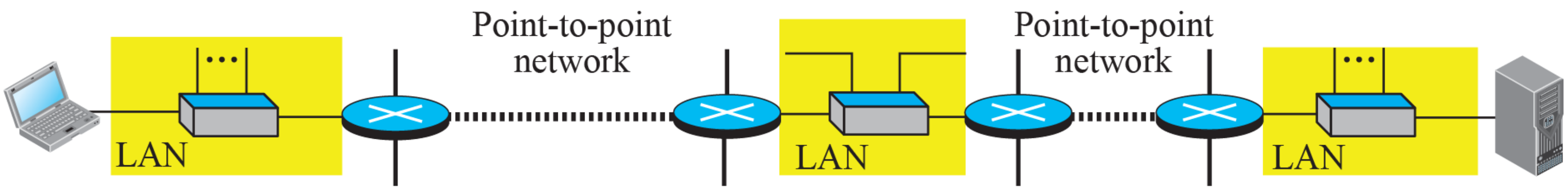
Chapter 9

Introduction to Data-Link Layer

Nodes and Links

- Communication at the data-link layer is node-to-node.
- A data unit from one point in the Internet needs to pass through many networks (LANs and WANs) to reach another point.
- Theses LANs and WANs are connected by routers.
- It is customary to refer to the two end hosts and the routers as nodes and the networks in between as links.
- Figure in next slide is a simple representation of links and nodes when the path of the data unit is only six nodes.

Figure 9.2: Nodes and Links



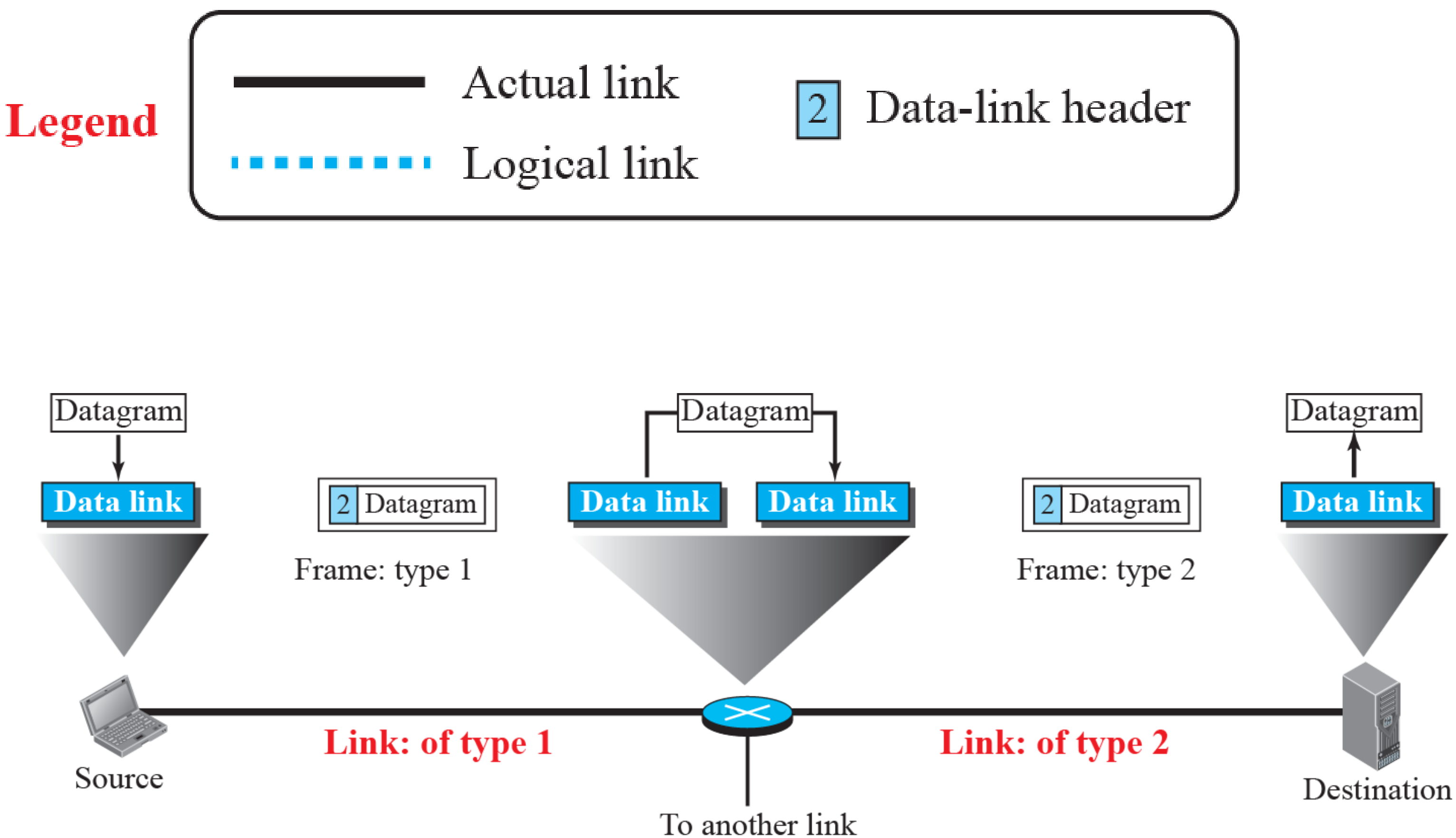
a. A small part of the Internet

b. Nodes and links

Services

- The data-link layer is located between the physical and the network layers.
- The data-link layer provides services to the network layer; it receives services from the physical layer.
- The duty scope of the data-link layer is node-to-node.
- When a packet is travelling in the Internet, the data-link layer of a node (host or router) is responsible for delivering a datagram to the next node in the path.
- For this purpose, the data-link layer of the sending node needs to encapsulate the datagram received from the network in a frame, and the data-link layer of the receiving node needs to decapsulate the datagram from the frame.
- The reason of conducting encapsulation and decapsulation at each intermediate node is that each link may be using a different protocol with a different frame format.
- Even if one link and the next are using the same protocol, encapsulation and decapsulation are needed because the link-layer addresses are normally different

