E6998-02: Internet Routing

Lecture 1
Introduction

John Ioannidis
AT&T Labs – Research
ji+ir@cs.columbia.edu

Announcements

Instructor: John Ioannidis, pronounced *jay-eye*

Email: <ji+ir@cs.columbia.edu>

- Mail to anything else will not be answered.

Class web page: http://www.cs.columbia.edu/~ji/F03/

- Check frequently!
- Slides will be available there.
- As will additional reading material (papers, RFCs, source code, man pages, etc.).

Class BBoard: coms6998-002-023@columbia.edu (to post), or https://www1.columbia.edu/sec/bboard/023/coms6998-002/

Office hours: MW 15:00-16:00 or by appointment.

TA(s): Angelos Stavrou, <angel@cs.columbia.edu>

TA office hours: TR 13:00-14:00.

The Necessary Evil

- Four homeworks: 10% each
- Midterm: 25%
- Final: 35%
- Class participation: at my discretion.

- Project may substitute for final with prior instructor approval.

Homeworks

- 1-2 weeks each.
- Look at the web site for announcements.
- Don’t start the night before, you won’t finish it.
- Involve some coding, some reading, some writing.
- Each counts for 10% of your grade.
- Submitted over email.
  - ji+hw*n*@cs.columbia.edu, *n*=1…4
- Only plain text or .pdf files accepted.
  - MS Word *etc*. strictly forbidden.
- Individual submissions.
  - Cheaters will be given an F in the course and sent to the Dean for further processing.

Exams

- Usual rules apply.
- Closed book.

Project (optional)

- Sizeable project to demonstrate your mastery of the material:
  - Coding.
  - Simulation.
  - Survey or original research paper.

- Individual or in groups of up to 3 people.
  - Past experience shows that individual projects are impossibly hard.
  - Group gets grade; members decide how to apportion the grade.
  - More work than just taking the final, but worth it.
  - Ask me for ideas.

Class Participation

- Fun for me.
- Shows you are paying attention.
- May affect your grade.
- No mobile phones or other interruptions.
- Ask questions early and often:
  - I may be assuming that you know something when you don't (frequent).
  - I may not be explaining something well enough (rare!).
- If you are having trouble with the material, talk to me or the TA. **Don't wait until it's too late!**
- "Class Participation" also means participating in bboard discussions and coming to office hours!

Prerequisites

- Formal:
  - W4118 (Operating Systems) and W4119 (Networks).
- Essential:
  - Fluency in C. C++ and Java recommended.
  - Sockets programming.
  - TCP/IP operations.
  - Have heard of the concept of routing.

Course Outline

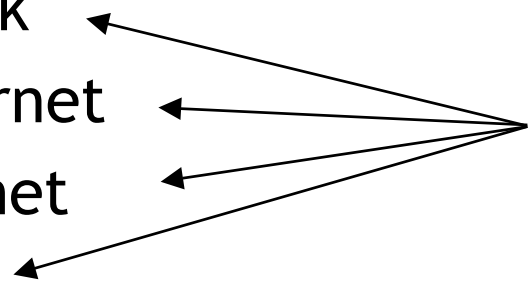
- Intro (1)
- Routing and Addressing architectures (1)
- Unix routing implementation (2)
- Bridging (1)
- Distance-Vector routing (2)
- Link-State routing (1)
- OSPF (3)
- Interdomain routing (1)
- BGP (6)
- Overlay networks (1)
- Multicast (1)
- Operations and traffic engineering (2)
- Guest lectures (1-2)
- Advanced topics (if we have time)

Networking in a Nutshell

- *Nodes* (hosts and routers) connected by *links*.
- Each node has an **address**.
  - Unique (usually).
  - Used to find node in network.
- Internet is a **packet-switched network**.
- Based on IP, a **best-effort, connectionless** protocol.
- "The Network delivers bits" (and does little else).
  - No guaranteed delivery.
  - No guaranteed in-order delivery.
  - No guaranteed correct delivery.
- Additional functionality implemented at the end nodes.
- Ancillary protocols needed to make it work.

