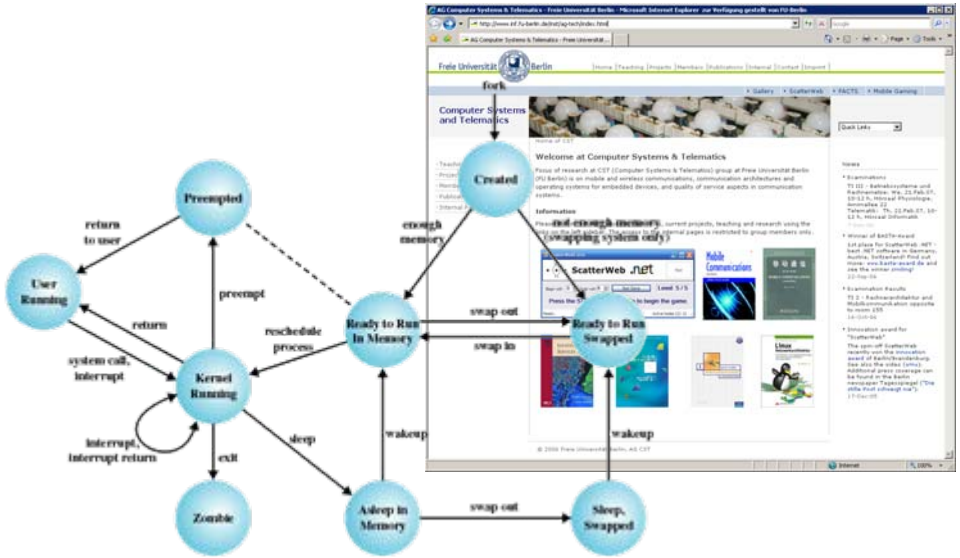
TI III: Operating & Communication Systems

Example

Under the Hood of

Surfing the Web

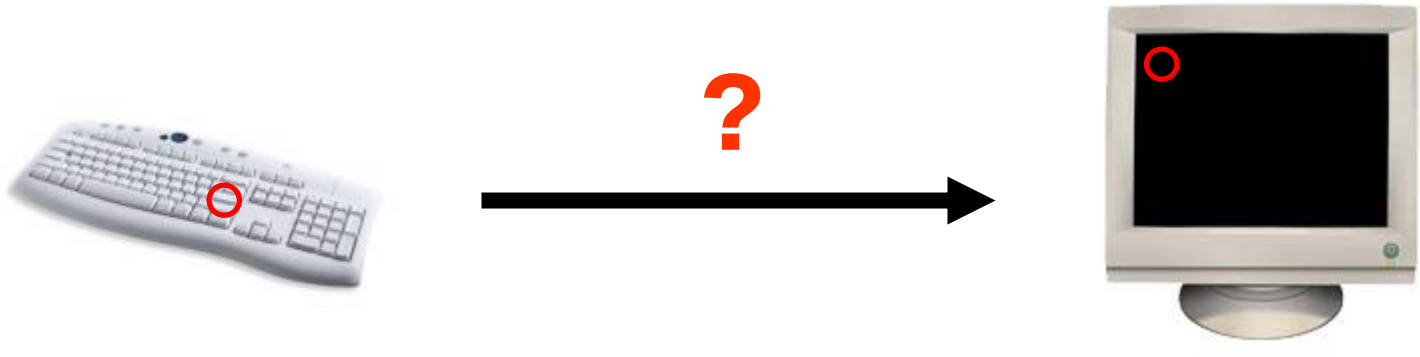


Content (3)

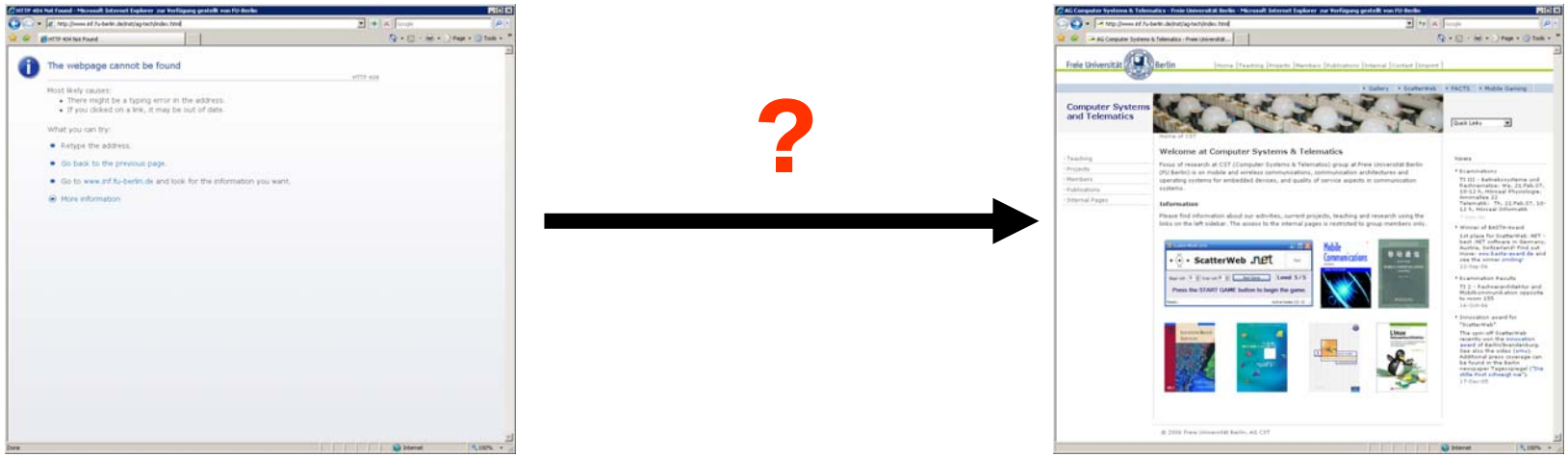
14. Security
    - Basic concepts & terms
    - Cryptology
    - Examples
        - Firewalls
        - Virtual Private Networks (VPNs)
        - IP Security
        - Email security with PGP
15. Application
    - Domain Name System
    - Email
    - World Wide Web
16. **Example**
    - **Under the Hood of Surfing the Web**

