Introduction to Data Communications and Networks
(CYB 204)
OSI and TCP/IP Models

Egena Onu, Ph.D.
Department of Computer Science
Bingham University
Karu.

Introduction

- Network construction was inconsistent in the beginning. Each vendor provided a unique solution.

- It was problematic that one vendor's solution was incompatible with another vendor’s.

Protocols

- A protocol is a set of rules and formats that govern the communication between communicating peers.
  - Set of valid messages
  - Meaning of each message

- A protocol is necessary for any function that requires cooperation between peers.

Protocols

- What a protocol tells us
  - Syntax of a message
    - What fields does it contain?
    - In what format?
  - Semantics of a message
    - What does a message mean?
    - For example, not-OK message means receiver got a corrupted file
  - Actions to take on receipt of a message
    - For example, on receiving not-OK message, retransmit the entire file

Protocols

- Another way to view a protocol
  - As providing a service
  - The example protocol provides reliable file transfer service
  - Peer entities use a protocol to provide a service to a higher-level peer entity
    - For example, postal workers use a protocol to present customers with the abstraction of an unreliable letter transfer service

Protocols

- Protocol layering
  - A network that provides many services needs many protocols
  - Turns out that some services are independent
  - But others depend on each other
  - Protocol A may use protocol B as a step in its execution
    - For example, packet transfer is one step in the execution of the example reliable file transfer protocol
  - This form of dependency is called layering reliable
    - File transfer is layered above packet transfer protocol
    - Like a subroutine

Protocols

- Protocol Layering
  - Layering brakes a rather complex problem into smaller, simpler pieces
  - Layering provides the application with sophisticated services
  - Each layer provides a clean abstraction to the layer above

Protocols

- Protocol stack
  - A set of protocol layers
  - Each layer uses the layer below and provides a service to the layer above
  - Key idea
    - Once we define a service provided by a layer, we need know nothing more about the details of how the layer actually implements the service
    - Information hiding
    - Decouples changes

Protocols

- The Importance of Layering
  - Breaks up a complex problem into smaller manageable pieces
    - can compose simple service to provide complex ones
  - Abstraction of implementation details
    - Separation of implementation and specification
    - Can change implementation as long as service interface is maintained
  - Can reuse functionality
    - Upper layers can share lower layer functionality

Open System Interconnection (OSI) Model

- A networking model offers a generic means to separate computer networking functions into multiple layers.
- Each of these layers relies on the layers below it to provide supporting capabilities and performs support to the layers above it.
- Such a model of layered functionality is also called a “protocol stack” or “protocol suite”.

Open System Interconnection (OSI) Model

- Open Systems Interconnection (OSI) Model.
  - A set of protocols is open if
    - Protocol details are publicly available
    - Changes are managed by an organization whose membership and transactions are open to the public
  - A system that implements open protocols is called an open system
  - International Organization for Standards (ISO) prescribes a standard to connect open systems
    - Open system interconnect (OSI)
  - It defines and is used to understand how data is transferred from one computer to another in a computer network using a set of protocols.
  - The OSI has greatly influenced thinking on protocol stacks

Open System Interconnection (OSI) Model

- Open Systems Interconnection (OSI) Model.
  - The OSI model divides the communications processes into seven layers.

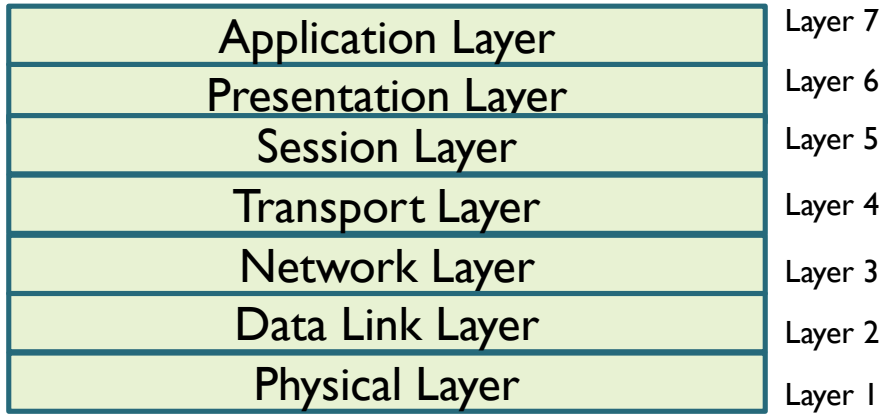


Figure 1: OSI Model

  - Each layer both performs specific functions to support the layers above it and offers services to the layers below it.
  - The three lowest layers focus on passing traffic through the network to an end system.
  - The top four layers come into play in the end system to complete the process.