No Such Thing as Presentation Layer!

- Forget what you learned about the 7-layer model!
- Layering as a conceptual tool.
  - Lots of "layer violations" in practice.
  - Don't be fundamentalist!
- Several layers:
  - Application
  - Transport
  - Network
    - Internet
    - Subnet
  - Link
  - Physical



We'll be concentrating on these

PDUs, PSUs, and other TLAs

- Protocol Data Units/Protocol Service Units.
- ISOisms, we don't like them – we don't use them.

- Link and below: **frame** (ethernet frame, PPP frame).
- Network: **packet** (IP packet).
- Transport:
  - **packet** (UDP packet, ICMP packet).
  - **segment** (TCP segment).
  - Some implication about block-oriented vs. stream-oriented abstractions.
- Above transport: wrong course.

- **Payload**: useful stuff carried inside a frame/packet/whatever.
- **Header**: what's not the payload!

Forwarding, Routing, Switching, Bridging

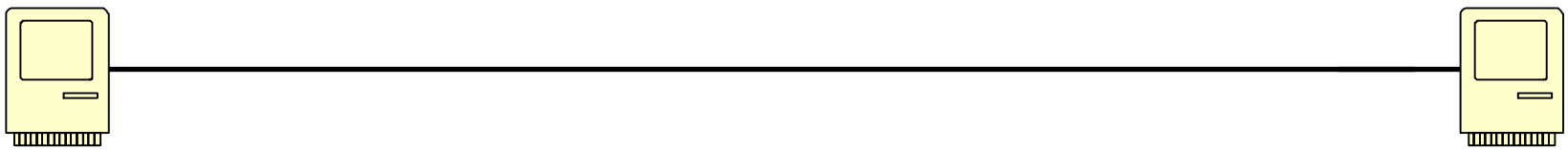
- Node has packet to send.
  - If it is for that node, send it up to higher-layer protocols.
  - Else, figure out which network interface to send it out on and/or what the next hop is.

- **Forwarding**: the problem of deciding which interface to send it out on.
  - Code/hardware on nodes (hosts/routers) does that.
- **Routing**: the problem of figuring out which interface/next hop is appropriate.
  - Combination of **routing protocols** and code on nodes.
- **Switching**: (usually) forwarding operations at <= Layer 2.
  - Also used to imply that state is kept.
- **Bridging**: Layer-2 routing.

Forwarding in a Nutshell

- Node has packet to send.
- Figure out
  - which network interface to send it out on.
  - who to send it to ("next hop").

- Two machines on the same physical link:

