EECS 489
Computer Networks

Fall 2020

Mosharaf Chowdhury

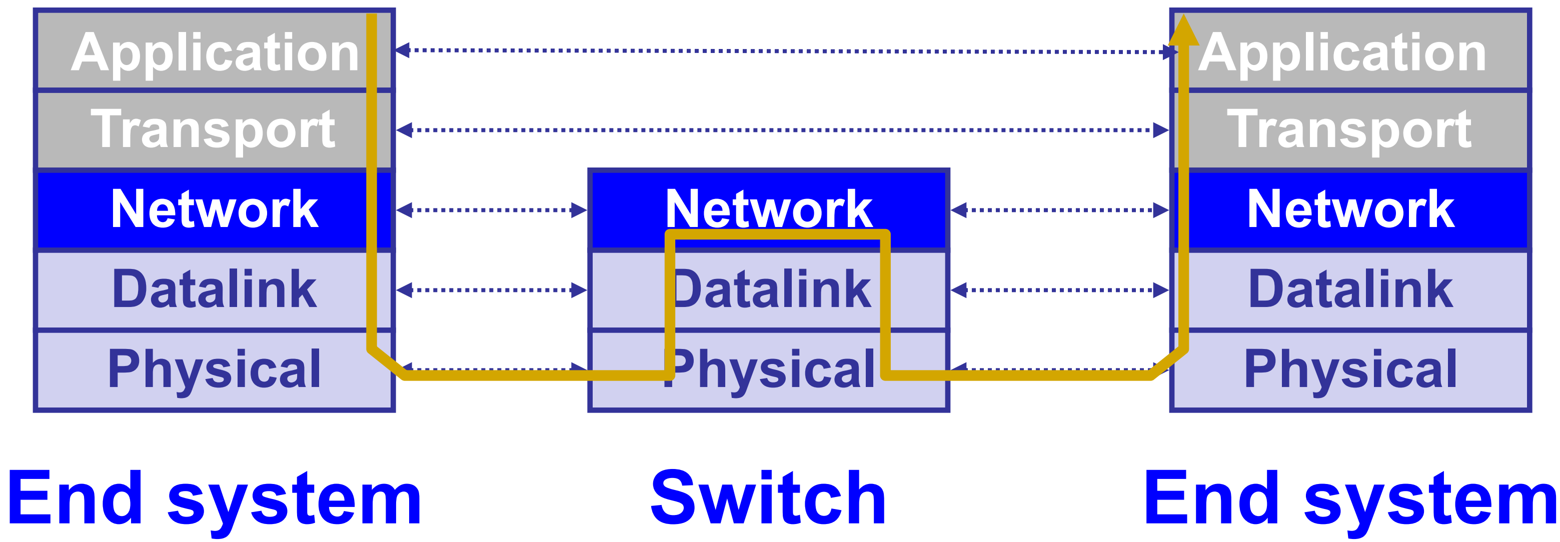
Material with thanks to Aditya Akella, Sugih Jamin, Philip Levis, Sylvia Ratnasamy, Peter Steenkiste, and many other colleagues.

Topics

- Network layer (lectures 12–16)
  - Intra-domain routing
  - Inter-domain routing
  - SDN

Network layer

- Present everywhere
- Performs addressing, forwarding, and routing, among other tasks

