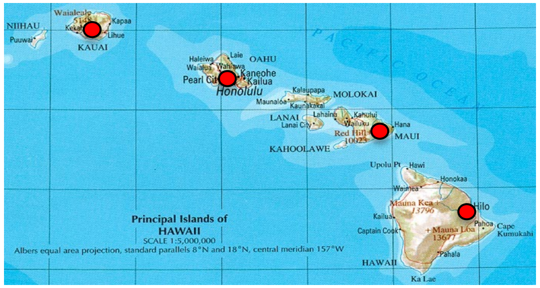


Class Meeting: Lectures 17 and 18: Wireless, SDN

COS 461: Computer Networks

Kyle Jamieson

[Various parts adapted from B. Karp, N. McKeown, J. Rexford]

Today

- Wireless Networks
  - What makes wireless networks hard?
  - ALOHA: taking turns
  - MACA: sensing other transmissions

- Programmable Networks

Wireless is increasingly prevalent

Smart Home

- Health and Fitness
- Virtual Reality
- UAVs
- Internet of Things

Vehicular Networks

Cellular Networks

Wireless Links

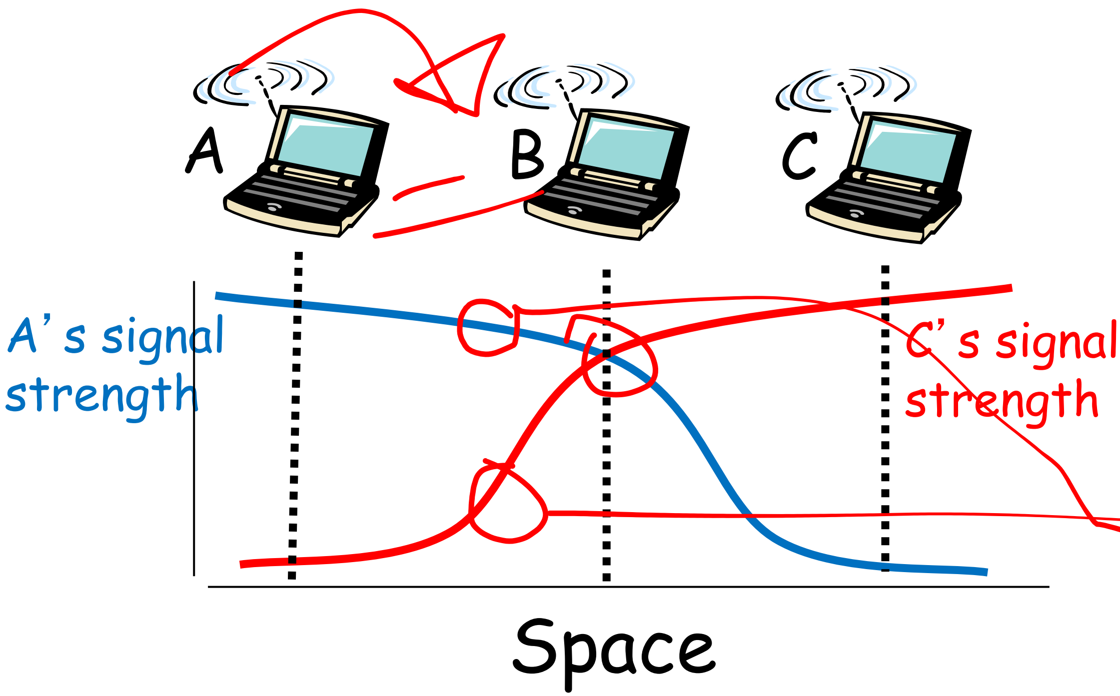
- Interference / bit errors
  - More sources of corruption vs wired
- Multipath propagation
  - Signal does not travel in a straight line
- (Often) a ***broadcast*** medium
  - All traffic to everyone nearby
- Power trade-offs
  - Important for mobile, battery-powered devices

Dealing With Bit Errors

- Wireless vs. wired links
  - Wired: most loss is due to queuing congestion
  - Wireless: higher, time-varying bit-error rate
- Dealing with high bit-error rates
  - Sender could increase transmission power
    - More interference with other senders
  - Stronger error detection and recovery
    - More powerful error detection/correction codes
    - Link-layer retransmission of corrupted frames

Wireless Broadcast and Interference:

Interference matters <u>at the receiver</u>



Signal/Interference ratio:

$$\frac{\text{A's signal strength @ B}}{\text{C's signal strength @ B}}$$

A and B hear each other... B and C hear each other

But, A and C do not

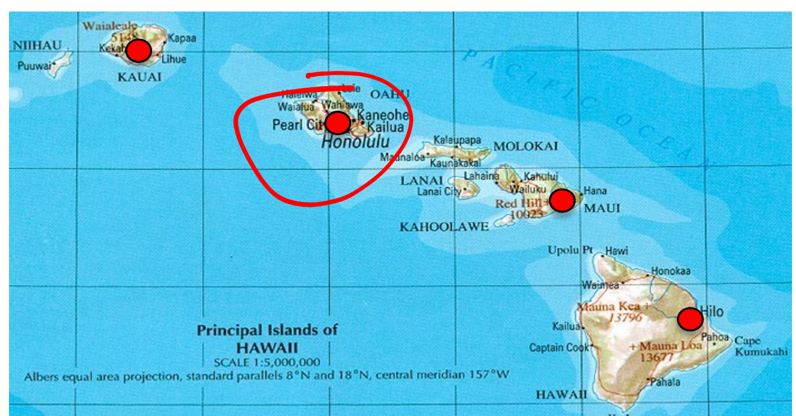
So, A and C are unaware of their interference <u>at B</u>