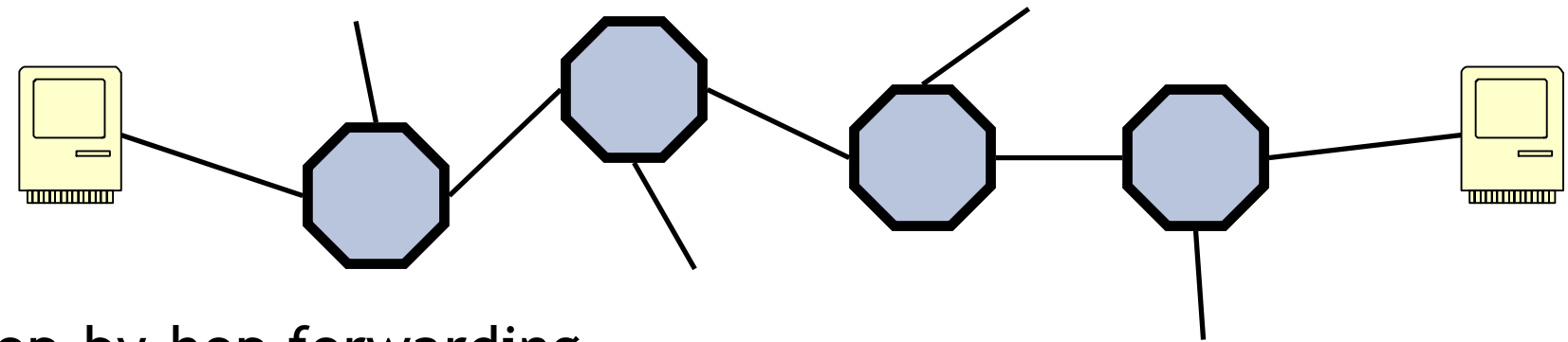


  - Only one interface.
  - Only one possible next hop.

Forwarding in a Nutshell

- Node has packet to send.
- Figure out
  - which network interface to send it out on.
  - who to send it to ("next hop").

- Two machines NOT on the same physical link:



- Hop-by-hop forwarding.
  - May require setup and hard state (circuit switching).

Forwarding Can Happen at Any Layer