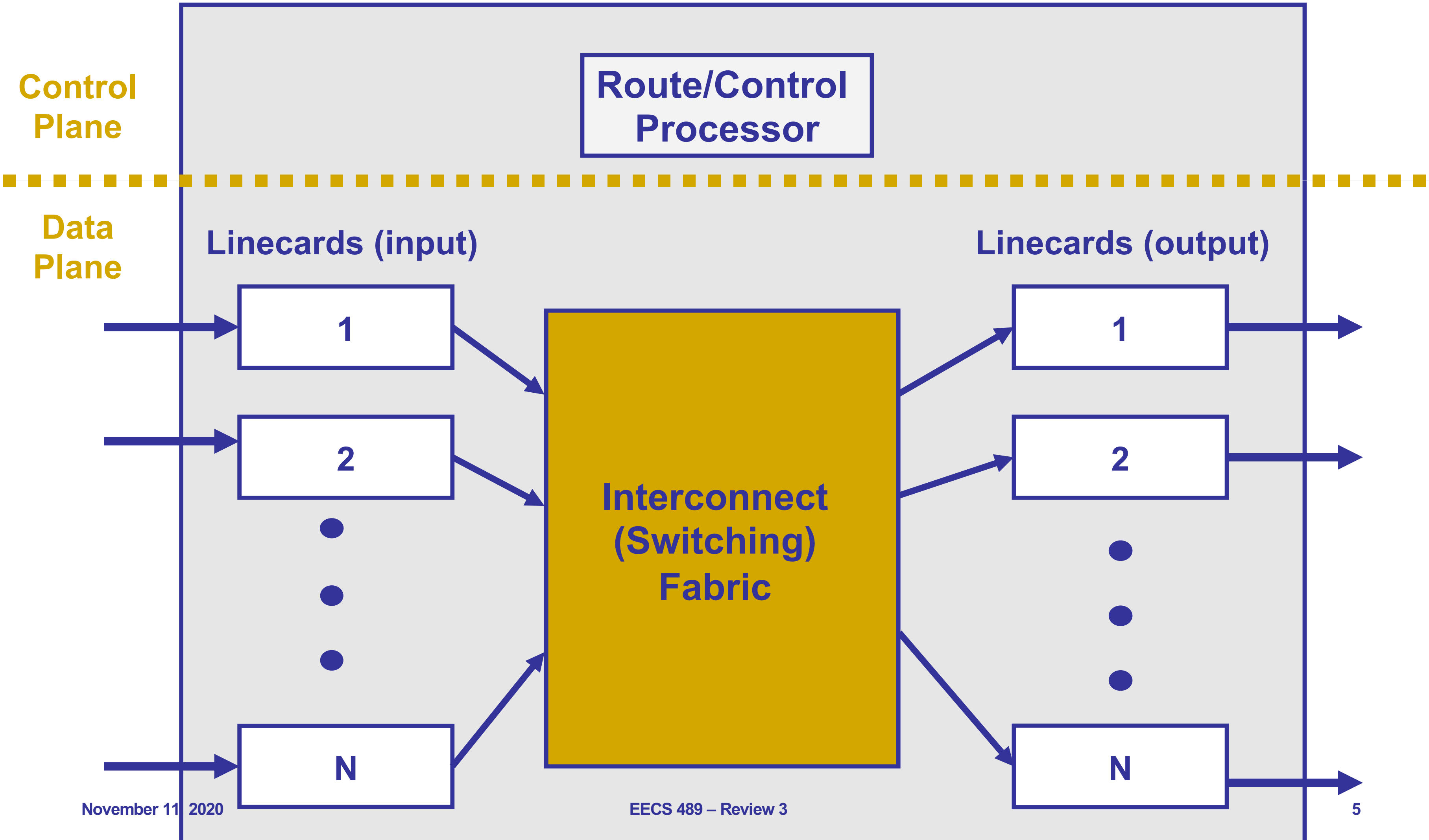


Forwarding vs. routing

- Forwarding: “data plane”
  - Directing one data packet
  - Each router using local routing state
- Routing: “control plane”
  - Computing the forwarding tables that guide packets
  - Jointly computed by routers using a distributed algorithm

- Very different timescales!

What's inside a router?



Routing: Local vs. global view

- *Local* routing state is the forwarding table in a single router
  - By itself, the state in a single router cannot be evaluated
  - It must be evaluated in terms of the global context
- *Global* state refers to the collection of forwarding tables in each of the routers
  - Global state determines which paths packets take

“Valid” routing state

- Global state is “valid” if it produces forwarding decisions that always deliver packets to their destinations
- Goal of routing protocols: compute valid state
  - How can we tell if routing state if valid?

Necessary and sufficient condition

- Global routing state is valid *if and only if*:
  - There are no dead ends (other than destination)
  - There are no loops
- A dead end is when there is no outgoing link (next-hop)
  - A packet arrives, but the forwarding decision does not yield any outgoing link
- A loop is when a packet cycles around the same set of nodes forever

Least-cost routes

- Least-cost routes provide an easy way to avoid loops
  - No reasonable cost metric is minimized by traversing a loop
- Least-cost paths form a spanning tree for each destination rooted at that destination

Intra-domain routing

- Link-state (LS) routing protocol
  - Dijkstra's algorithm
  - Broadcast neighbors' info to everyone
- Distance vector (DV) routing protocol
  - Bellman-Ford algorithm
  - Gossip to neighbors about everyone

Link-state routing

- Every router knows its local "link state"
  - Router u: "(u,v) with cost=2; (u,x) with cost=1"
- Each router floods its local link state to all other routers in the network
  - Does so periodically or when its link state changes
- Every router learns the entire network graph
  - Each runs Dijkstra's Shortest-Path First (SPF) algorithm locally to compute forwarding table

Distance-vector protocol

- Link-state routing protocol
  - Each node broadcasts its local information
- Distance-vector routing protocol
  - The opposite (sort of)
  - Each node tells its neighbors about its global view
- Use Bellman-Ford equation

Similarities between LS and DV routing

- Both are shortest-path based routing
  - Minimizing cost metric (link weights) a common optimization goal
    - Routers share a common view as to what makes a path “good” and how to measure the “goodness” of a path
- Due to shared goal, commonly used inside an organization
  - RIP and OSPF are mostly used for intra-domain routing

Comparison of LS and DV routing

Messaging complexity

- LS: with N nodes, E links, O(NE) messages sent
- DV: exchange between neighbors only

Speed of convergence

- LS: relatively fast
- DV: convergence time varies
  - **Count-to-infinity** problem

Robustness: what happens if router malfunctions?

- LS:
  - Node can advertise incorrect link cost
  - Each node computes its *own* table
- DV:
  - Node can advertise incorrect path cost
  - Each node's table used by others (errors propagate)

INTER-DOMAIN ROUTING

Autonomous systems (AS)

- An AS is a network under a single administrative control
  - Currently over 70,000 ASes
  - Updated daily at http://www.cidr-report.org/as2.0/
- ASes are sometimes called "domains"
- Each AS is assigned a unique identifier (ASN)
  - E.g., University of Michigan owns ASNs 177 to 180

Addressing is key to scalable inter-domain routing

- Ability to aggregate addresses is crucial for
  - State: Small forwarding tables at routers
    - Much less than the number of hosts
  - Churn: Limited rate of change in routing tables

Classful addressing

- Three classes
  - 8-bit network prefix (Class A),
  - 16-bit network prefix (Class B), or
  - 24-bit network prefix (Class C)
- Example: an organization needs 500 addresses.
  - A single class C address is not enough (<500 hosts)
  - Instead, a class B address is allocated (~65K hosts)
    - » Huge waste!