Wireless LANs: a Timeline

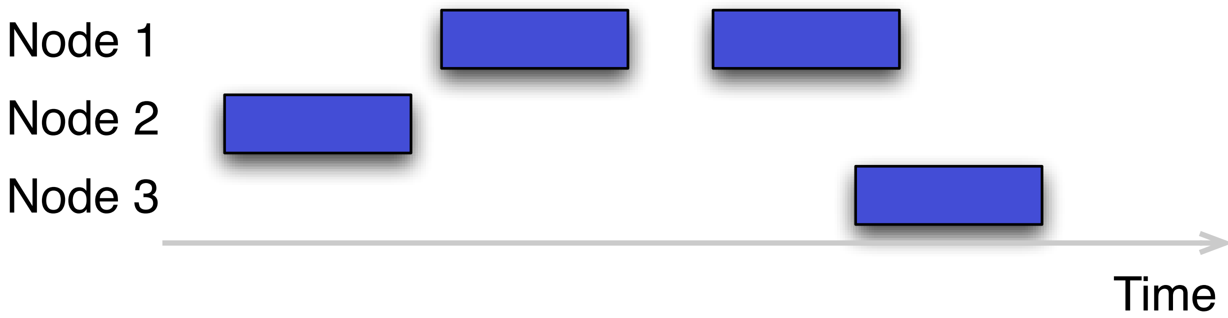
Packet radio	Wireless LAN	Wired LAN	
ALOHAnet			*1960s*
Amateur packet radio		Ethernet	*1970s*

ALOHAnet: Context

- Norm Abramson, 1970 at the University of Hawaii
  - Seven campuses, on four islands
  - Wanted to **connect** campus terminals and mainframe
  - Telephone costs high, so built a **packet radio network**



An *Unslotted ALOHA* Network



- **Suppose:** Chance new packet **in time** Δt: $\lambda \times \Delta t$
  - N senders in total, send frames of time duration 1

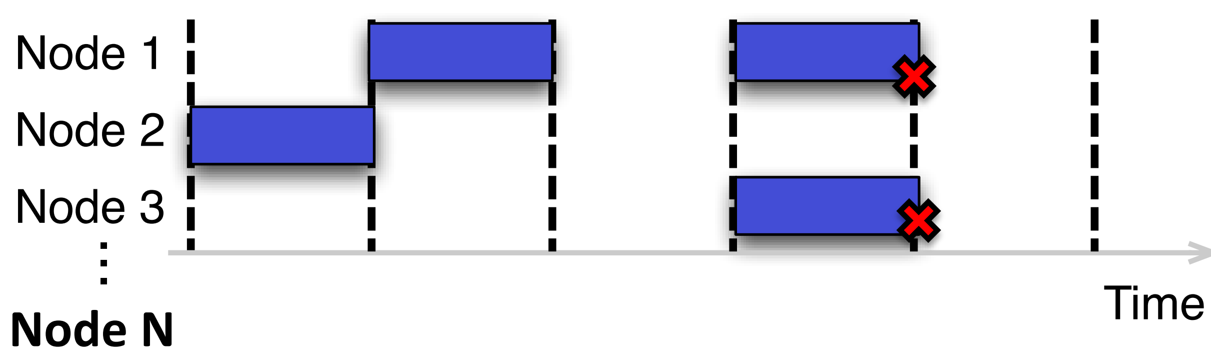
- **Then:** λ frames/sec *aggregate rate* from all N senders
  - *Individual rate* λ/N for **each sender**

- **Collision and loss of data if the frames overlap (even a bit!)**

Medium Access Control Refinement: "Slotted ALOHA"

- **Divide time into slots** of duration 1, **synchronize** so that nodes transmit **only** in a slot
  - Each of N **nodes** transmits w/prob. p in each slot
  - So **total transmission rate λ** = $N \times p$

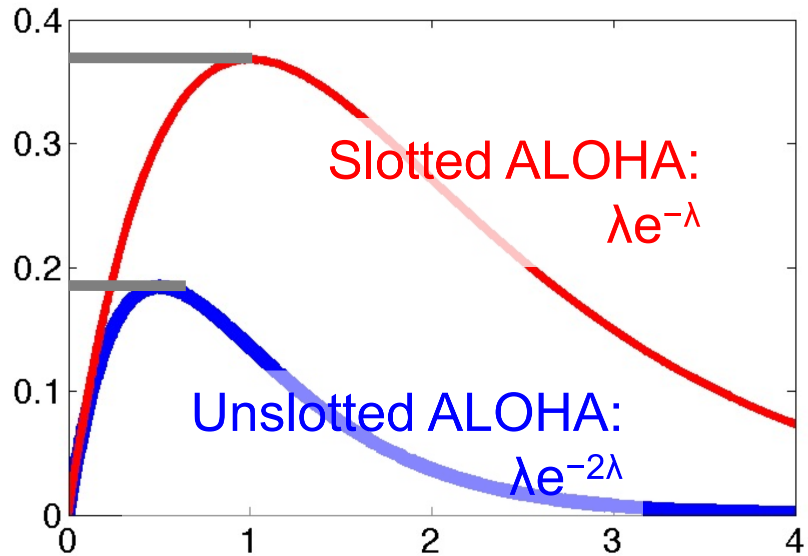
- As before, if **exactly one** transmission in slot, **can receive;** if **two or more** in slot, **no one can receive (collision)**



ALOHA Medium Access Control: Timeslots Double Throughput!