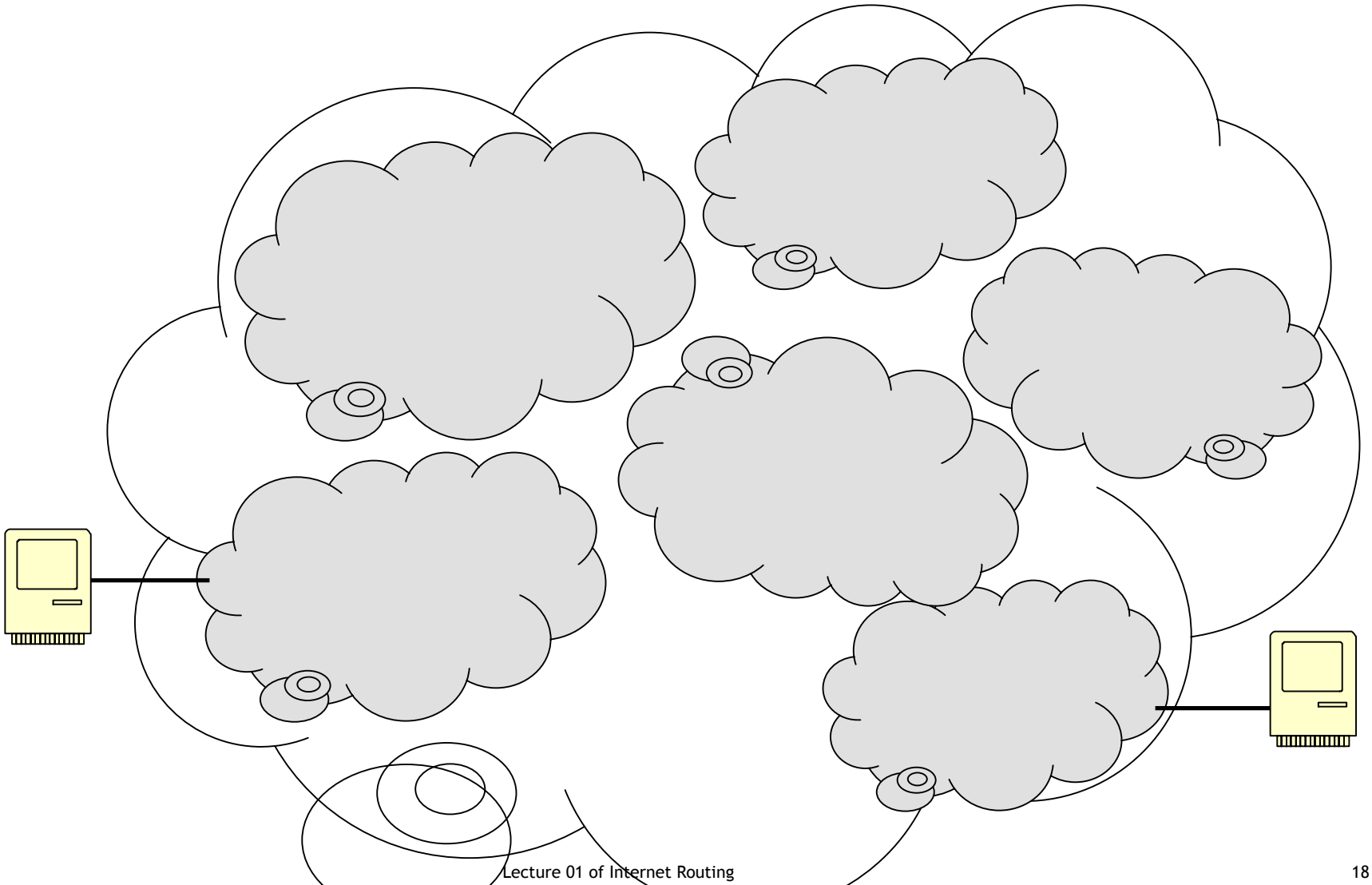
- Physical layer: connect wires together!
  - Does not scale at all.
- Link layer: bridging.
  - Independent of network protocol.
  - Can be kludged to work with similar link protocols.
  - Scales to a few hundred nodes.
- Network layer: routing.
  - Independent of link protocols.
  - Has scaled to tens of millions of nodes.
- Higher layers.
  - Application-specific.
  - CDNs etc.