CIDR: Classless inter-domain routing

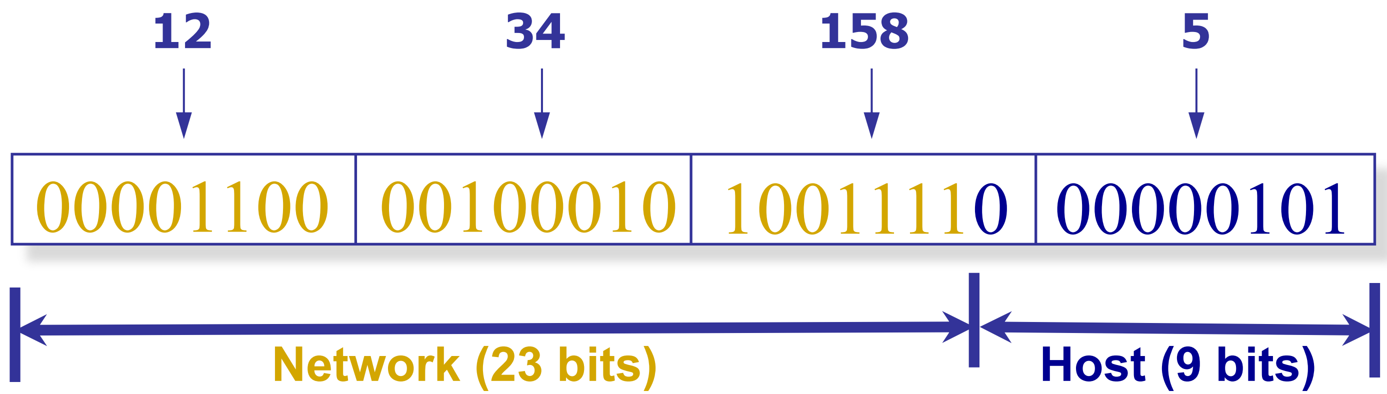
- Flexible division between network and host addresses
- Offers a better tradeoff between size of the routing table and efficient use of the IP address space

Allocation done hierarchically

- Internet Corporation for Assigned Names and Numbers (ICANN) gives large blocks to…
- Regional Internet Registries, such as the American Registry for Internet Names (ARIN), which give blocks to…
- Large institutions (ISPs), which give addresses to…
- Individuals and smaller institutions
- FAKE Example:
  - ICANN → ARIN → AT&T → UMICH → EECS

Hierarchy in IP addressing

- 32 bits are partitioned into a prefix and suffix components
- Prefix is the network component; suffix is the host component



- Inter-domain routing operates on network prefix

5-MINUTE BREAK!