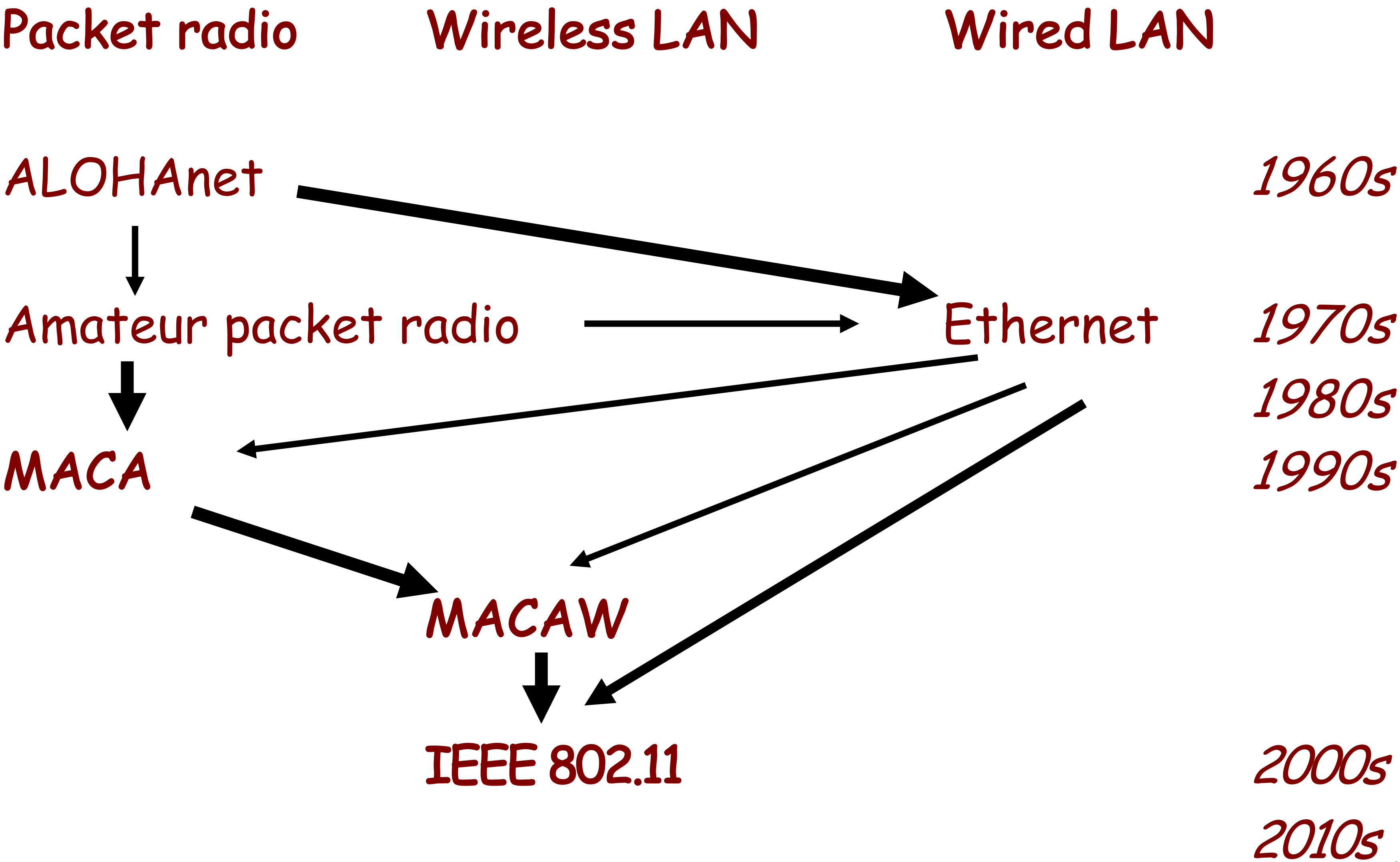
$1/e \approx 36\%$

$1/2e \approx 18\%$



Just by forcing nodes to transmit on slot boundaries, we double peak medium utilization!

Wireless LANs: Timeline

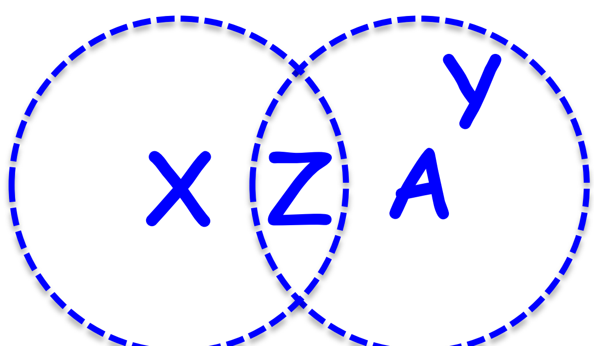


Assumptions

- Uniform, circular radio propagation
  - Fixed transmit power, all same ranges
  - Equal **interference** and **communication** ranges

Radios modeled as "conditionally connected" wires based on circular radio ranges

- Def'n: Node is connected to other node *iff other located within* circular radio range:



MACA: Goals

- Goals
  - Fairness in sharing of medium
  - Efficiency (total bandwidth achieved)
  - Reliability of data transfer at MAC layer

When Does Listen-Before-Talk *Carrier Sense* (CS) Work Well?

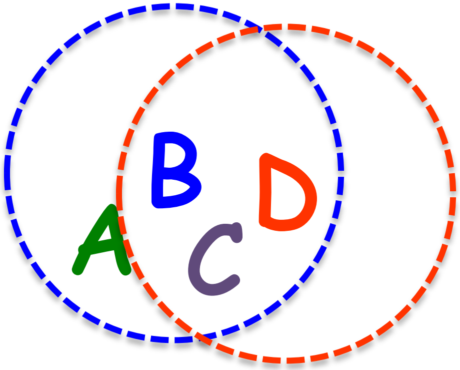
- Two pairs far away from each other
  - Neither sender carrier-senses the other

B transmits to A, while D transmits to C.

When Does CS Work Well?

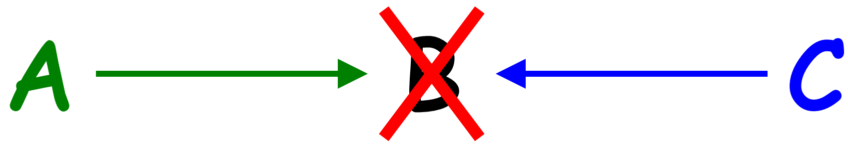
- Both transmitters can carrier sense each other

But what about cases in between these extremes?



B transmits to A, D transmits to C, taking turns.

