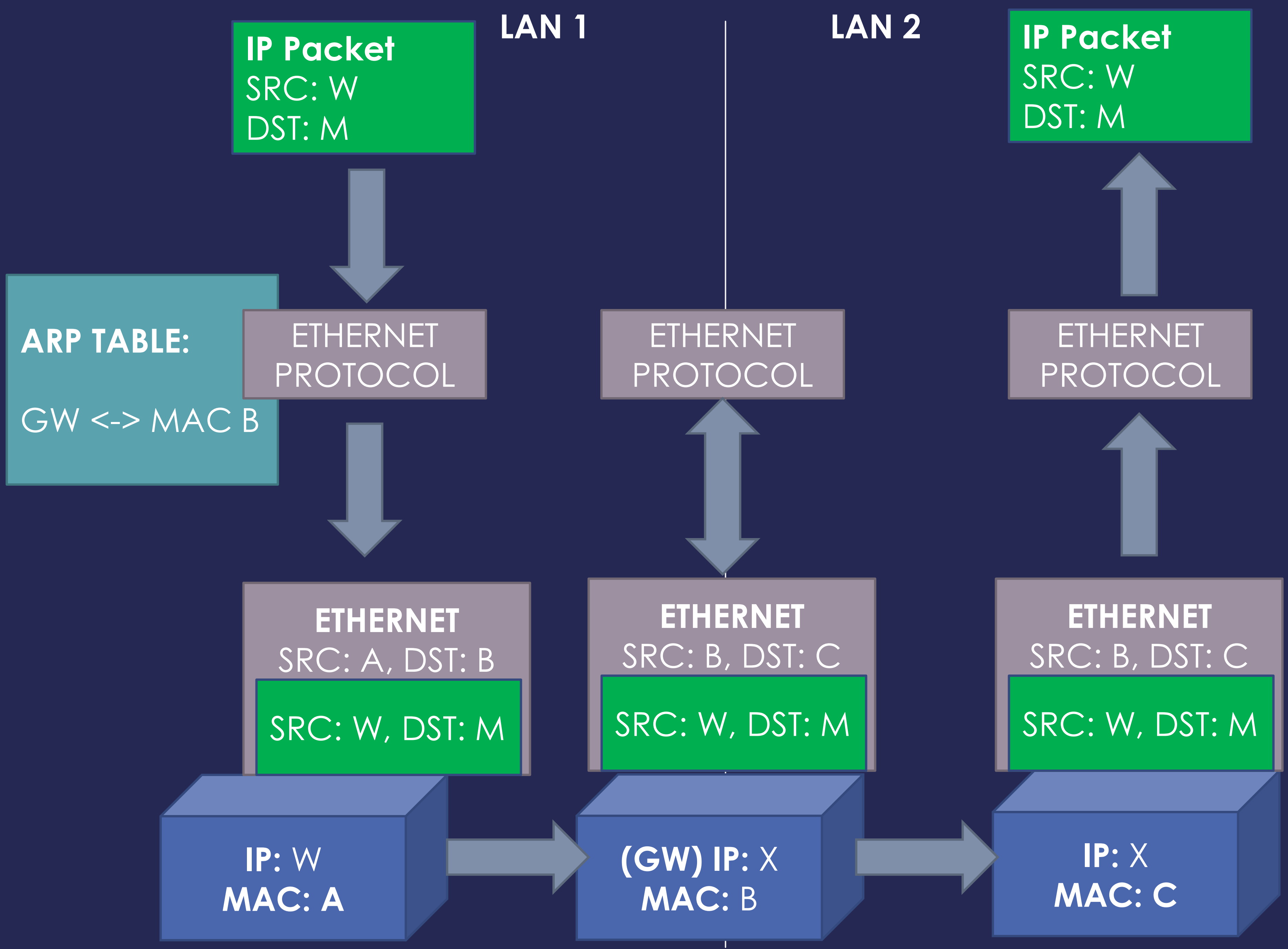
- Once ARP has the IP to MAC in the table, conversion is easy
- Translation occurs in Ethernet layer during “push” from IP
- IP destination addr is looked up in ARP table
- PS, what “layer” in OSI is ARP?