Internet Routing in a Nutshell

The Internet…

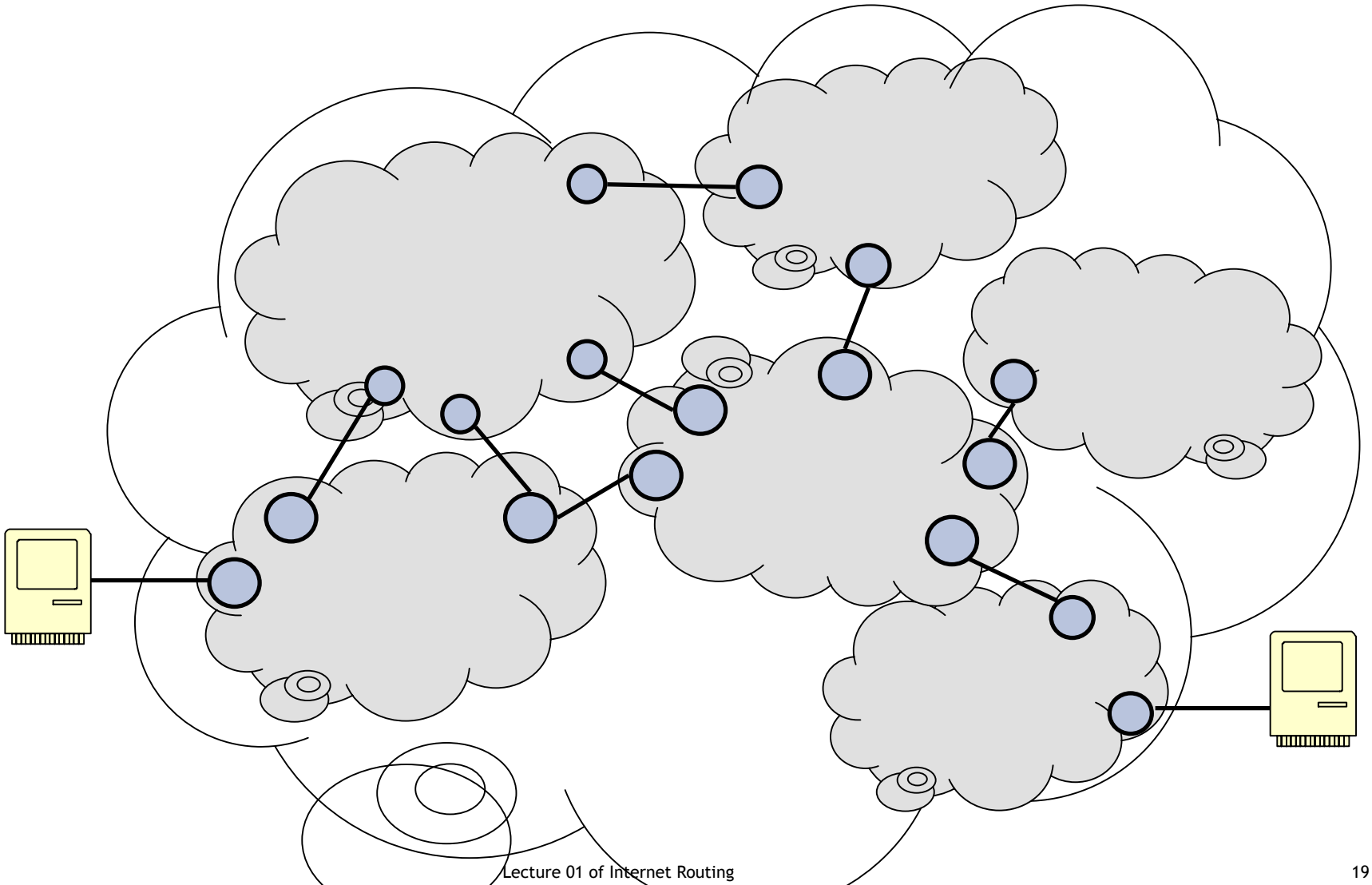
Internet Routing in a Nutshell