Hidden Terminal Problem



- C can't hear A, so C will transmit while A transmits
  - Result: Collision at B

- Carrier Sense insufficient to detect all transmissions on wireless networks!

- **Key insight:** Collisions are **spatially located at receiver**

Exposed Terminal Problem

- If C transmits, does it cause a collision at A?
  - Yet C cannot transmit while B transmits to A!

- Same insight: Collisions spatially located at receiver

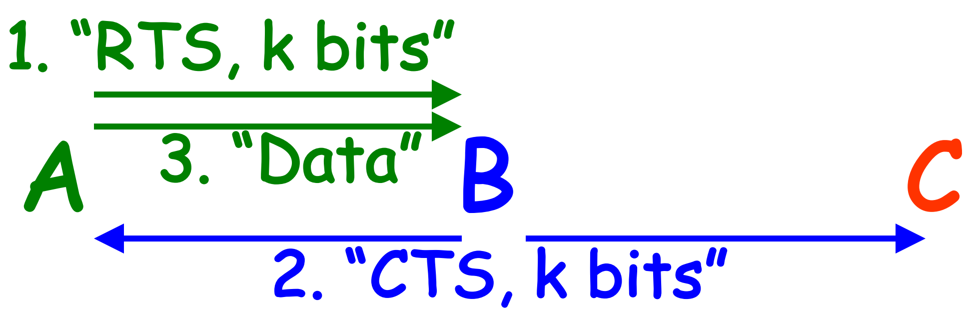
- One possibility: directional antennas rather than omnidirectional. Why does this help? Why is it hard?

MACA: Multiple Access with Collision Avoidance

- Carrier sense became adopted in packet radio
- But **distances** (cell size) remained large
- Hidden and Exposed terminals abounded
- **Simple solution:** use *receiver's* **medium state** to determine **transmitter behavior**

RTS/CTS

- Exchange of two short messages: *Request to Send* (RTS) and *Clear to Send* (CTS)
- **Algorithm**
  1. A sends an RTS (tells B to prepare)
  2. B replies an CTS (echoes message length)
  3. A sends its Data

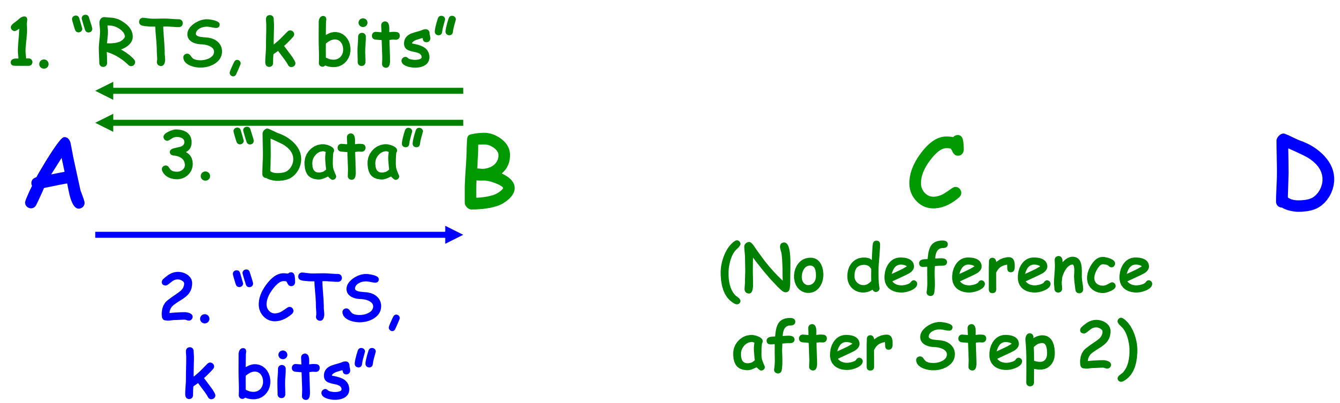


Deference to CTS

- Hear CTS → Defer for length of expected data transmission time
  - Solves hidden terminal problem

Deference to RTS, but not CS

- Hear RTS → Defer one CTS-time (*why?*)
- MACA: No carrier sense before sending!
  - Karn concluded useless because of **hidden** terminals
- So exposed terminals B, C can transmit concurrently:



Today

- Wireless Networks

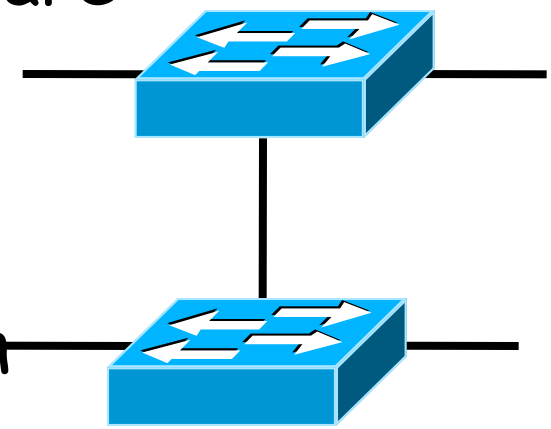
- Programmable Networks
  - Division of labor: control, data planes
  - Software-defined networking
  - Examples

The Internet: A Remarkable Story

- Tremendous success
  - From research experiment to global infrastructure
- Brilliance of under-specifying
  - Network: best-effort packet delivery
  - Hosts: arbitrary applications
- Enables innovation in applications
  - Web, P2P, VoIP, social networks, smart cars, …
- But, change is easy only at the edge… ☹

Inside the 'Net: A Different Story…

- Closed equipment
  - Software bundled with hardware
  - Vendor-specific interfaces
- Over specified
  - Slow protocol standardization
- Few people can innovate
  - Equipment vendors write the code
  - Long delays to introduce new features