Internetworking Concepts

- We’ve covered how to talk to a LAN node.
- What about a node outside our broadcast domain?
- Requires ***internetwork routing***

Exiting the LAN

- A ***gateway*** node is connected to TWO+ LANs
- For any external IP address, use the MAC addr of the gateway
- (True dest IP addr remains the same!)
- Gateway node will examine the dest IP addr
- Gateway node determines where to send the data next



LAN 1

LAN 2

IP Packet
SRC: W
DST: M

ARP TABLE:

GW <-> MAC B

ETHERNET
PROTOCOL

ETHERNET
SRC: A, DST: B

SRC: W, DST: M

IP: W
MAC: A

ETHERNET
PROTOCOL

ETHERNET
SRC: B, DST: C

SRC: W, DST: M

(GW) IP: X
MAC: B

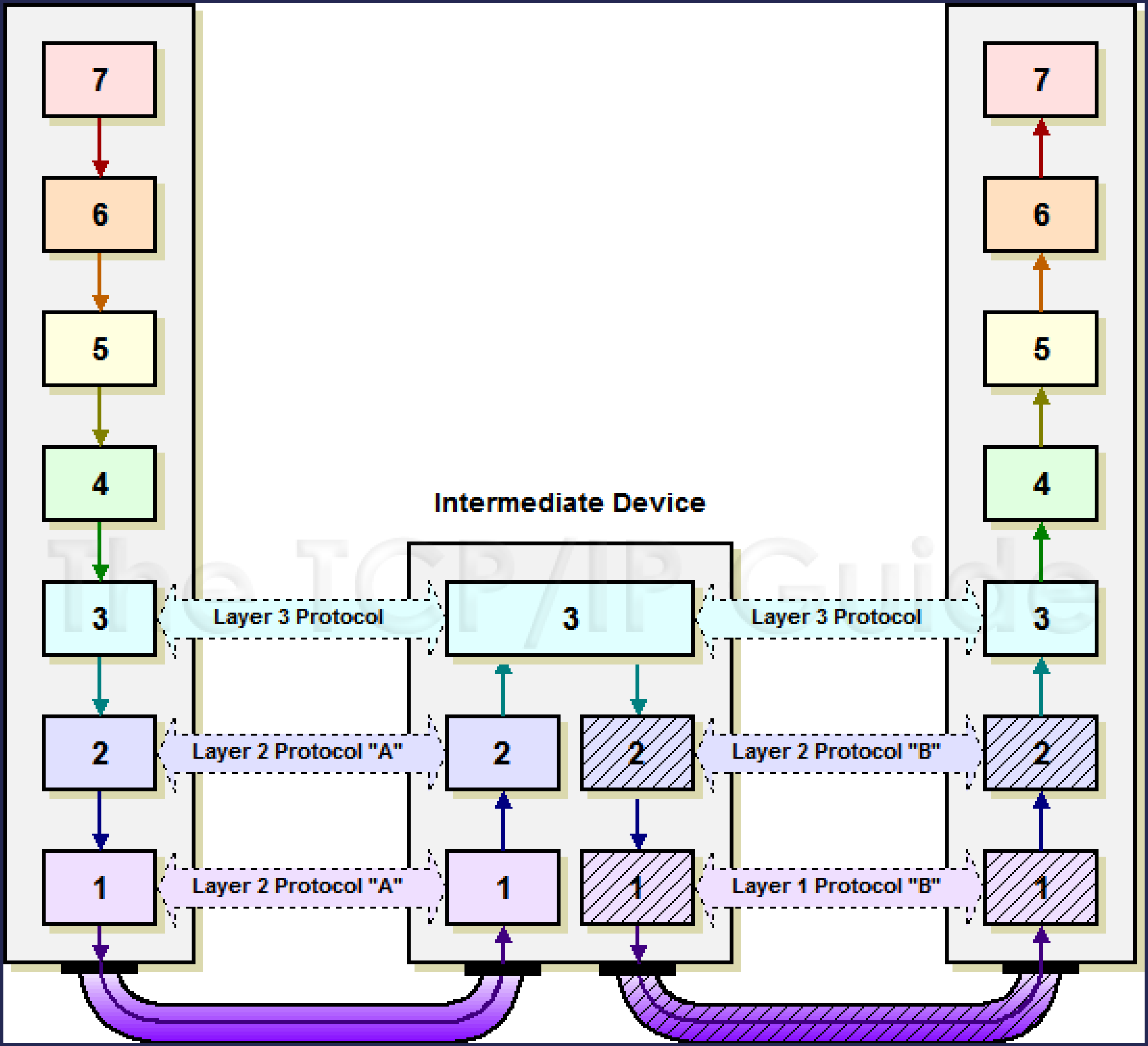
IP Packet
SRC: W
DST: M

ETHERNET
PROTOCOL

ETHERNET
SRC: B, DST: C

SRC: W, DST: M

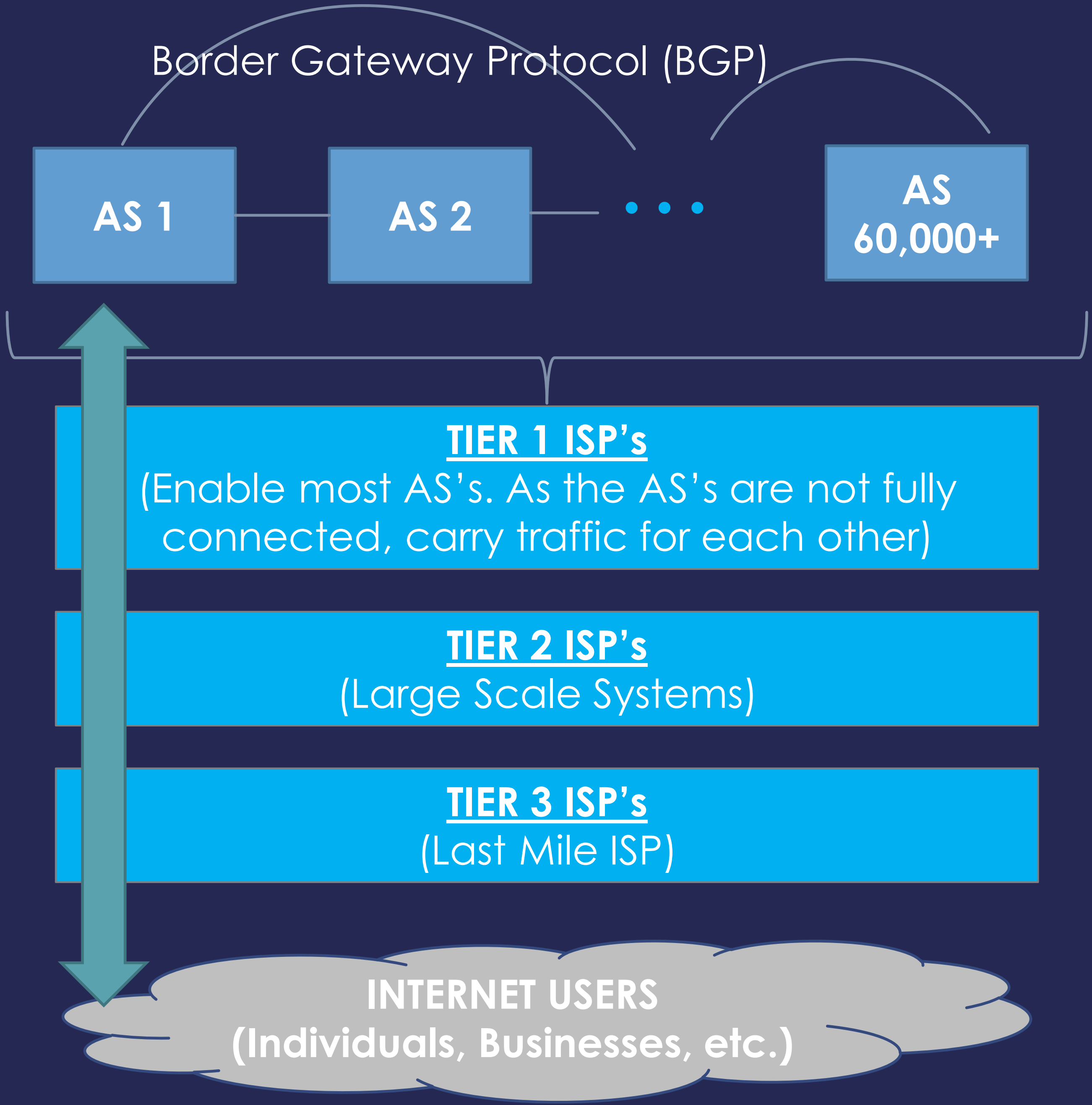
IP: X
MAC: C



http://www.tcpipguide.com/free/t_IndirectDeviceConnectionandMessageRouting.htm

Routing Across the Internet

- Data may have to move up a hierarchy
- You to a Tier-3 ISP to a Tier-2 ISP, to a Tier 1 ISP
- Top level ISP's usually manage the "Autonomous Systems" (AS)
- AS's are the top level of the Internet in terms of routing
- AS's are interconnected and represent the "backbone"