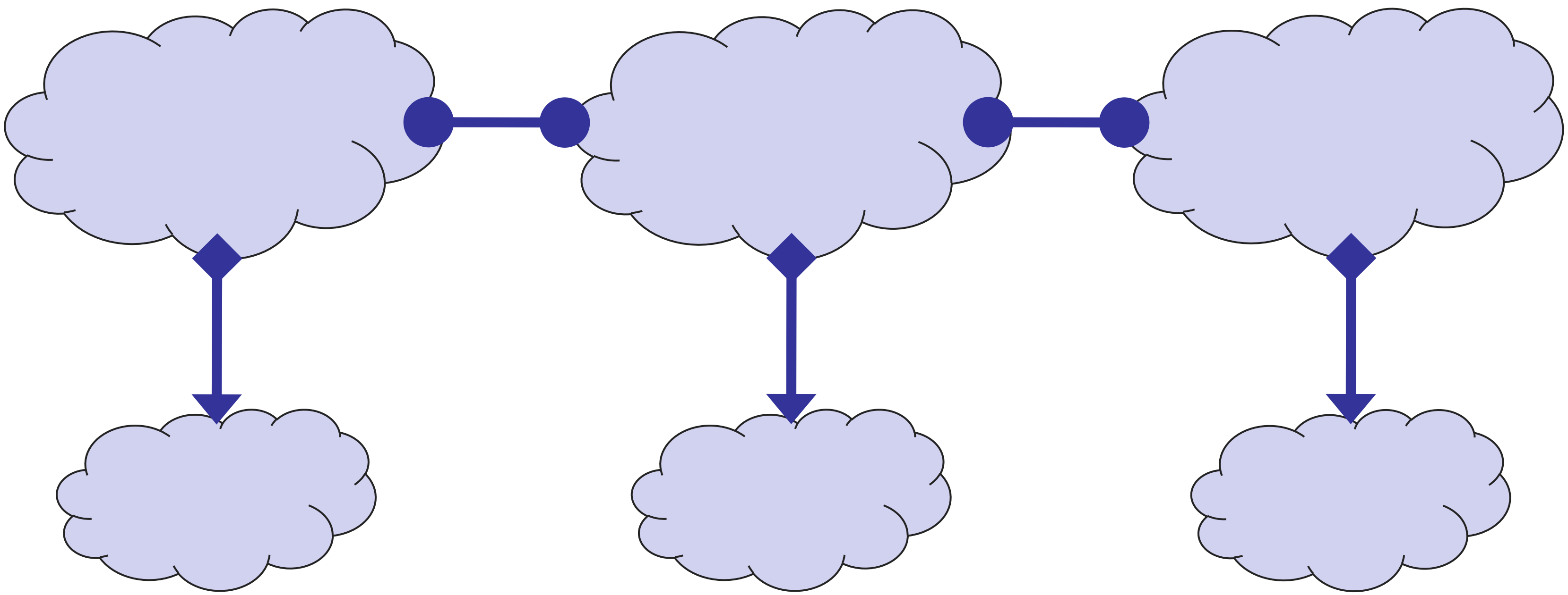
Announcements

- Prof. Jennifer Rexford will be giving a distinguished lecture on Nov 13 2:45-3:45PM
  - Topic: Networks Capable of Change
  - https://eecs.engin.umich.edu/event/networks-capable-of-change/

Administrative structure shapes Inter-domain routing

- ASes want freedom to pick routes based on policy
- ASes want autonomy
- ASes want privacy

Business relationships



Relations between ASes

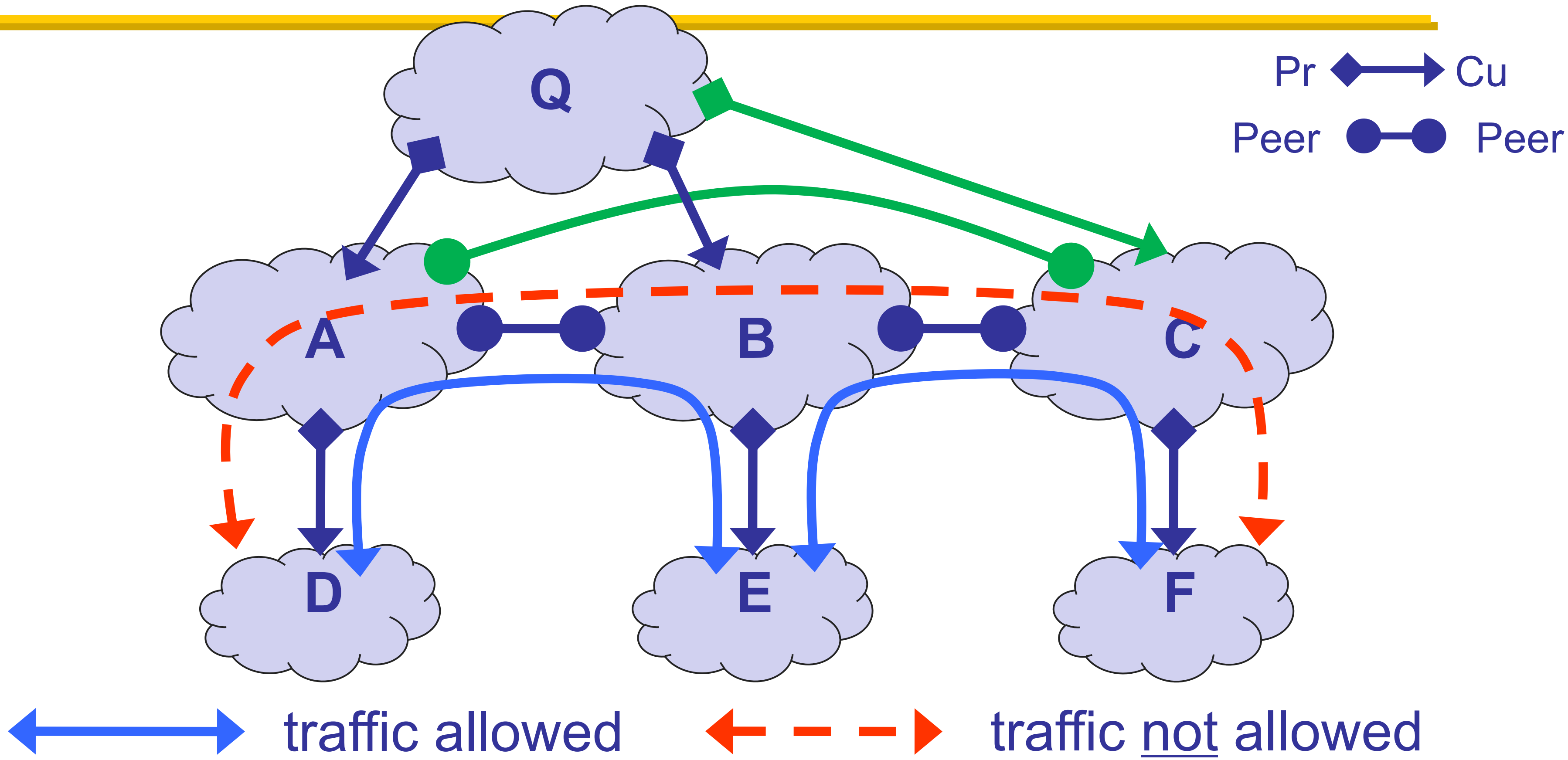
provider ◆——▶ customer

peer ●——● peer

Business implications

- Customers pay provider
- Peers don't pay each other

Routing follows the money!