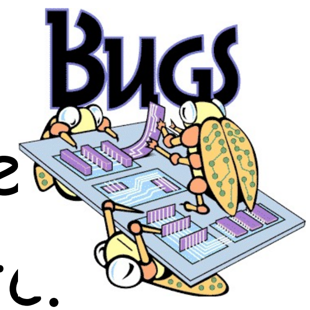


Impacts performance, security, reliability, cost…

Networks are Hard to Manage

- Operating a network is expensive
  - More than half the cost of a network
  - Yet, operator error causes most outages

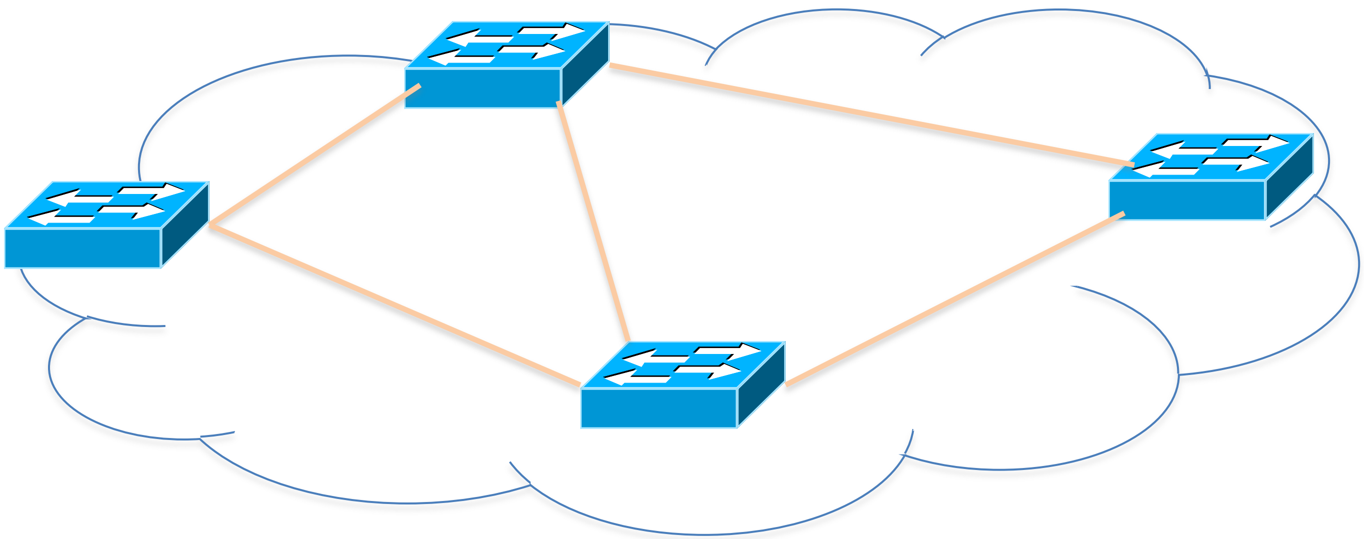
- Buggy software in the equipment
  - Routers with 20+ million lines of code
  - Cascading failures, vulnerabilities, etc.



- The network is "in the way"
  - Especially in data centers and the home

Rethinking the "Division of Labor"