Open System Interconnection (OSI) Model

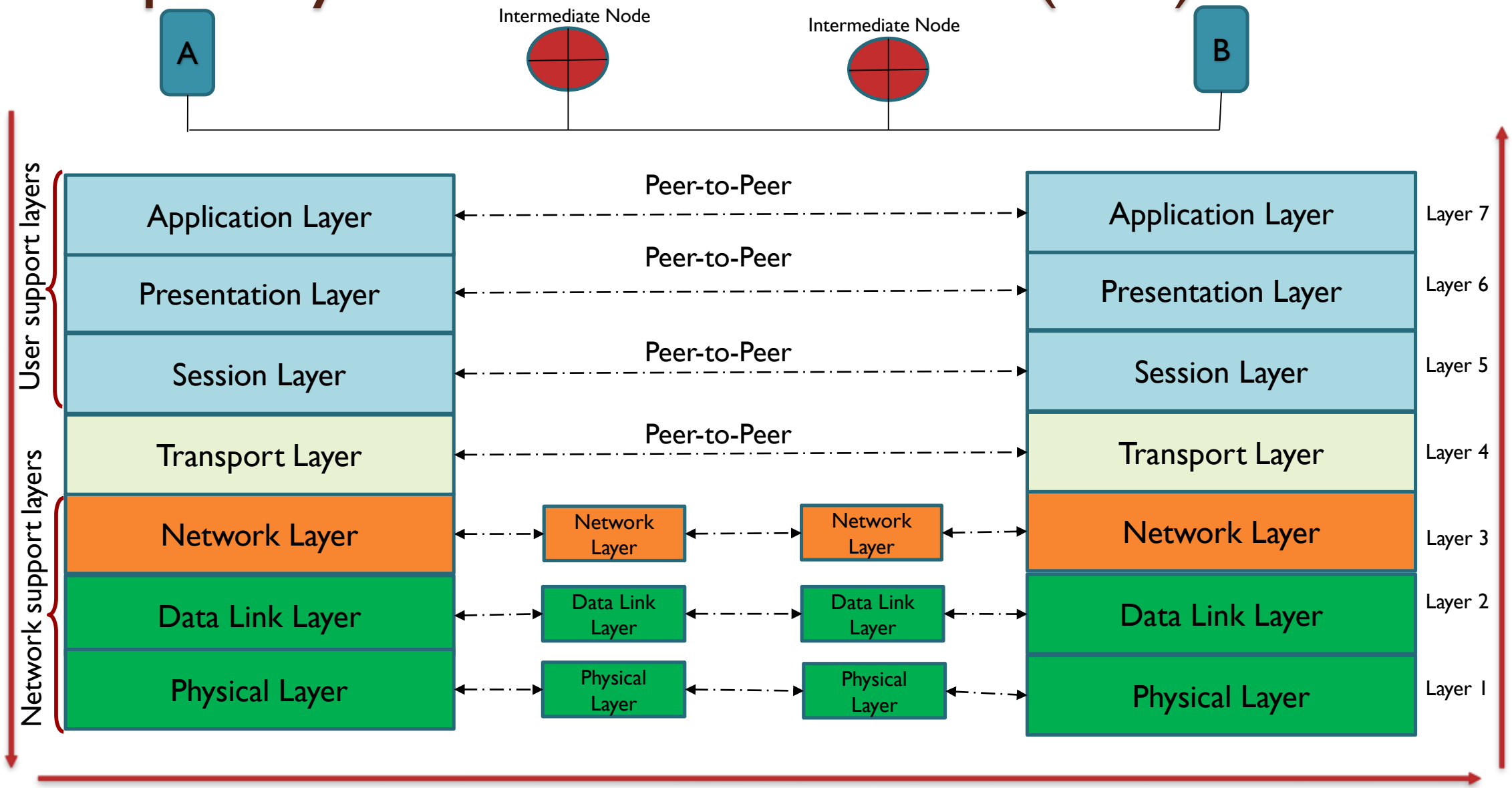


Figure 2: Data transmission through OSI layers

Open System Interconnection (OSI) Model

- Interfaces Between Layers
  - Within a single machine, the passing of the data and network information down through the layers of the sending machine and back up through the layers of the receiving machine is made possible by an interface between each pair of adjacent layers.
  - Each interface defines what information and services a layer must provide for the layer above it.
  - Layer 3, for example, uses the services provided by layer 2

Open System Interconnection (OSI) Model

- **Encapsulation and Protocol Data Unit (PDU)**
  - As application data is passed down the protocol stack on its way to be transmitted across the network media, various protocols add information to it at each level.
  - This is commonly known as the encapsulation process.
  - The form that a piece of data takes at any layer is called a protocol data unit (PDU).
  - During encapsulation, each succeeding layer encapsulates the PDU that it receives from the layer above in accordance with the protocol being used.
  - At each stage of the process, a PDU has a different name to reflect its new functions.
    - Data - The general term for the PDU used at the application layer
    - Segment - Transport layer PDU
    - Packet - Network layer PDU
    - Frame - Data Link layer PDU
    - Bits - A PDU used when physically transmitting data over the medium.

Open System Interconnection (OSI) Model

- Open Systems Interconnection (OSI) Model.
  - Benefits of the OSI:
    - Helps users understand the big picture of networking.