Figure 9.3: A communication with only three nodes



Different Services

Different Services provided by Data-link Layer:

- Framing
- Flow Control
- Error Control
- Congestion Control

Framing

- Definitely, the first service provided by the data-link layer is framing.
- The data-link layer at each node needs to encapsulate the datagram (packet received from the network layer) in a frame before sending it to the next node.
- The node also needs to decapsulate the datagram from the frame received on the logical channel.
- Different data-link layers have different formats for framing.

Flow Control

- Whenever we have a producer and a consumer, we need to think about flow control.
- If the producer produces items that cannot be consumed, accumulation of items occurs.
- The sending data-link layer at the end of a link is a producer of frames; the receiving data-link layer at the other end of a link is a consumer.
- If the rate of produced frames is higher than the rate of consumed frames, frames at the receiving end need to be buffered while waiting to be consumed (processed).
- Definitely, we cannot have an unlimited buffer size at the receiving side.
- We have two choices:
  - The first choice is to let the receiving data-link layer drop the frames if its buffer is full.
  - The second choice is to let the receiving data-link layer send a feedback to the sending data-link layer to ask it to stop or slow down.
- Different data-link-layer protocols use different strategies for flow control.
- Flow control also occurs at the transport layer, with a higher degree of importance.

Error Control

- At the sending node, a frame in a data-link layer needs to be changed to bits, transformed to electromagnetic signals, and transmitted through the transmission media.
- At the receiving node, electromagnetic signals are received, transformed to bits, and put together to create a frame.
- Since electromagnetic signals are susceptible to error, a frame is susceptible to error.
- The error needs first to be detected.
- After detection, it needs to be either corrected at the receiver node or discarded and retransmitted by the sending node

Congestion Control

- Although a link may be congested with frames, which may result in frame loss, most data-link-layer protocols do not directly use a congestion control to alleviate congestion, although some wide-area networks do.
- In general, congestion control is considered an issue in the network layer or the transport layer because of its end-to-end nature

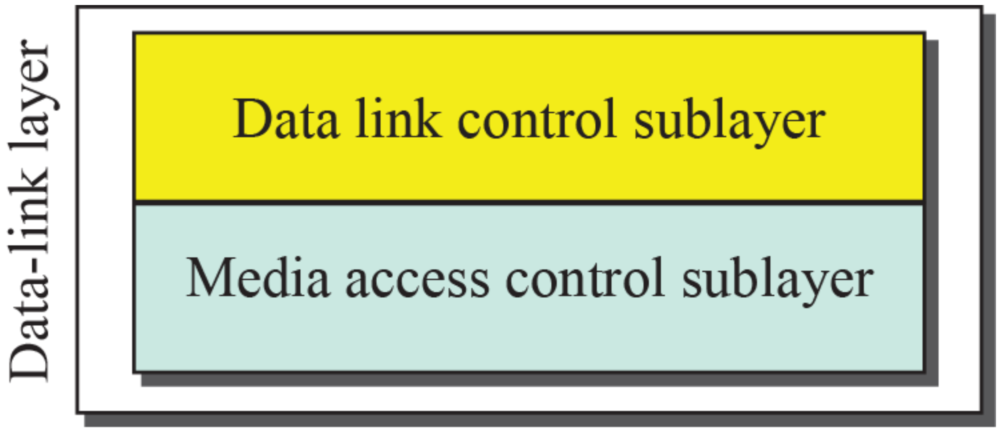
Two Categories of Links

- Although two nodes are physically connected by a transmission medium such as cable or air, we need to remember that the data-link layer controls how the medium is used.
- We can have a data-link layer that uses the whole capacity of the medium; we can also have a data-link layer that uses only part of the capacity of the link.
- In other words, we can have a point-to-point link or a broadcast link.
- In a point-to-point link, the link is dedicated to the two devices. (Example: Traditional Phone Service)
- In a broadcast link, the link is shared between several pairs of devices. (Example: Cellular Phone Service)

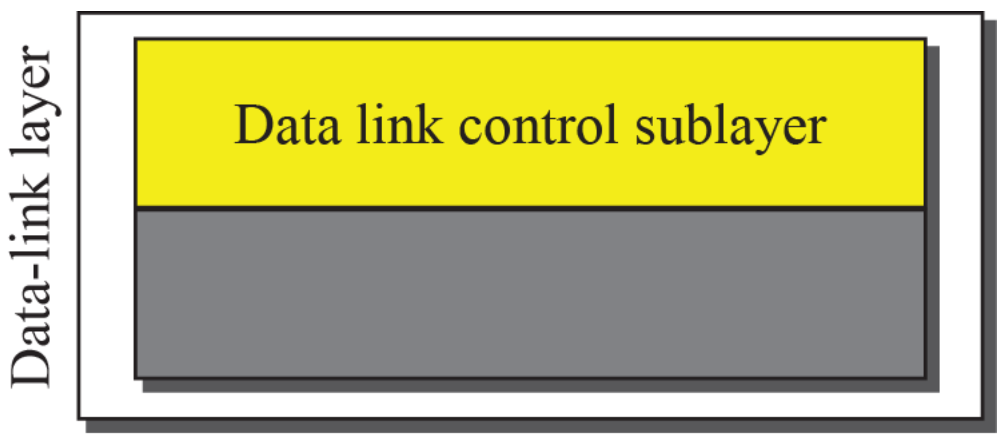
Two Sublayers

- To better understand the functionality of and the services provided by the link layer, we can divide the data-link layer into two sub-layers:
    - Data Link Control (DLC)
    - Media Access Control (MAC)
- The data link control sub-layer deals with all issues common to both point-to-point and broadcast links.
- The media access control sub-layer deals only with issues specific to broadcast links