Traditional Computer Networks

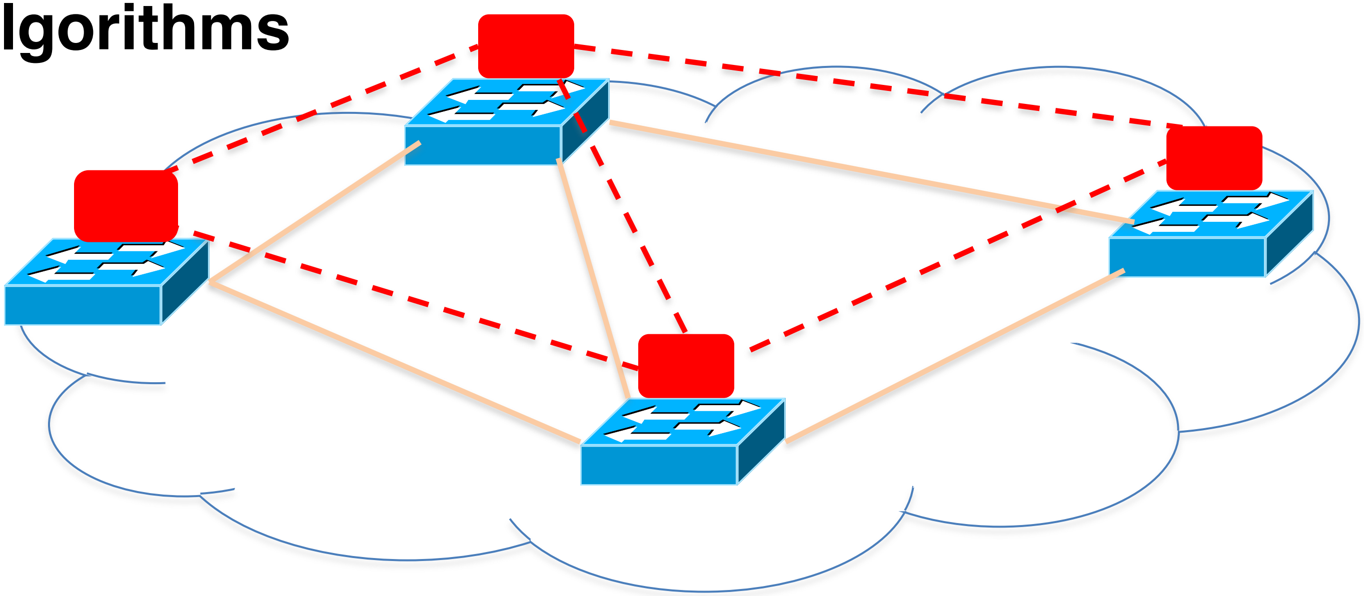


Data plane:
Packet streaming

Forward, filter, buffer, mark, rate-limit, and measure packets

Traditional Computer Networks

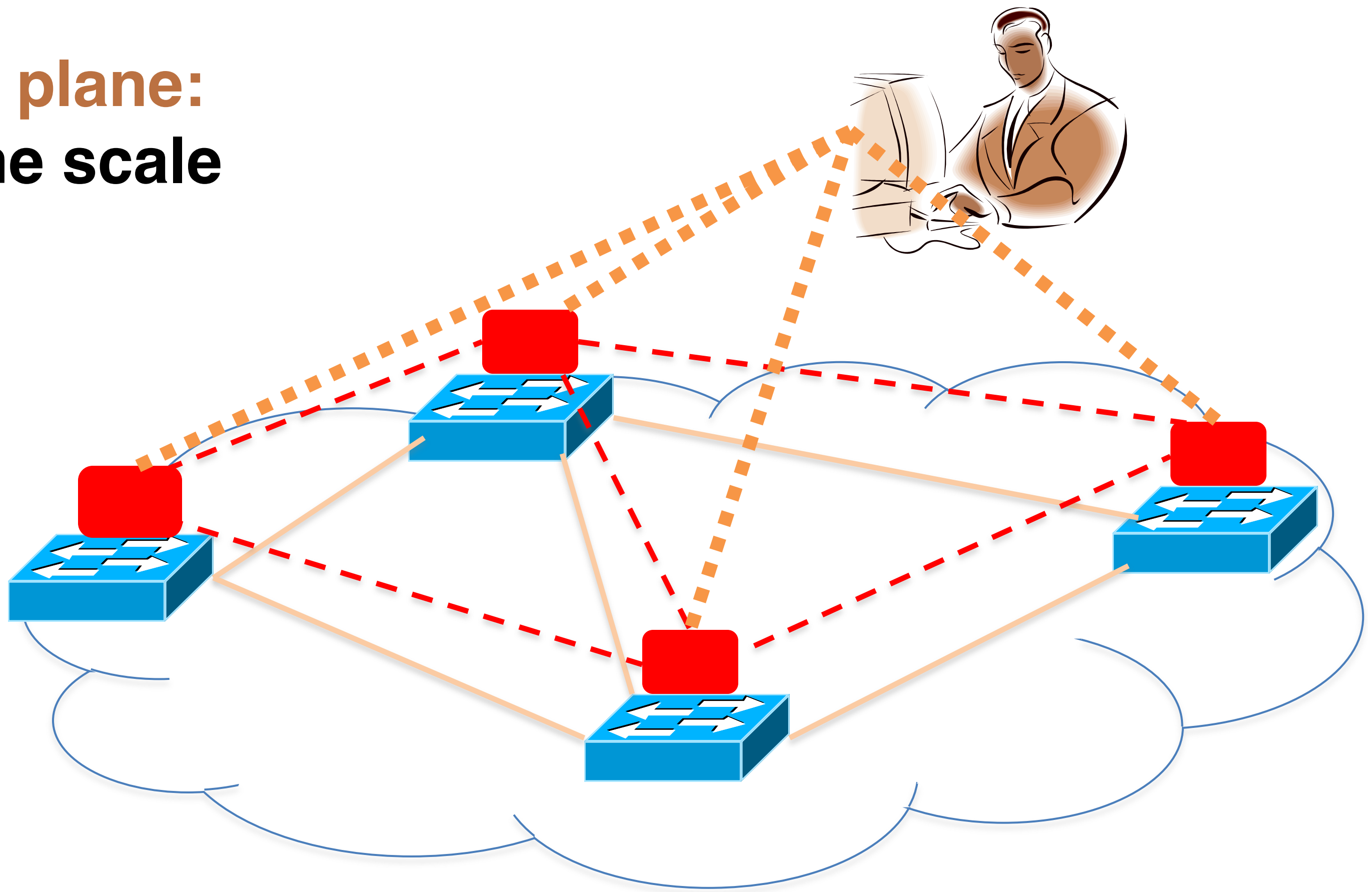
Control plane:
Distributed algorithms



Track topology changes, compute routes, install forwarding rules

Traditional Computer Networks

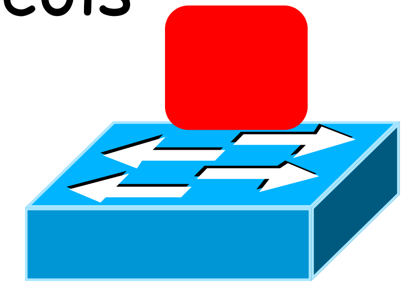
Management plane:
Human time scale



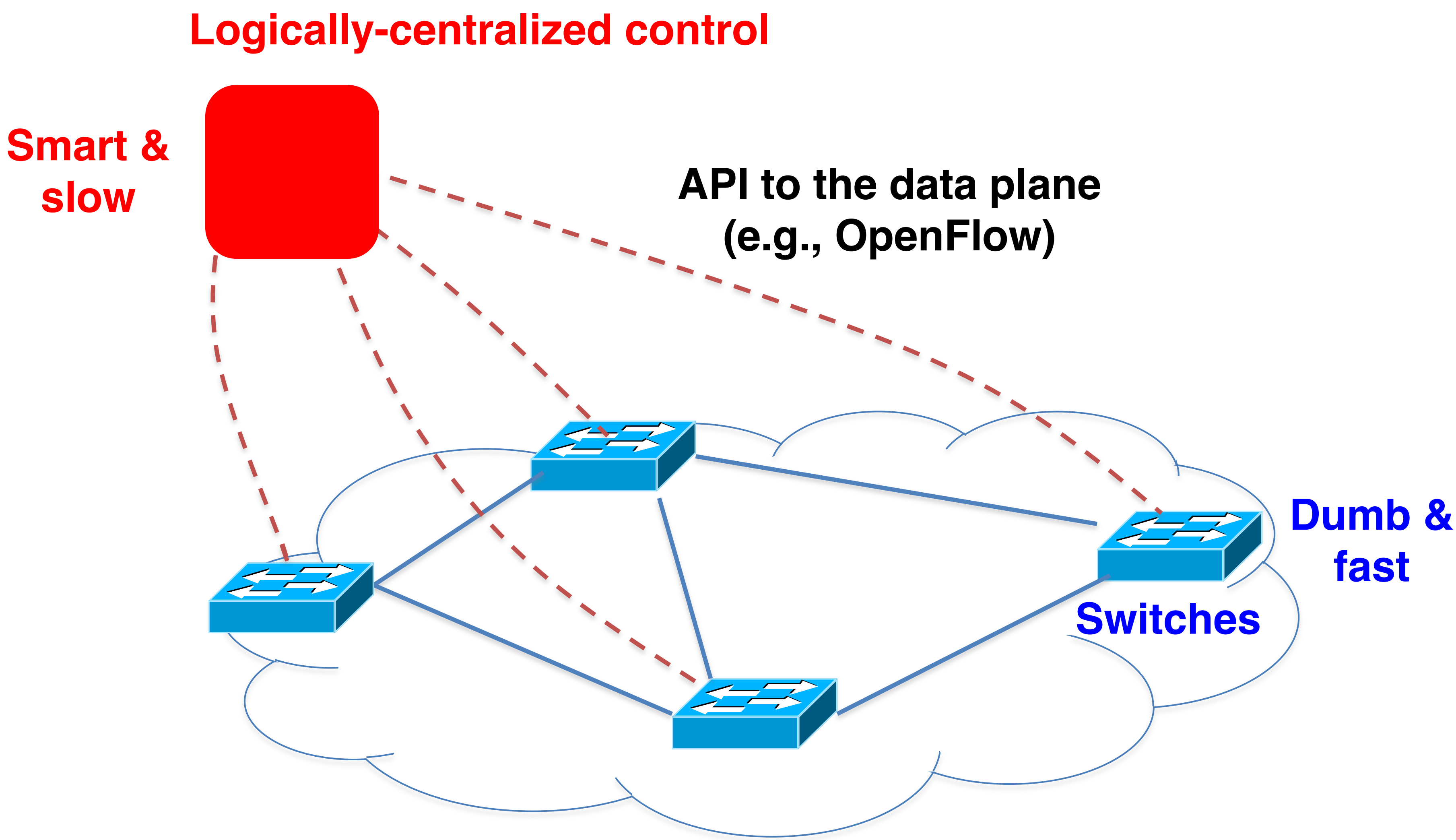
Collect measurements and configure the equipment

Remove that Control Plane!

- Simpler management
  - No need to "invert" control-plane operations
- Faster pace of innovation
  - Less dependence on vendors and standards
- Easier interoperability
  - Compatibility only in "wire" protocols
- Simpler, cheaper equipment
  - Minimal software



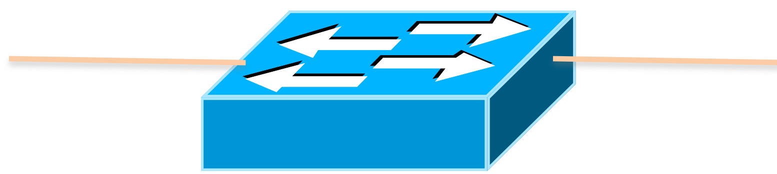
Software Defined Networking (SDN)



Data Plane: Simple Packet Handling

- Simple packet-handling rules
  - Pattern: match packet header bits
  - Actions: drop, forward, modify, send to controller
  - Priority: disambiguate overlapping patterns
  - Counters: #bytes and #packets