…consists of many individual networks…



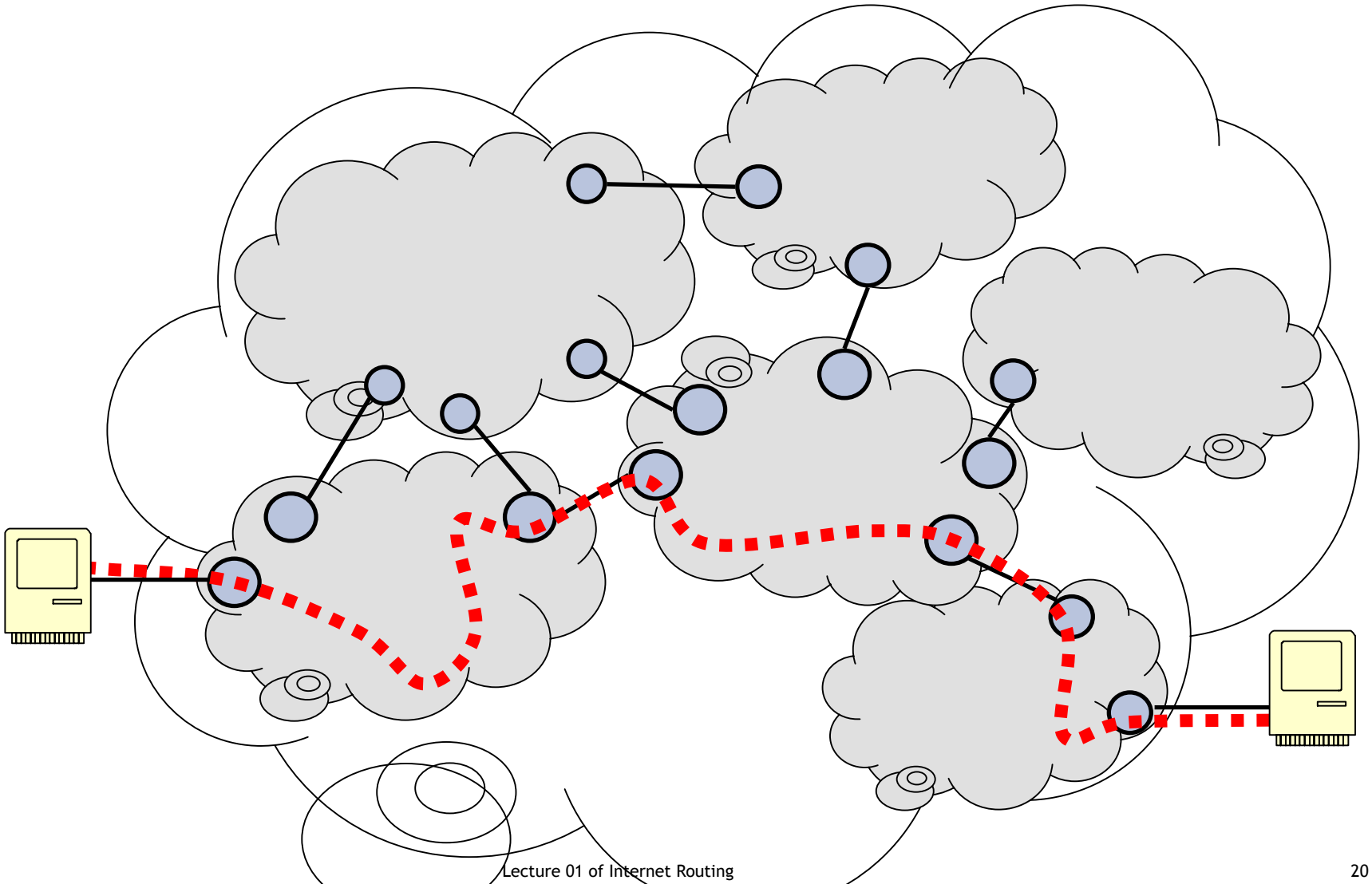
Internet Routing in a Nutshell

…linked at Border Routers…