Figure 9.3: Dividing the data-link layer into two sublayers



a. Data-link layer of a broadcast link

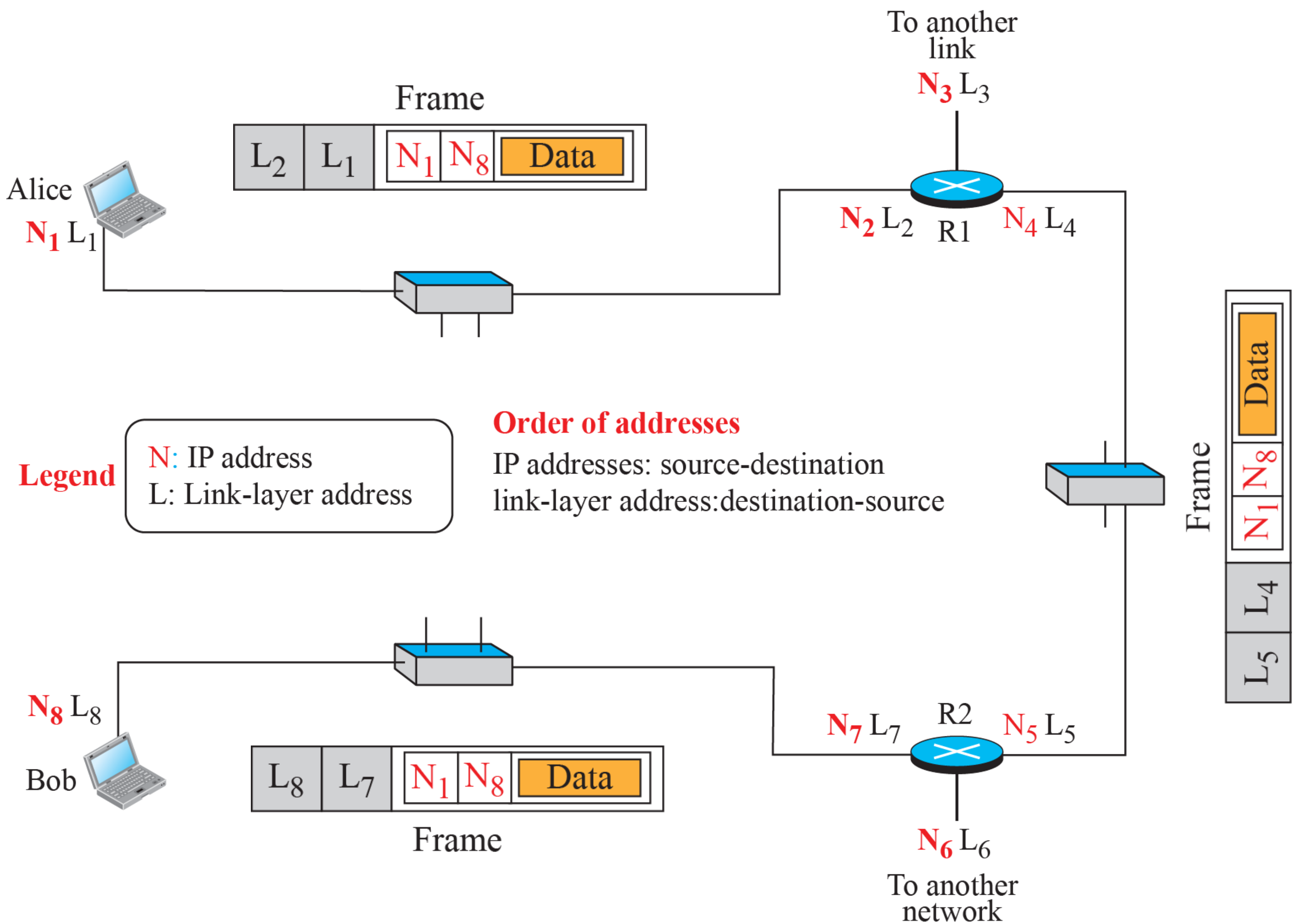


b. Data-link layer of a point-to-point link

LINK-LAYER ADDRESSING

- In Chapter 18, we have already discussed IP addresses as the identifiers at the network layer.
- However, in a internetwork such as the Internet we cannot make a datagram reach its destination using only IP addresses.
- The reason is that each datagram in the Internet, from the same source host to the same destination host, may take a different path.
- The source and destination IP addresses define the two ends but cannot define which links the packet should pass through.
- As the IP addresses in a datagram should not be changed we need another addressing mechanism in a connectionless internetwork: the link-layer addresses of the two nodes.
- A link-layer address is sometimes called a link address, sometimes a physical address, and sometimes a MAC address.

Figure 9.5: IP addresses and link-layer addresses in a small internet



Some Important Questions

- If the IP address of a router does not appear in any datagram sent from a source to a destination, why do we need to assign IP addresses to routers?
- Why do we need more than one IP address in a router, one for each interface?
- How are the source and destination IP addresses in a packet determined?
- How are the source and destination link-layer addresses determined for each link?
- What is the size of link-layer addresses?