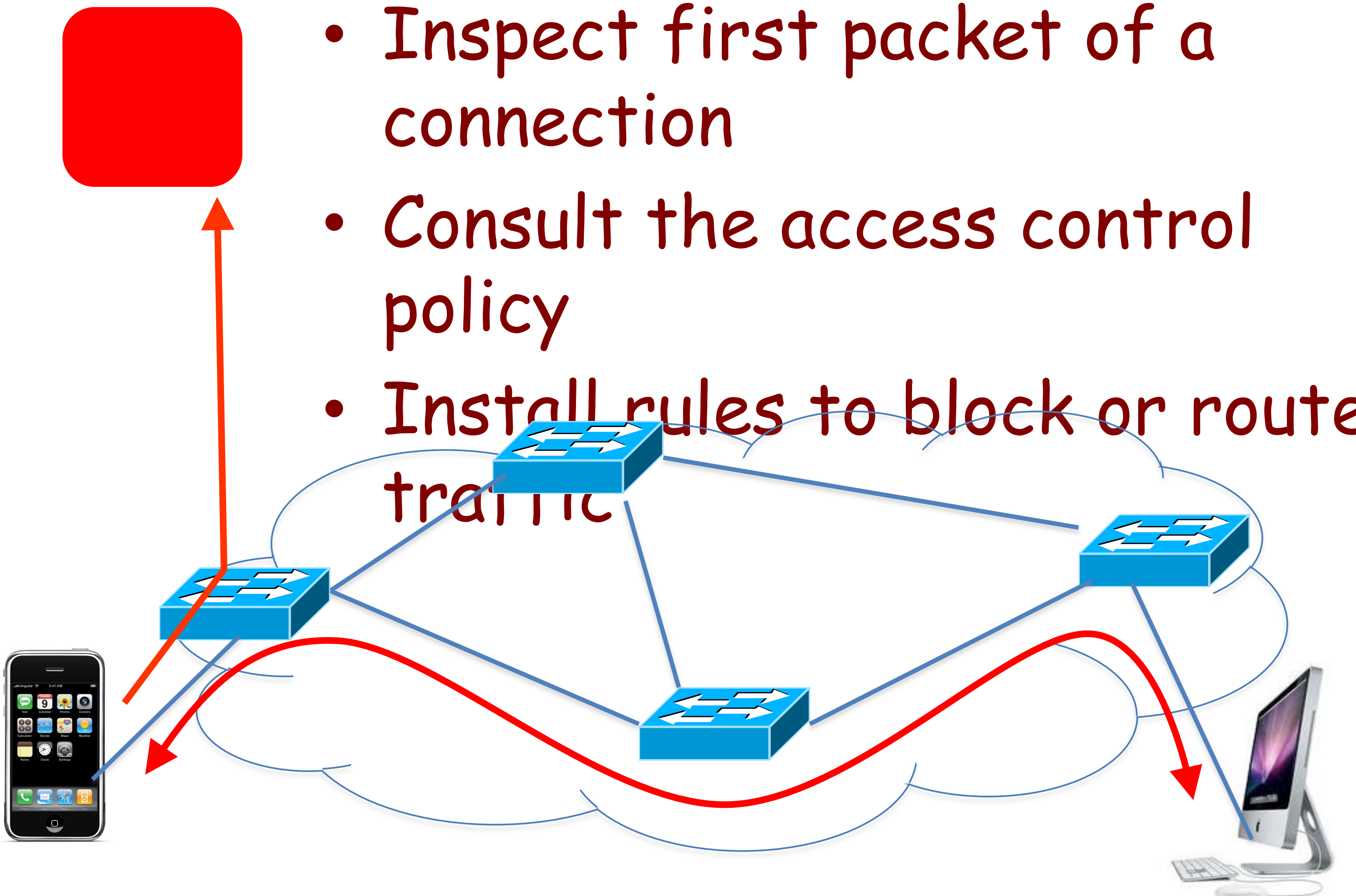
OpenFlow



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**

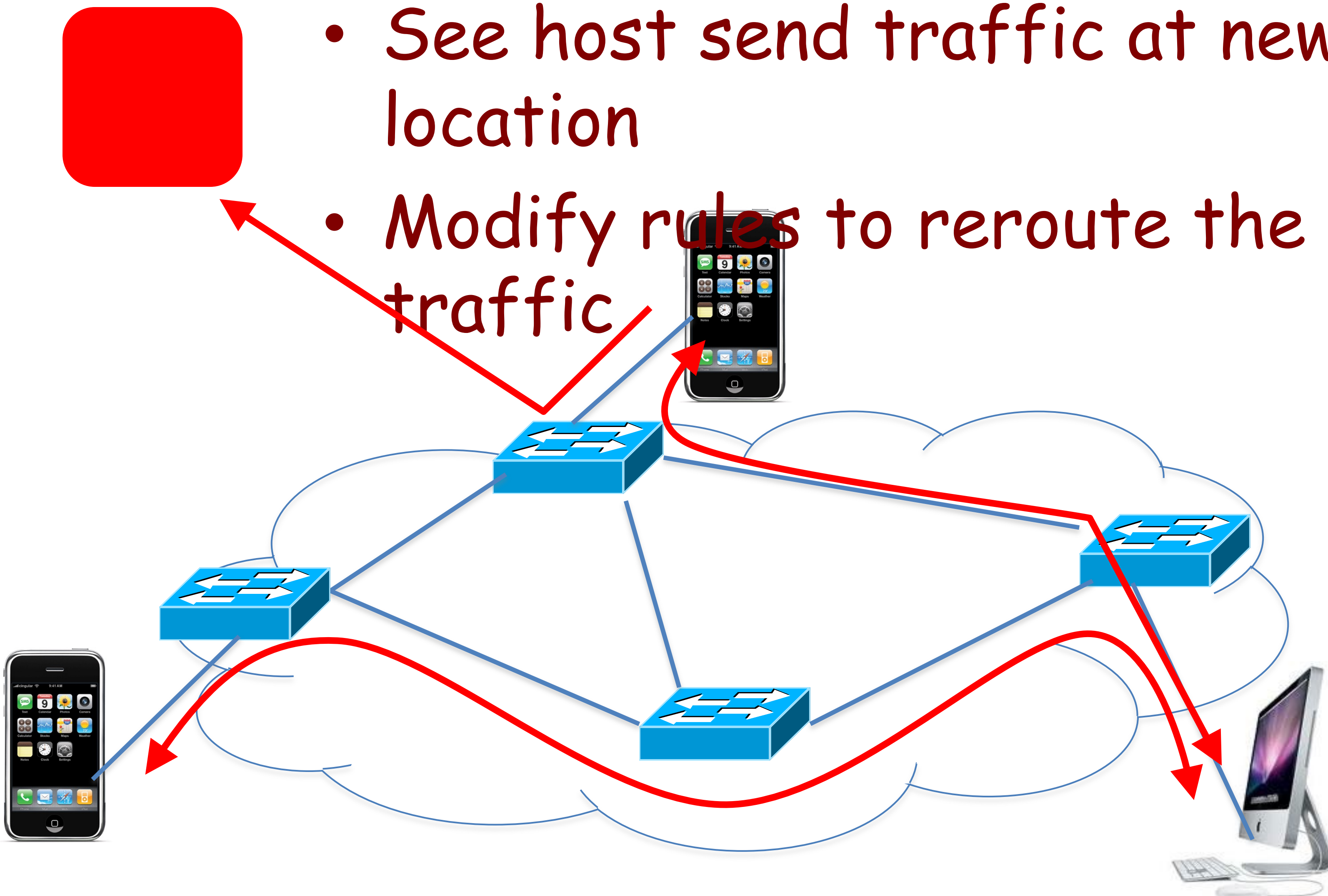
E.g.: Dynamic Access Control

- Inspect first packet of a connection
- Consult the access control policy
- Install rules to block or route traffic



E.g.: Seamless Mobility/Migration

- See host send traffic at new location
- Modify rules to reroute the traffic



Summary

- Wireless networks: de facto means of accessing the Internet
  - Evolution from ALOHAnet, Ethernet, MACA, toward IEEE 802.11 Wi-Fi
- Software-Defined Networks: new ways of managing networks
  - New API, OpenFlow, enables new applications