Border Gateway Protocol (BGP)

AS 1

AS 2

• • •

AS 60,000+

TIER 1 ISP’s
(Enable most AS’s. As the AS’s are not fully connected, carry traffic for each other)

TIER 2 ISP’s
(Large Scale Systems)

TIER 3 ISP’s
(Last Mile ISP)

INTERNET USERS
(Individuals, Businesses, etc.)

Default Routes

- Many routers don’t know specifically where to send data
- A default route (or a default router) passes up the hierarchy
- Routers in the AS’s ***have no default route***
- Routers in the AS’s build tables to know where to send data
- (The buck stops here)

Domain Name System (DNS)

- We don’t usually browse the web with IP addresses
- You can, of course. Try browsing to 172.217.13.4
- But what a pain to remember
- DNS is how we convert “google.com” to “172.217.13.4”

Client-Server Concepts

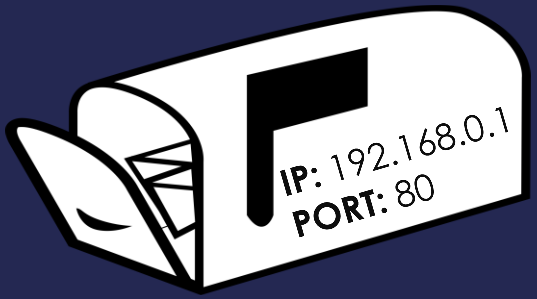
- Put resources in a high-performance, centralized machine
- Clients can be much “dumber” *by comparison*
- Much more efficient
  - Sharing data between devices, applications, and people
  - Access from multiple locations (including hackers!)
  - Time-sharing a central machine is more scalable & cost-effective

Server Abstraction

This Photo by Unknown Author is licensed under CC BY-SA

SERVER ***LISTENS*** FOR INCOMING REQUESTS

Preview of TCP/IP



This Photo by Unknown Author is licensed under CC BY-NC

This Photo by Unknown Author is licensed under CC BY-SA

Now I have an Address/Port! Maybe I'll get Requests!

SERVER HAS AN **IP ADDRESS** AND **TCP PORT**

Meanwhile, Client Abstraction

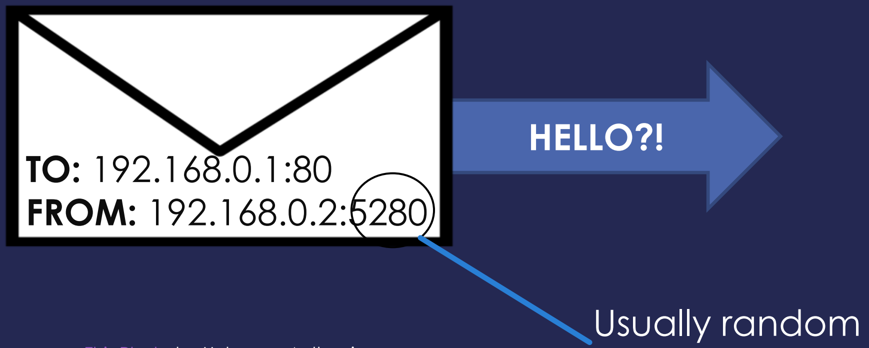
HELLO?!