- ASes provide "transit" between their customers
- Peers do not provide transit between other peers

BGP inspired by Distance-Vector with four differences

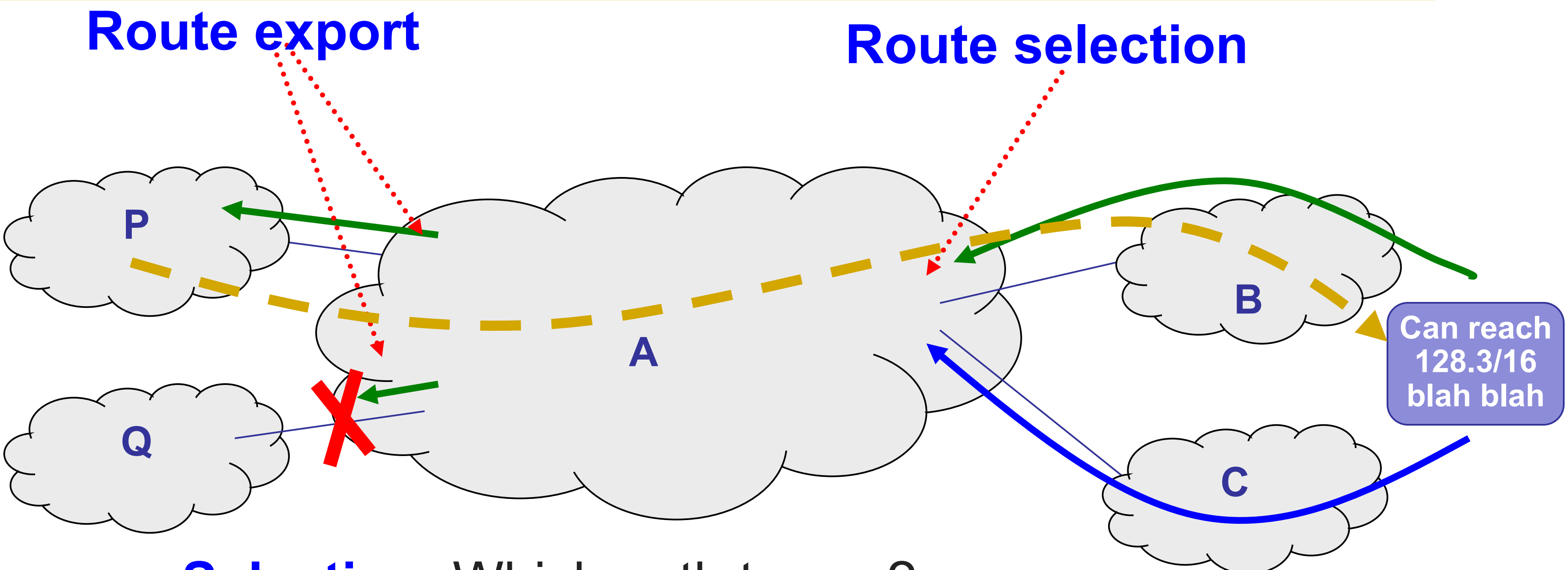
- Shortest-path routes may not be picked to enforce policy
- Path-Vector routing to avoid loops
- Selective route advertisement may affect reachability
- Routes may be aggregated for scalability

BGP: Basic idea

An AS advertises (“exports”) its best routes to one or more IP prefixes

Each AS selects the “best” route it hears advertised for a prefix

Policy dictates how routes are "selected" and "exported"



- **Selection**: Which path to use?
  - Controls whether/how traffic leaves the network
- **Export**: Which path to advertise?
  - Controls whether/how traffic enters the network

Typical export policy

Destination prefix advertised by…	Export route to…
Customer	Everyone (providers, peers, other customers)
Peer	Customers
Provider	Customers