Three Types of addresses

Some link-layer protocols define three types of addresses: unicast, multicast, and broadcast.

Example 9.1

The unicast link-layer addresses in the most common LAN, Ethernet, are 48 bits (six bytes) that are presented as 12 hexadecimal digits separated by colons; for example, the following is a link-layer address of a computer. The second digit needs to be an odd number.

A3:34:45:11:92:F1

Example 9.2

The multicast link-layer addresses in the most common LAN, Ethernet, are 48 bits (six bytes) that are presented as 12 hexadecimal digits separated by colons. The second digit, however, needs to be an even number in hexadecimal. The following shows a multicast address:

A2:34:45:11:92:F1

Example 9.3

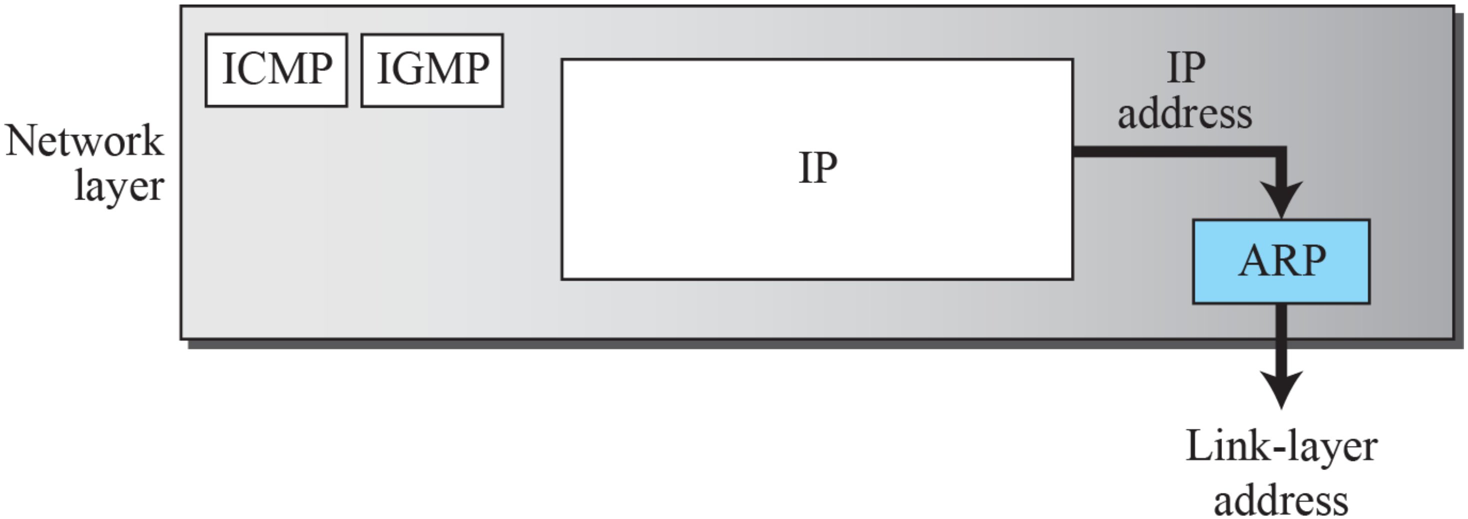
The broadcast link-layer addresses in the most common LAN, Ethernet, are 48 bits, all 1s, that are presented as 12 hexadecimal digits separated by colons. The following shows a broadcast address:

FF:FF:FF:FF:FF:FF

ARP (Address Resolution Protocol)

- Anytime a node has an IP datagram to send to another node in a link, it has the IP address of the receiving node.
- The source host knows the IP address of the default router.
- Each router except the last one in the path gets the IP address of the next router by using its forwarding table.
- The last router knows the IP address of the destination host.
- However, the IP address of the next node is not helpful in moving a frame through a link; we need the link-layer address of the next node.
- This is the time when the **Address Resolution Protocol (ARP)** becomes helpful. The ARP protocol is one of the auxiliary protocols defined in the network layer
- It belongs to the network layer, but we discuss it in this chapter because it maps an IP address to a logical-link address.
- ARP accepts an IP address from the IP protocol, maps the address to the corresponding link-laye address, and passes it to the data-link layer.