Internet Routing in a Nutshell

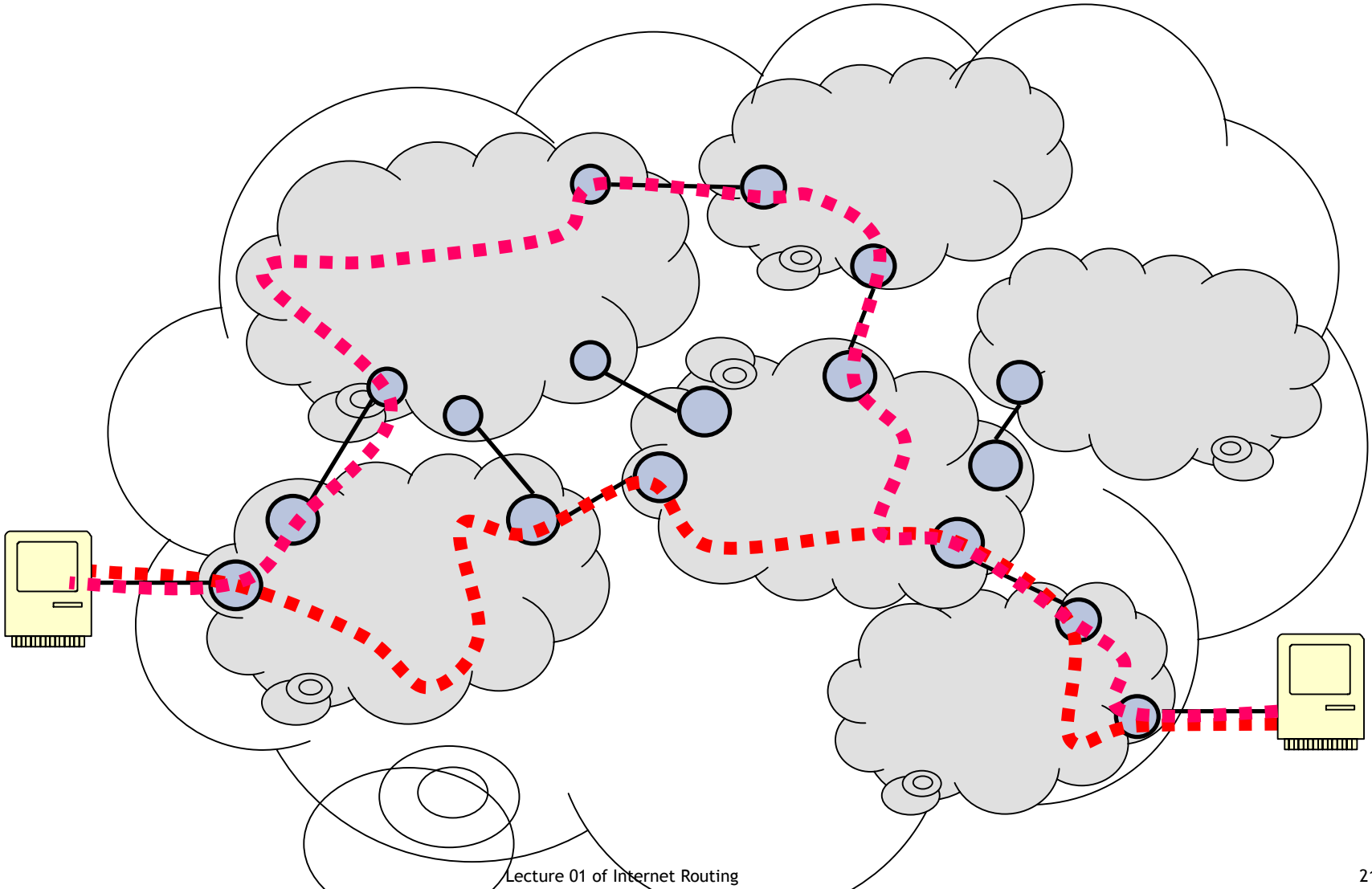
Interdomain routing protocols the coarse flow of packets.