    - Helps users understand how hardware and software elements function together.
    - Makes troubleshooting easier by separating networks into manageable pieces.
    - Defines terms that networking professionals can use to compare basic functional relationships on different networks.
    - Helps users understand new technologies as they are developed.
    - Aids in interpreting vendor explanations of product functionality.

TCP/IP Model

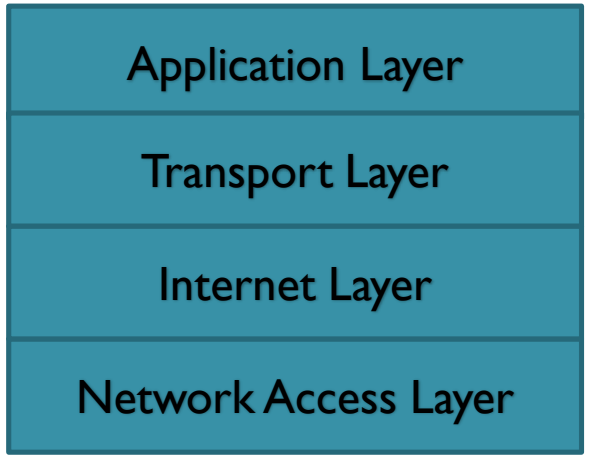
- The name "TCP/IP" refers to an entire suite of data communications protocols.
- The suite gets its name from two of the protocols that belong to it: the Transmission Control Protocol (TCP) and the Internet Protocol (IP).
- TCP/IP is the traditional name for this protocol suite. The TCP/IP protocol suite is also called the Internet Protocol Suite (IPS). Both names are acceptable.

TCP/IP Model

- Features of TCP/IP
  - Open protocol standards, freely available and developed independently from any specific computer hardware or operating system. Because it is so widely supported, TCP/IP is ideal for uniting different hardware and software components, even if you don't communicate over the Internet.
  - Independence from specific physical network hardware. This allows TCP/IP to integrate many different kinds of networks. TCP/IP can be run over an Ethernet, a DSL connection, a dial-up line, an optical network, and virtually any other kind of physical transmission medium.
  - A common addressing scheme that allows any TCP/IP device to uniquely address any other device in the entire network, even if the network is as large as the worldwide Internet.
  - Standardised high-level protocols for consistent, widely available user services.