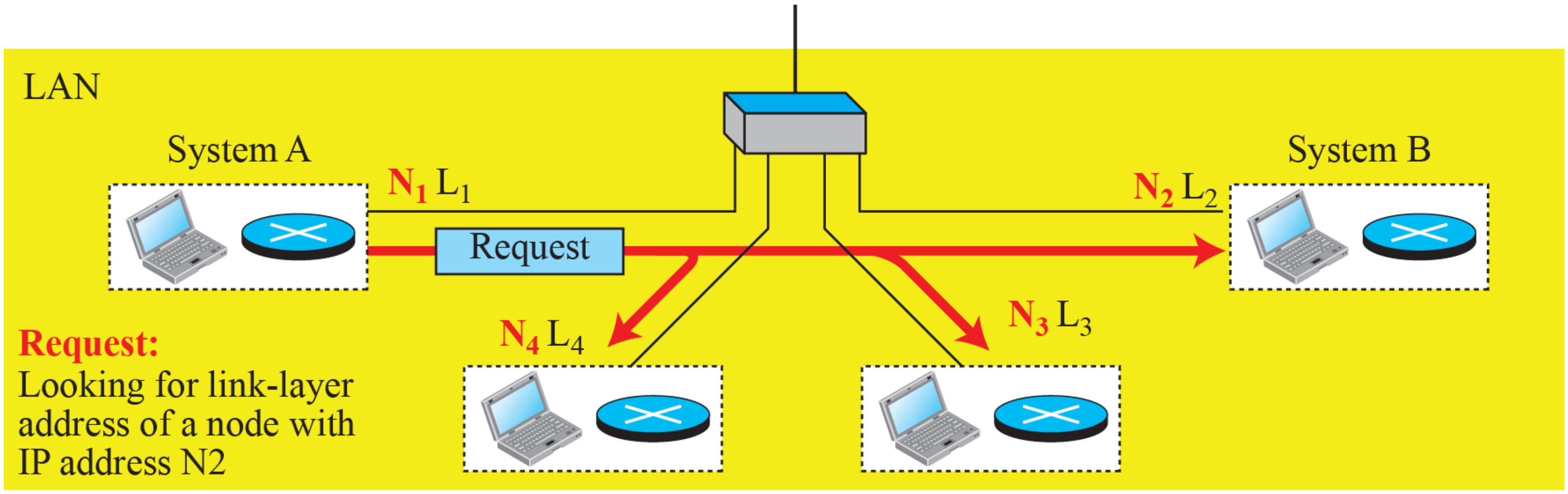
Figure 9.6: ***Position of ARP in TCP/IP protocol suite***



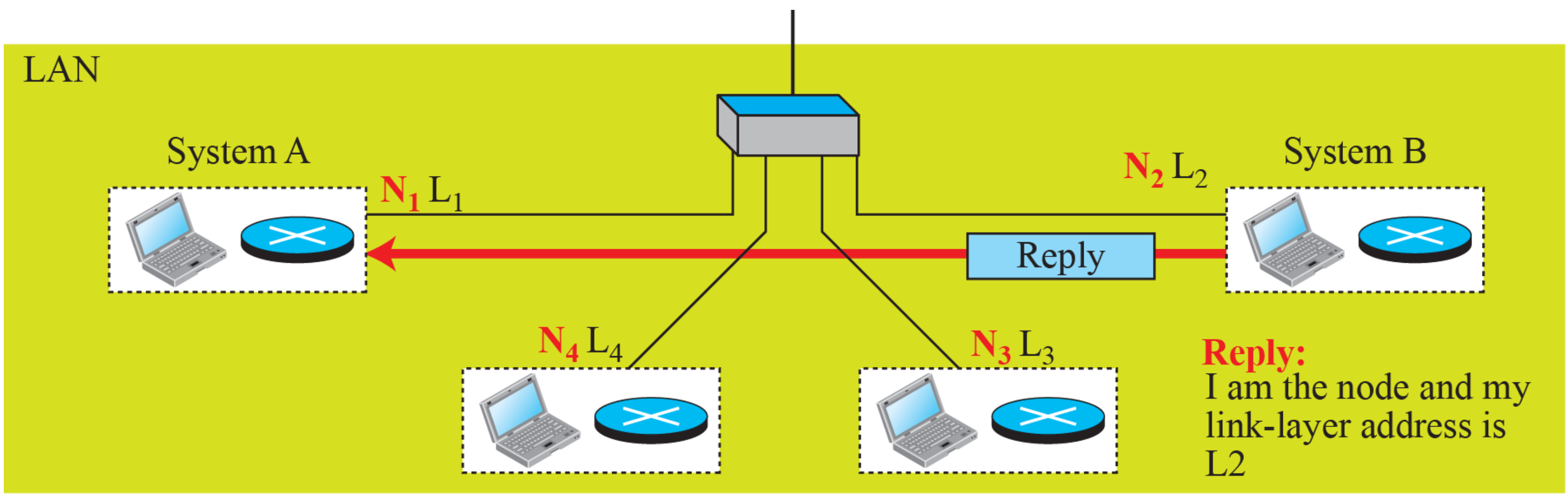
ARP Operation

- Anytime a host or a router needs to find the link-layer address of another host or router in its network, it sends an ARP request packet.
- The packet includes the link-layer and IP addresses of the sender and the IP address of the receiver.
- Because the sender does not know the link-layer address of the receiver, the query is broadcast over the link using the link-layer broadcast address

Figure 9.7: ARP operation



a. ARP request is broadcast



b. ARP reply is unicast

Caching

- Let us assume that there are 20 systems connected to the network (link): system A, system B, and 18 other systems. We also assume that system A has 10 datagrams to send to system B in one second.
- **a.** Without using ARP, system A needs to send 10 broadcast frames. Each of the 18 other systems need to receive the frames, decapsulate the frames, remove the datagram and pass it to their network-layer to find out the datagrams do not belong to them. This means processing and discarding 180 broadcast frames.
- **b.** Using ARP, system A needs to send only one broadcast frame. Each of the 18 other systems need to receive the frames, decapsulate the frames, remove the ARP message and pass the message to their ARP protocol to find that the frame must be discarded. This means processing and discarding only 18 (instead of 180) broadcast frames. After system B responds with its own data-link address, system **A can store the link-layer address in its cache memory**. The rest of the nine frames are only unicast. Since processing broadcast frames is expensive (time consuming), the first method is preferable.

ARP Packet Format

- The hardware type field defines the type of the link-layer protocol: Ethernet
- is given the type 1.
- The protocol type field defines the network-layer protocol: IPv4 protocol is $(0800)_{16}$.
- The source hardware and source protocol addresses are variable-length fields defining the link-layer and network-layer addresses of the sender.
- The destination hardware address and destination protocol address fields define the receiver link-layer and network-layer addresses.
- An ARP packet is encapsulated directly into a data-link frame.
- The frame needs to have a field to show that the payload belongs to the ARP and not to the network-layer datagram.