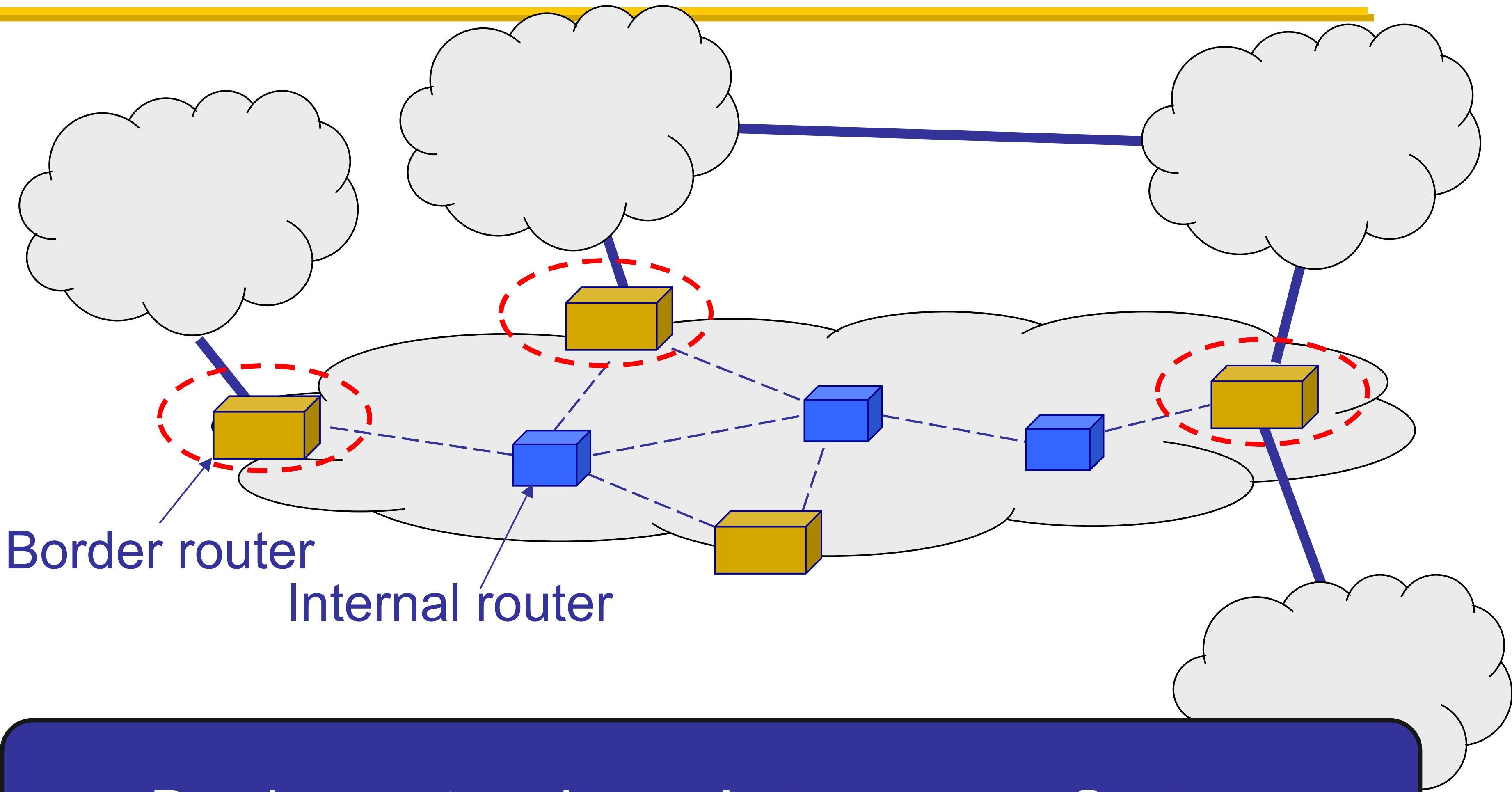
We'll refer to these as the "Gao-Rexford" rules (capture common – but not required! – practice)

Selection using attributes

- Rules for route selection in priority order

Priority	Rule	Remarks
1	LOCAL PREF	Pick highest LOCAL PREF
2	ASPATH	Pick shortest ASPATH length
3	MED	Lowest MED preferred
4	eBGP > iBGP	Did AS learn route via eBGP (preferred) or iBGP?
5	iBGP path	Lowest IGP cost to next hop (egress router)
6	Router ID	Smallest next-hop router's IP address as tie-breaker

Who speaks BGP?



Border routers in an Autonomous System

eBGP, iBGP, and IGP

- eBGP: BGP sessions between border routers in different ASes
  - Learn routes to external destinations
- iBGP: BGP sessions between border routers and other routers within the same AS
  - Distribute externally learned routes internally
- IGP: “Interior Gateway Protocol” = Intra-domain routing protocol
  - Provide internal reachability via shortest path
  - E.g., OSPF, RIP

SOFTWARE-DEFINED AND PROGRAMMABLE NETWORKS

“The Power of Abstraction”

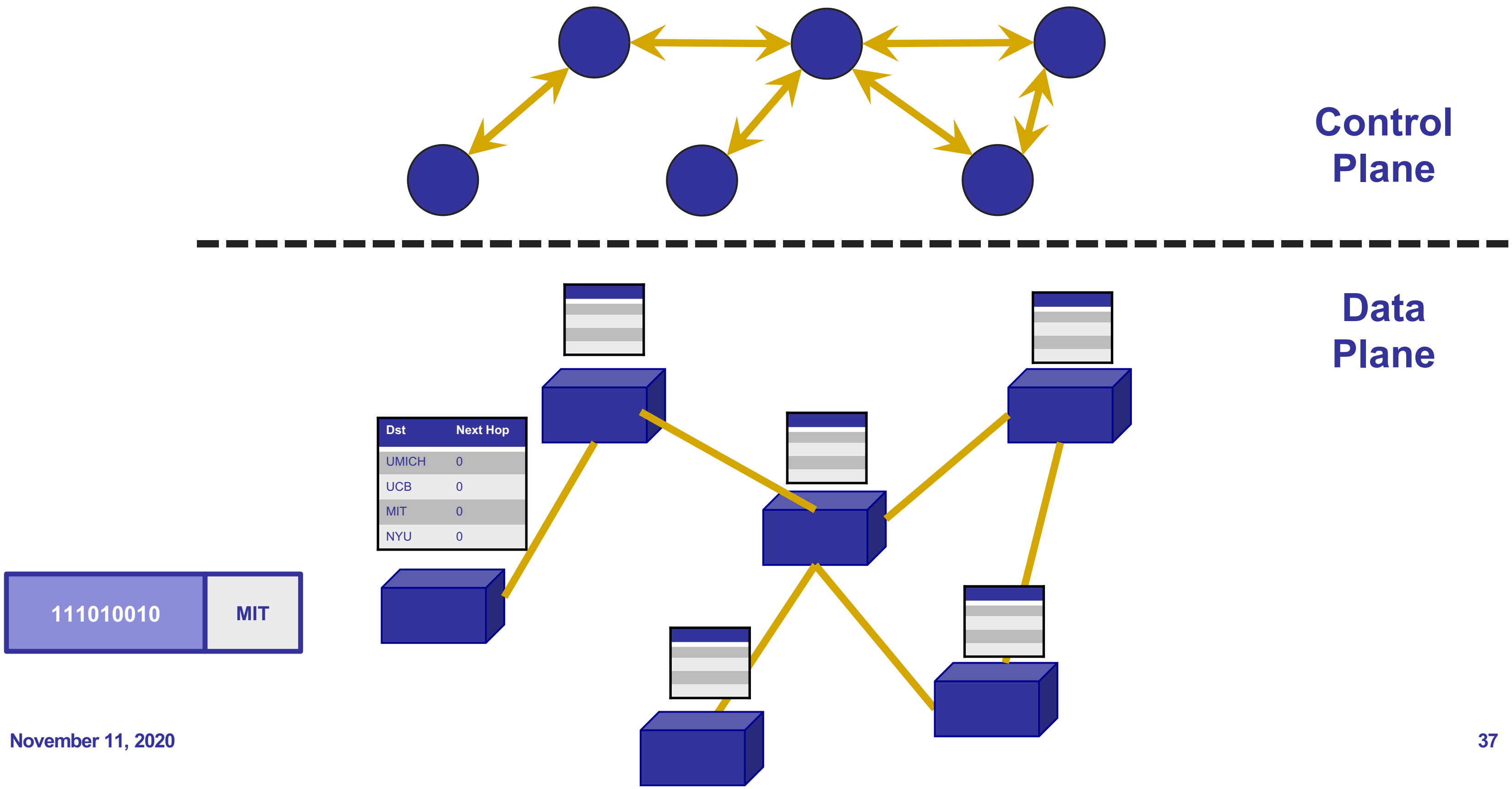
- “Modularity based on abstraction is the way things get done”
  - Barbara Liskov
- Abstractions → Interfaces → Modularity