TCP/IP Model

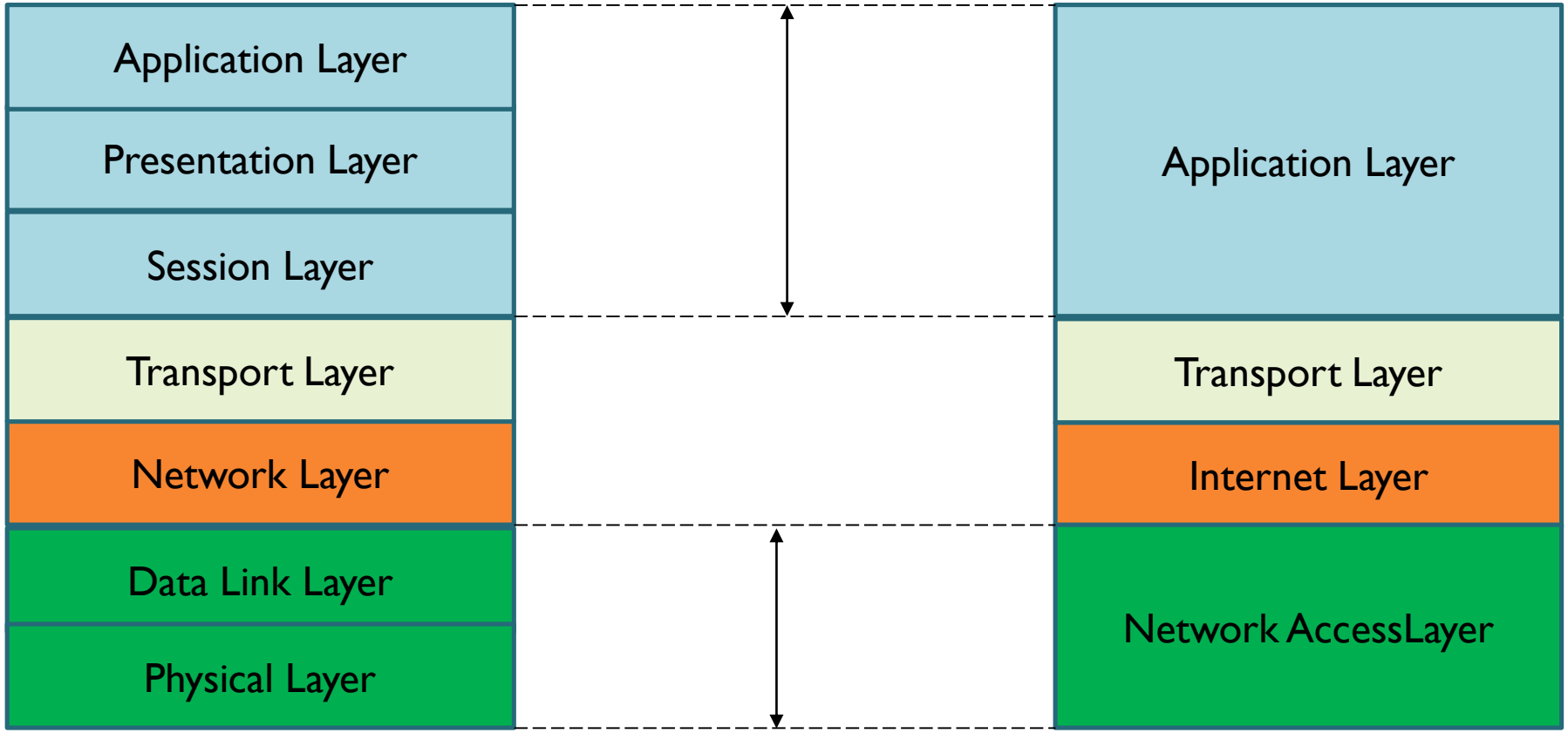
- TCP/IP Architecture
  - While there is no universal agreement about how to describe TCP/IP with a layered model, TCP/IP is generally viewed as being composed of fewer layers than the seven used in the OSI model.
  - Most descriptions of TCP/IP define three to five functional levels in the protocol architecture.



  - The whole of the Internet is based on the TCP/IP model.

TCP/IP Model

- OSI vs TCP/IP



Questions!