Figure 9.8: ARP packet Format

Hardware: LAN or WAN protocol
Protocol: Network-layer protocol

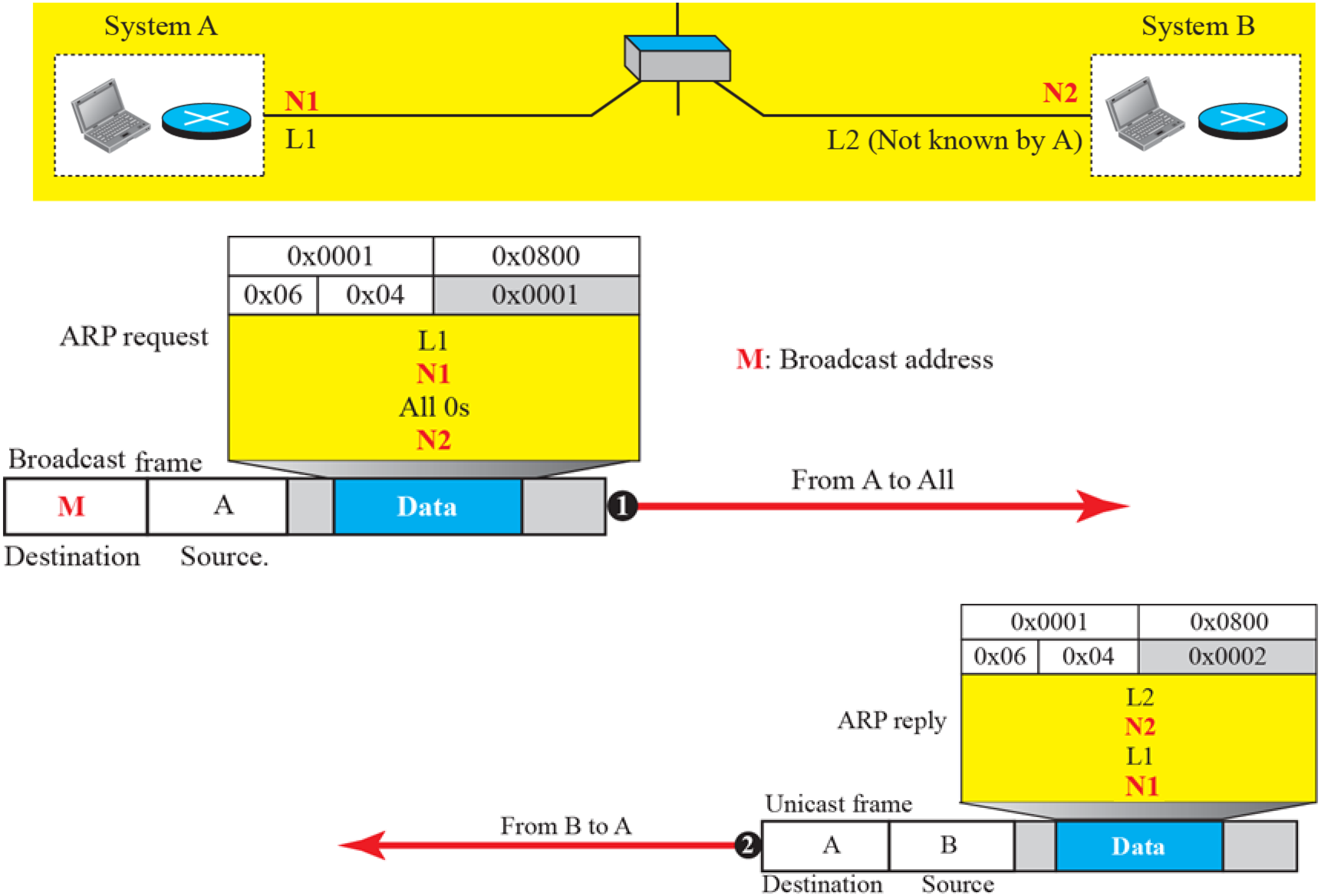
0 8 16 31

Hardware Type		Protocol Type
Hardware length	Protocol length	Operation **Request:1, Reply:2**
Source hardware address		
Source protocol address		
Destination hardware address (Empty in request)		
Destination protocol address		

Example 9.4

A host with IP address N1 and MAC address L1 has a packet to send to another host with IP address N2 and physical address L2 (which is unknown to the first host). The two hosts are on the same network. Figure 9.9 shows the ARP request and response messages.

Figure 9.9: Example 9.4



An Example of Communication

- To show how communication is done at the data-link layer and how link-layer addresses are found, we will go through a simple example:
- Assume Alice needs to send a datagram to Bob, who is three nodes away in the Internet. How Alice could finds the network-layer address of Bob using DNS.