Internet Routing in a Nutshell

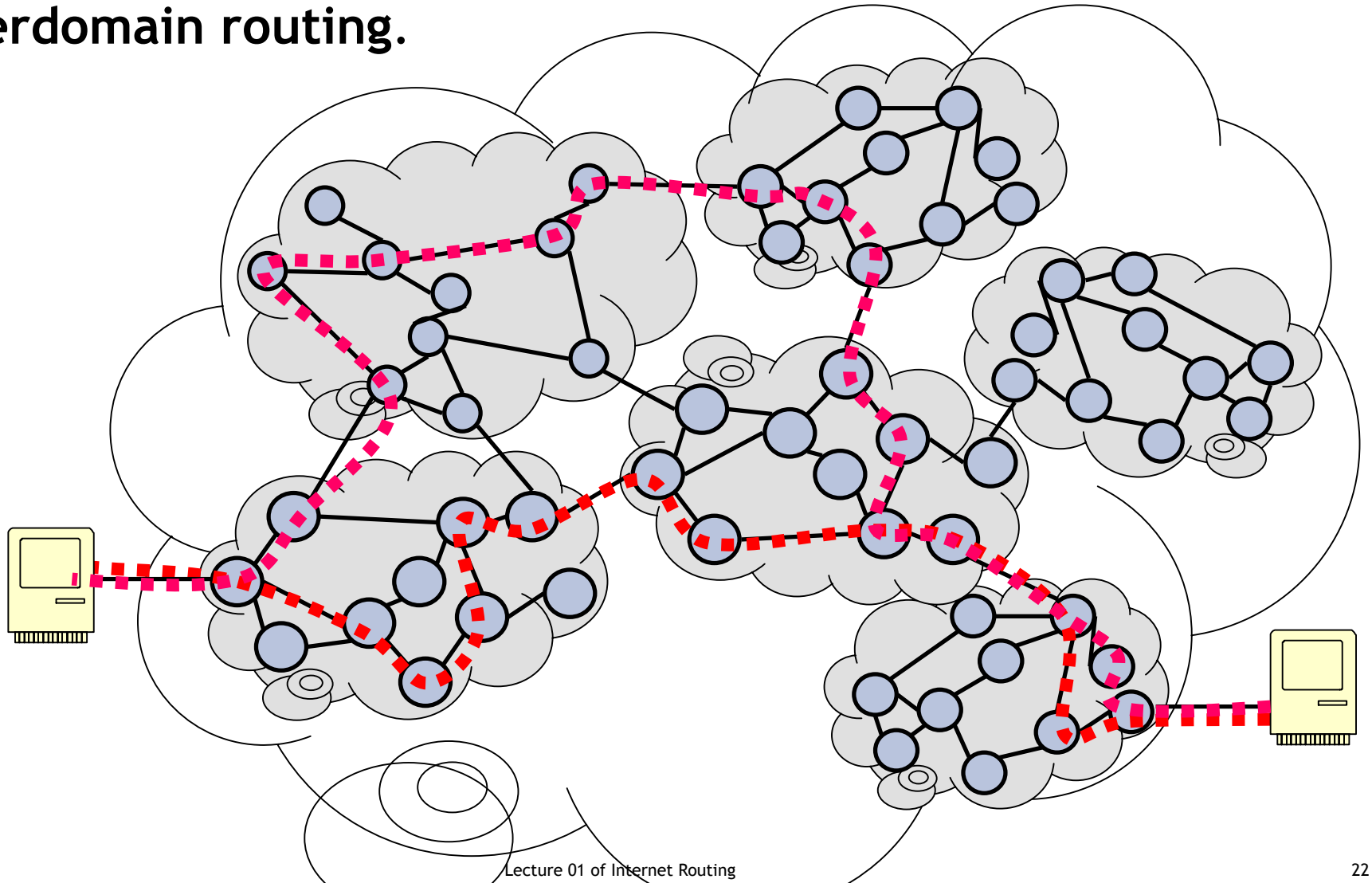
Interdomain routing protocols the coarse flow of packets.

Paths are not necessarily symmetric.



Internet Routing in a Nutshell

- Each network is responsible for moving packets inside it.
- **Intradomain routing** is (mostly) independent from **Interdomain routing.**



Routing can happen at all layers

- Physical: move connectors around!
- Link: "bridging"
- Network: "routing"
- Transport: redirectors (avoid).
- Application: ALG (definitely wrong course).

- We shall briefly examine bridging.
- We shall mostly learn about network-layer routing.

Types of Links/Networks

- **Point-to-point**
- **Point-to-multipoint**
- **Broadcast**
- **Non-broadcast Multiple Access (NBMA)**

Point-to-Point

- Two nodes connected by a real or virtual “wire”.
  - Copper wire, fiber strand, microwave link.
  - Dialup, ATM or FR PVC (permanent virtual circuit)
- No need for link-layer addresses.

- WANs are made of point-to-point links.

Broadcast

- Or **Broadcast Multiple Access.**
- Many nodes connected to a shared (physical) medium.
- All nodes on "local" network hear each other's transmissions.
- Frame carries link-layer address of destination.
  - Destination "hears" frame, picks it up.
- Ethernet, token ring, wireless, … (IEEE 802.*)

- Mostly a LAN technology.

Non-Broadcast Multiple Access (NBMA)

- Virtual LAN:
  - Made of Point-to-point links.
  - Behaves like a Multiple Access network.
- Nodes do not hear each other's transmissions.
  - Frame still carries link-layer address of destination.
  - Switching fabric gets packet to destination.
- One L3 technology tunneled over another L3 technology.
- ATM, Frame Relay, X.25, IP-over-IP, …

- Mostly a WAN technology.

- ATM made an attempt to be used as a LAN ("LANE").

Before the next class

- Read RFC791 and RFC1958.

- Read Perlman Chapters 1&2.