This Photo by Unknown Author is licensed under CC BY-NC

CLIENT ***CONNECTS*** TO MAKE OUTBOUND REQUESTS

TCP/IP Again

This Photo by Unknown Author is licensed under CC BY-NC



TO: 192.168.0.1:80
FROM: 192.168.0.2:5280

Usually random

This Photo by Unknown Author is licensed under CC BY-NC

CLIENT ***CONNECTS*** TO MAKE OUTBOUND REQUESTS

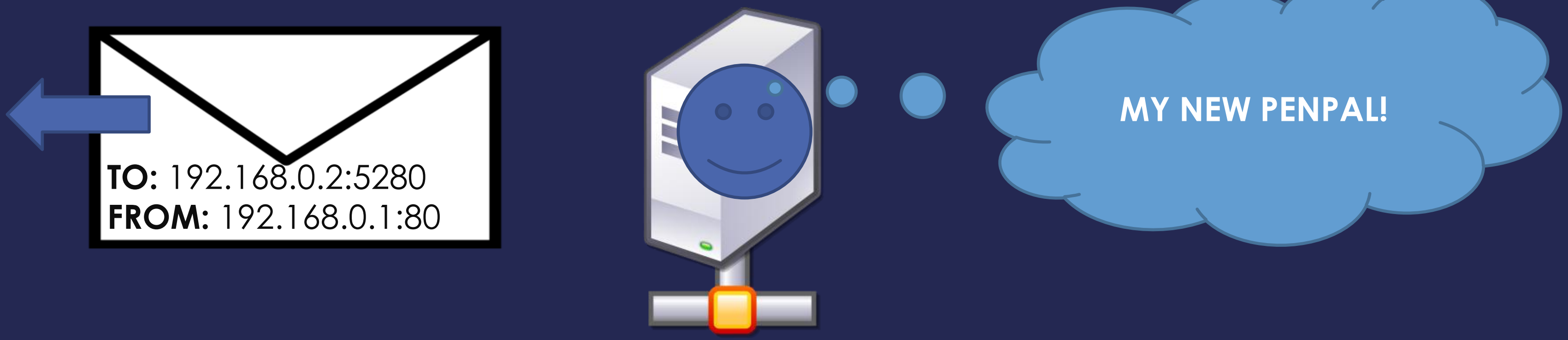
Incoming Request

TO: 192.168.0.1:80
FROM: 192.168.0.2:5280

I GOT A REQUEST!!!

SERVER RECEIVES REQUEST

Request Response



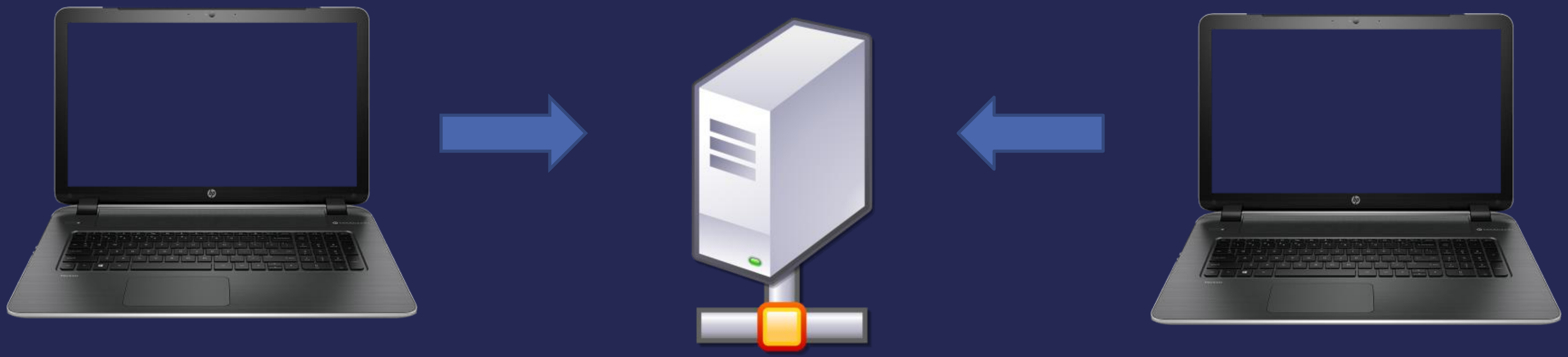
TO: 192.168.0.2:5280
FROM: 192.168.0.1:80

MY NEW PENPAL!

This Photo by Unknown Author is licensed under CC BY-SA

SERVER **INVERTS TO/FROM** FOR RESPONSE

Server Listens to Many Requests



SERVER USES **(SRC IP, SRC PORT, DST IP, DST PORT)*** TO MULTIPLEX

This is how one server on one port (e.g., webserver) handles many clients, even from the same computer

192.168.1.1, 5280, 192.168.1.2, 80	**Web Server Process 1**
192.168.1.1, 9019, 192.168.1.2, 80	Web Server Process 2