Figure 9.10: The internet for our example

Figure 9.11: Flow of packets at Alice site

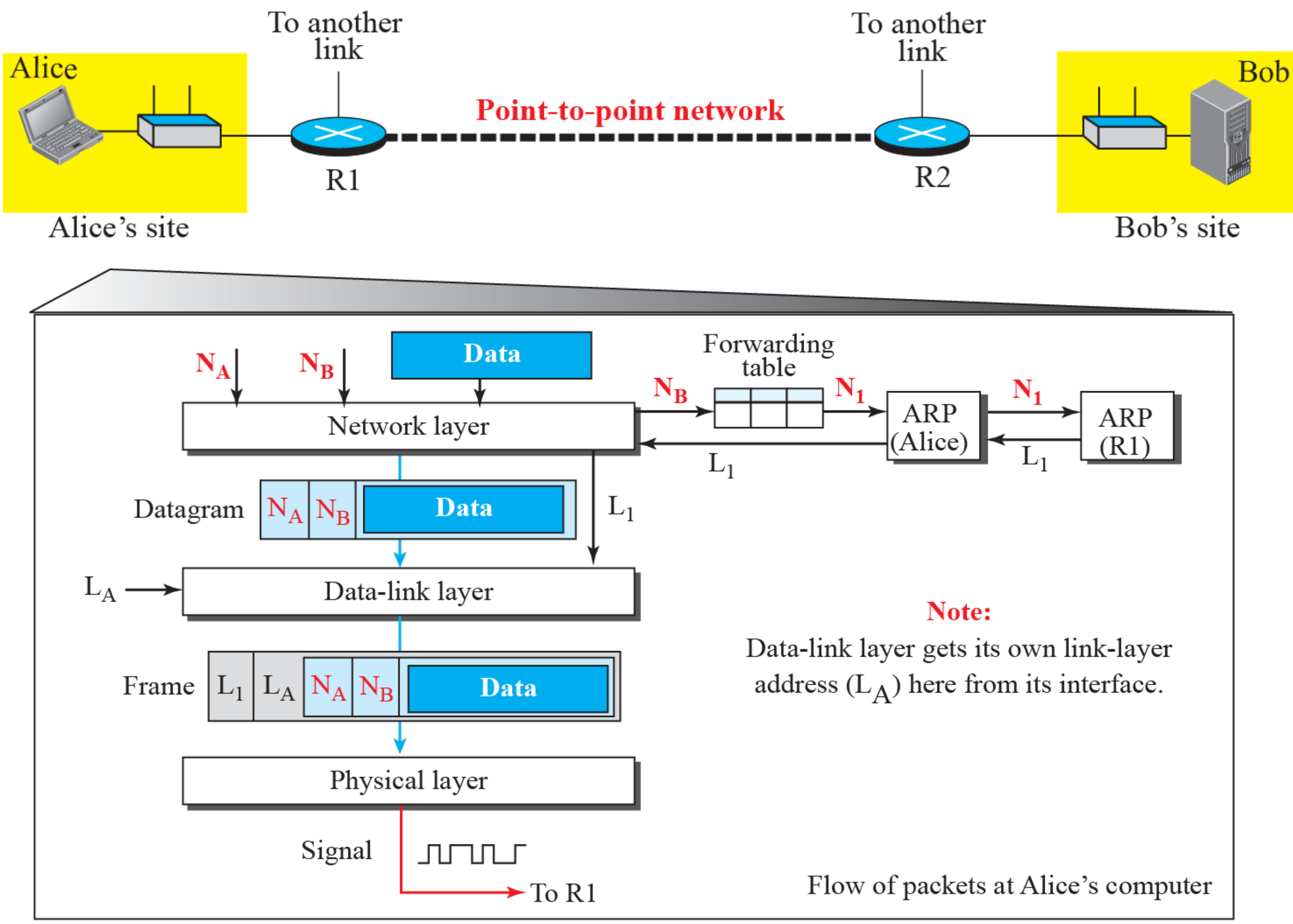


Figure 9.12: Flow of activities at router R1