Separate concerns with abstractions

- Be compatible with low-level hardware/software
  - Need an abstraction for general forwarding model
- Make decisions based on entire network
  - Need an abstraction for network state
- Compute configuration of each physical device
  - Need an abstraction that simplifies configuration

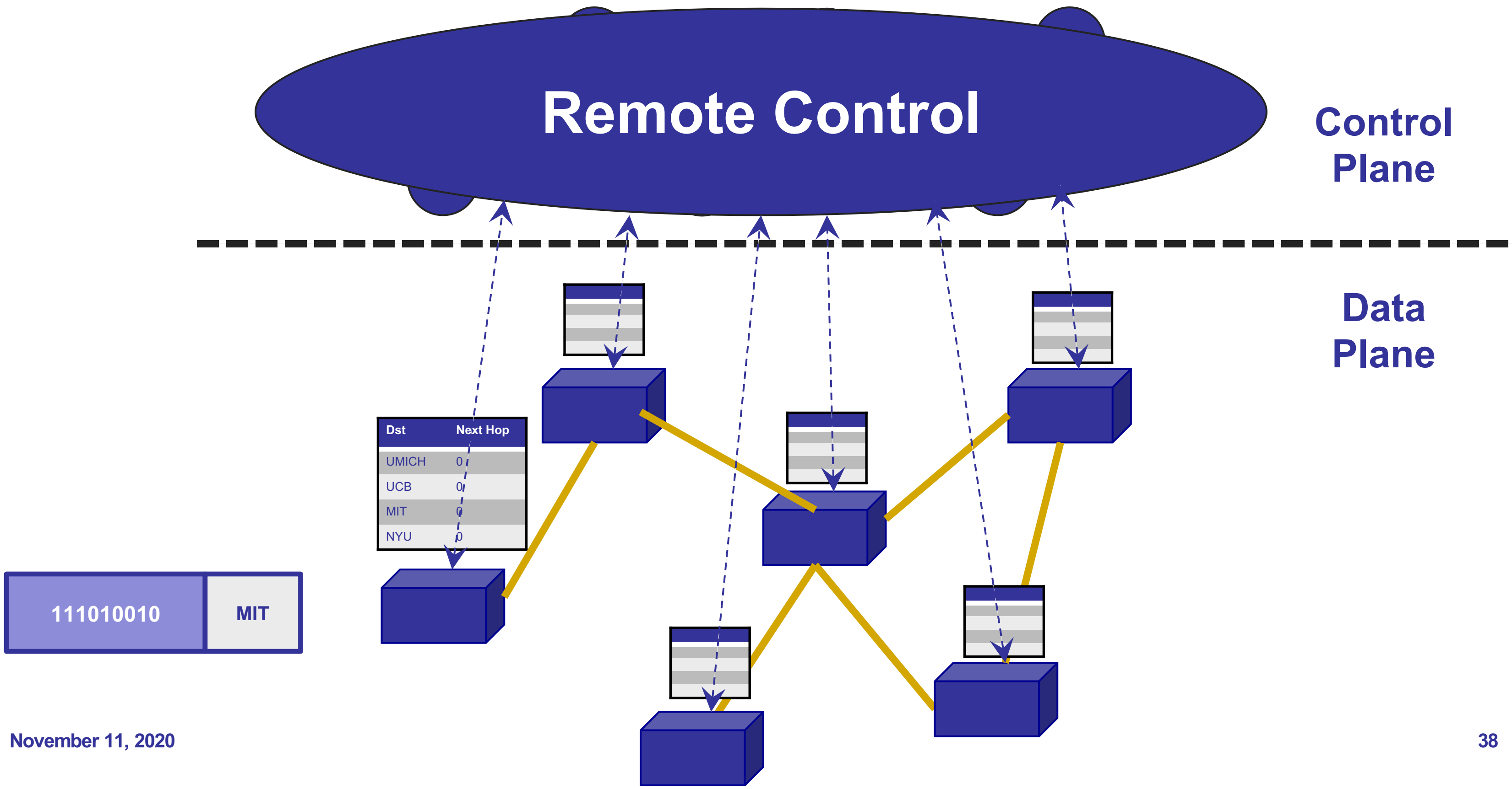
Traditional fully decentralized control plane

