

Protocol Architecture The "Common Language"

Need for Protocols

- **Protocols are a set of rules and conventions.** By enforcing that communicating parties adhere to a common protocol, communication is made possible.

Copyright by Jorg Liebeherr 98, 99

Network Architecture

- A **Network Architecture** is a structured set of protocols that implement the exchange of information between computers

Copyright by Jorg Liebeherr 98, 99

Programming the Communication

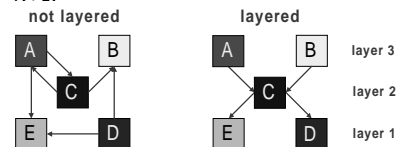
- Any complex programming task is made simpler by using several levels of abstraction
- Abstraction hides the programming details of lower levels
- Communication tasks give rise to standard abstractions
- Committees such as the IETF (Internet Engineering Task Force) are in charge for standardizing the abstractions related to internet communication

The Levels of Abstraction

- Multiple levels of communication abstractions have been defined by standardization bodies
- Each level is enforced by a corresponding protocol
- Each level uses the abstract services of the level below
- Abstraction achieves separation of concerns
 - A protocol needn't know how the lower-level abstractions are implemented and what the higher level abstractions are.

Layered Network Architecture

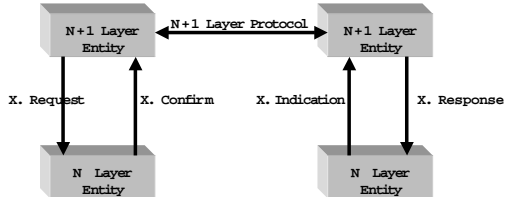
- In a **Layered Network Architecture**, the services are grouped in a hierarchy of layers.
 - A protocol of layer N uses only services of layer N-1.
 - A protocol of layer N provides services only to layer N+1.



Copyright by Jorg Liebeherr 98, 99

Service Primitives

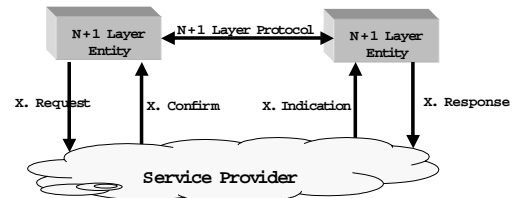
- Assume that some protocol implements service "X".



Copyright by Jorg Liebeherr 98, 99

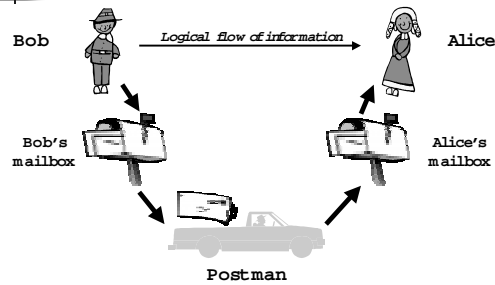
Service Primitives

- Recall:** A layer N+1 protocol entity sees the lower layers only as an abstract service provider



Copyright by Jorg Liebeherr 98, 99

Example: Sending a Letter



Copyright by Jorg Liebeherr 98, 99

(Un-)Acknowledged Service

- Unconfirmed service:



- Acknowledged service:



Copyright by Jorg Liebeherr 98, 99

Protocol Architectures

- There are only few protocol architectures that are relevant today:
 - OSI Reference Model
 - TCP/IP Protocols Suite
 - ATM Protocol Stack

Copyright by Jorg Liebeherr 98, 99

OSI Reference Model

- The OSI model defines seven layers:
 - Layer 7: Application Layer
 - Layer 6: Presentation Layer
 - Layer 5: Session Layer
 - Layer 4: Transport Layer
 - Layer 3: Network Layer
 - Layer 2: Data Link Layer
 - Layer 1: Physical Layer
- Interconnection Media

Copyright by Jorg Liebeherr 98, 99

A Tour of the OSI Layers

- **Transport Layer (Layer 4):**
 - Controls delivery of data between hosts
 - Connection management, error control, flow control, multiplexing, TCP, UDP.

Copyright by Jorg Liebeherr 98, 99

A Tour of the OSI Layers

- **Session Layer (Layer 5):**
 - Support the dialog between cooperating application programs
 - Session management, synchronization
 - Example: RPC

Copyright by Jorg Liebeherr 98, 99

A Tour of the OSI Layers

- **Presentation Layer (Layer 6):**
 - Data conversion into application format
 - Encryption/decryption
 - Secure sockets

Copyright by Jorg Liebeherr 98, 99

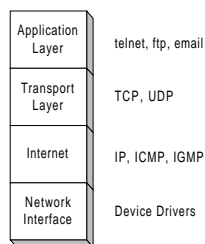
A Tour of the OSI Layers

- **Application Layer (Layer 7):**
 - Provides network access to application programs
 - Everything is application specific
 - Example: File Transfer, Electronic Mail

Copyright by Jorg Liebeherr 98, 99

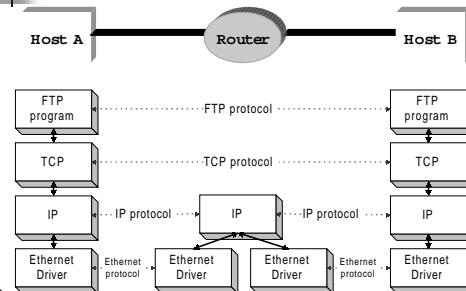
TCP/IP Protocol Suite

- The TCP/IP protocol suite was first defined in 1974
- The TCP/IP protocol suite is the protocol architecture of the **Internet**
- The TCP/IP suite has four layers: Application, Transport, Internet, and Network Interface Layer



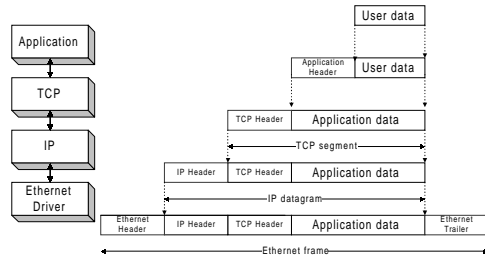
Copyright by Jorg Liebeherr 98, 99

Example: File Transfer



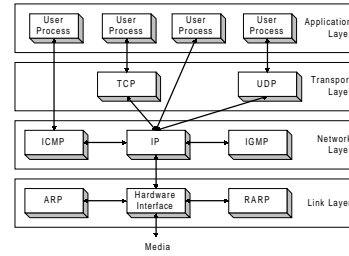
Copyright

Encapsulation in the TCP/IP Suite



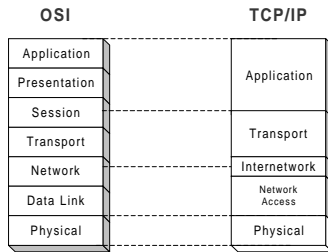
Copyright by Jorg Liebeherr 98, 99

TCP/IP Protocol Suite



Copyright by Jorg Liebeherr 98, 99

Comparison of OSI Model and TCP/IP Suite



Copyright by Jorg Liebeherr 98, 99