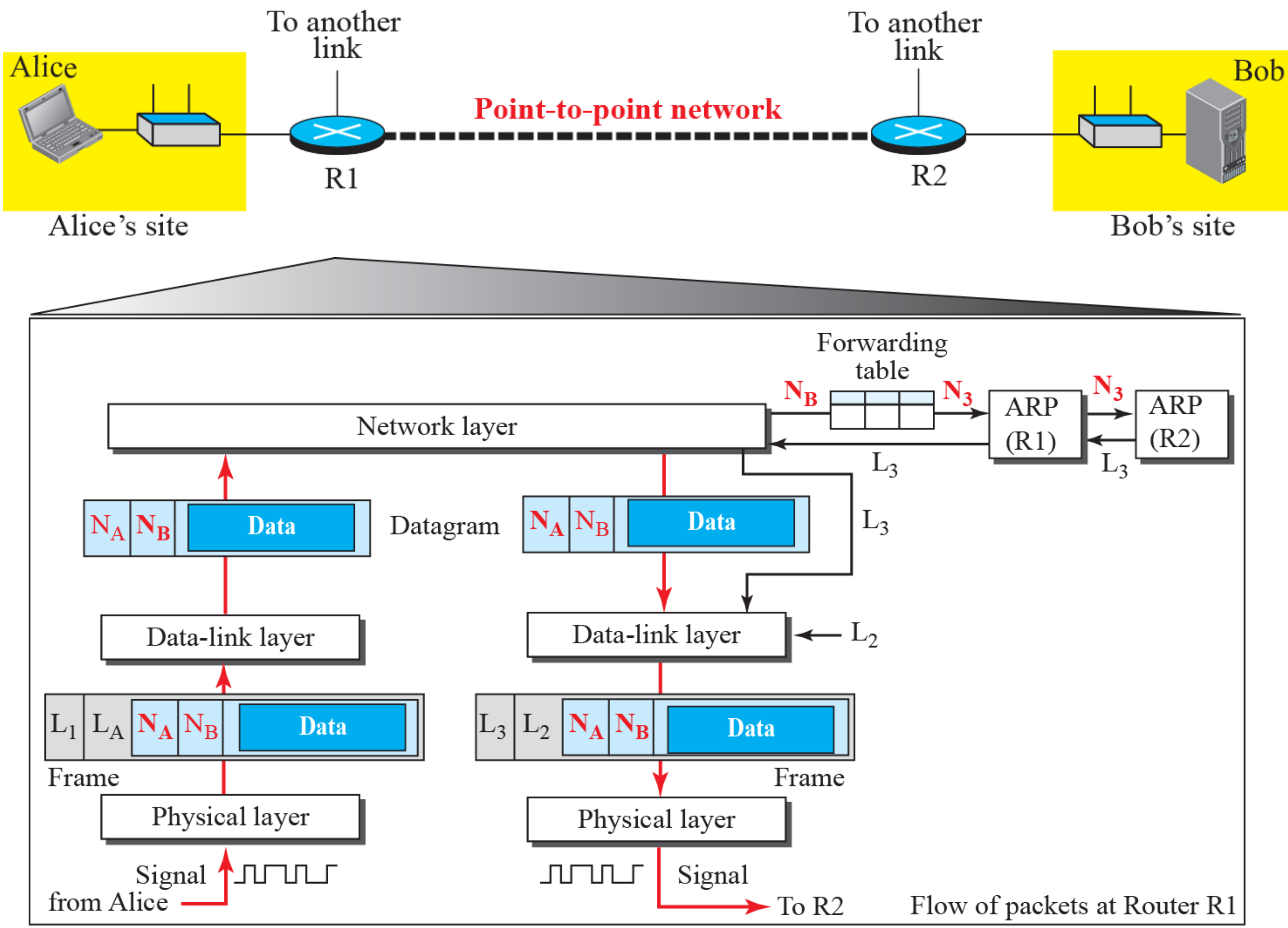


Figure 9.13: Flow of activities at router R2

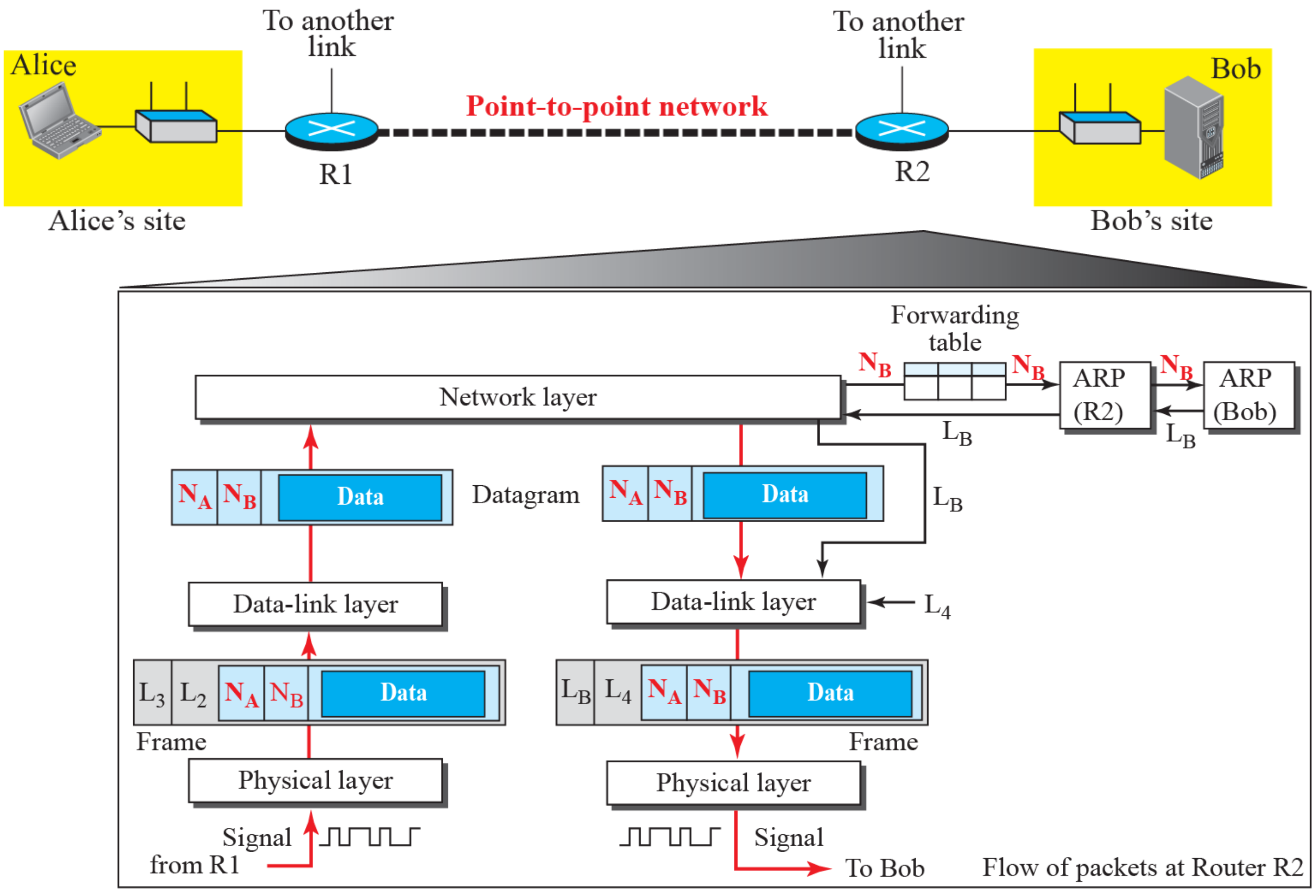


Figure 9.14: Activities at Bob's site