- Individual routing algorithm components in every router interact in the control plane



Logically centralized control plane

- A distinct (typically remote) controller interacts with local control agents (CAs)



SDN: Many challenges remain

- Hardening the control plane: dependable, reliable, performance-scalable, secure distributed system
  - Robustness to failures: leverage strong theory of reliable distributed system for control plane
  - Security: “baked in” from day one?
- Networks, protocols meeting mission-specific requirements
  - E.g., real-time, ultra-reliable, ultra-secure
- Internet-scaling

OpenFlow data plane abstraction

- Flow is defined by header fields
- Generalized forwarding: simple packet-handling rules
  - Pattern: match values in packet header fields
  - Actions: for matched packet: drop, forward, modify, matched packet or send matched packet to controller
  - Priority: disambiguate overlapping patterns
  - Counters: #bytes and #packets

1. src=1.2.*.*, dest=3.4.5.* → drop
2. src = *.*.*.*, dest=3.4.*.* → forward(2)
3. src=10.1.2.3, dest=*.*.*.* → send to controller

Fixed-function data plane

- Traditional switches are fixed-function
  - They can do whatever they can do at birth, but they cannot change!
  - Bottom-up design
- Even OpenFlow was designed to be a fixed protocol
  - With a fixed table format
  - Capable of doing limited things

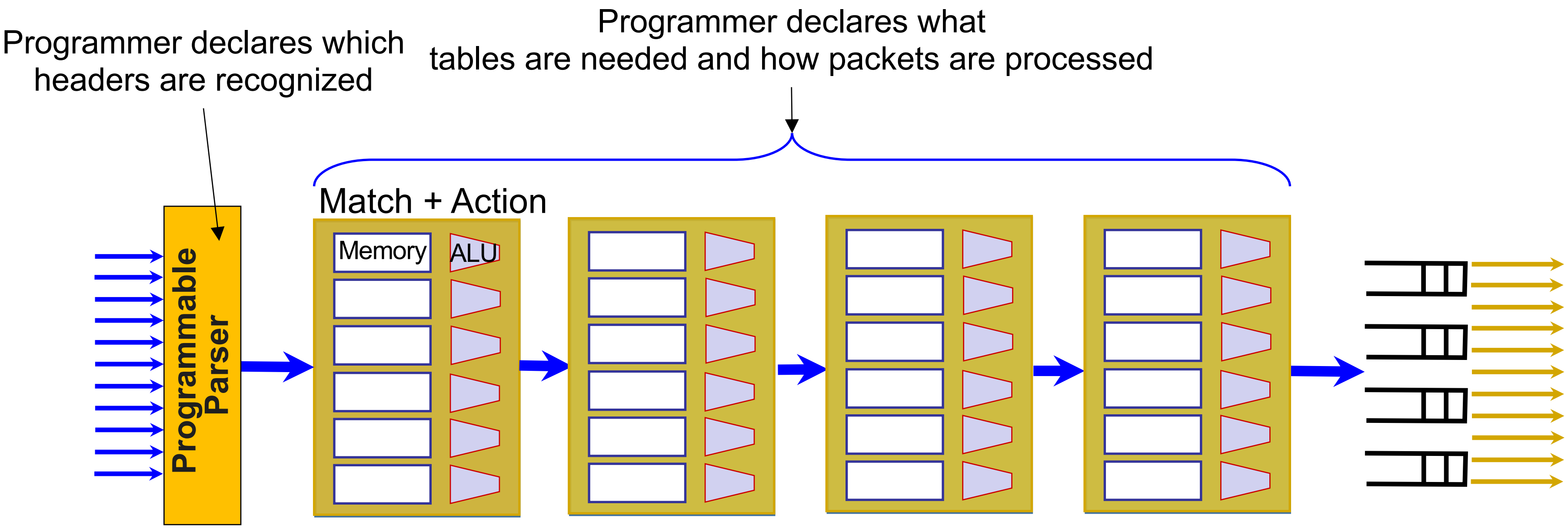
Programmable data plane

- What if we could tell switches exactly what we want?
  - What table to keep?
  - What rules to use?
  - What data to keep track of?
  - …

Top-down workflow

- Precisely specify using a well-defined language
- Compile it down to run on a standardized hardware (e.g., using P4)
- Run at line speed

PISA: Protocol Independent Switch Architecture



All stages are identical – makes PISA a good "compiler target"

Summary

- Intra-domain routing minimizes a cost metric
- Inter-domain routing is more complex due to policies
- Programmable networks are more flexible than fixed-function ones

- Next week: Layer 2

- Join us on Friday in welcoming Prof. Jen Rexford back to Michigan for a DLS talk