A Comprehensive Example

- What happens if one presses a key on the computer?



- What if that key causes an web page to be displayed?



Keyboard Interrupt

- Keyboard controller raises interrupt flag
- CPU interrupts execution of current process and starts Interrupt Service Routine (ISR)
  - unconditional jump

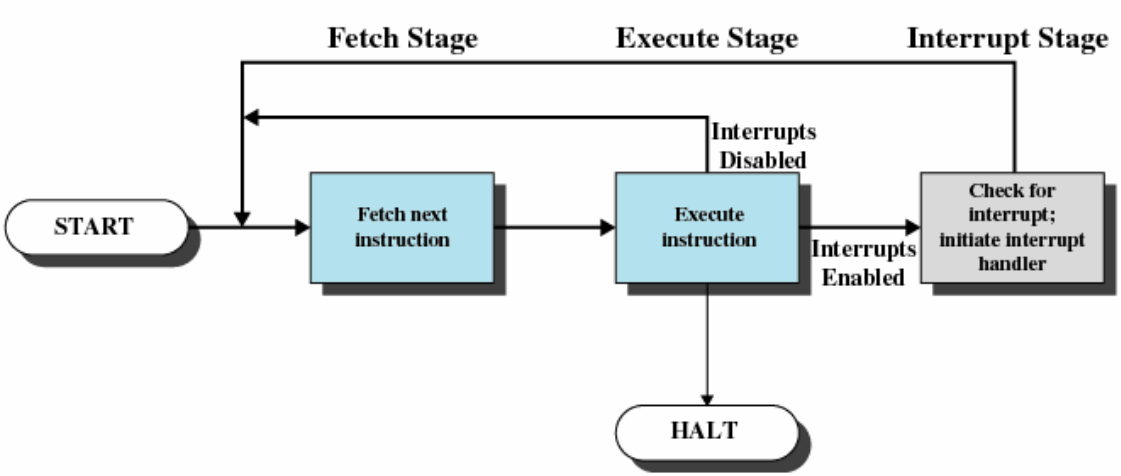
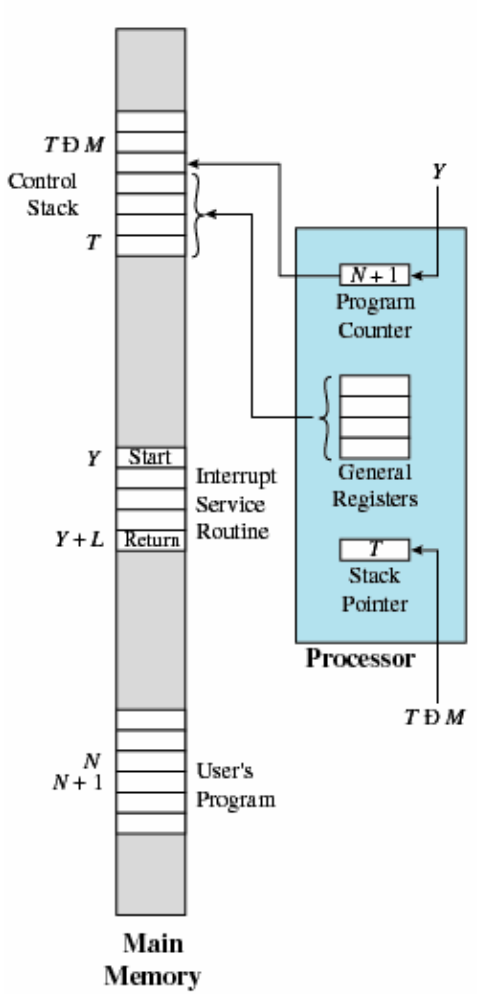


Figure 1.7 Instruction Cycle with Interrupts



(a) Interrupt occurs after instruction at location *N*

Keyboard Interrupt Handling

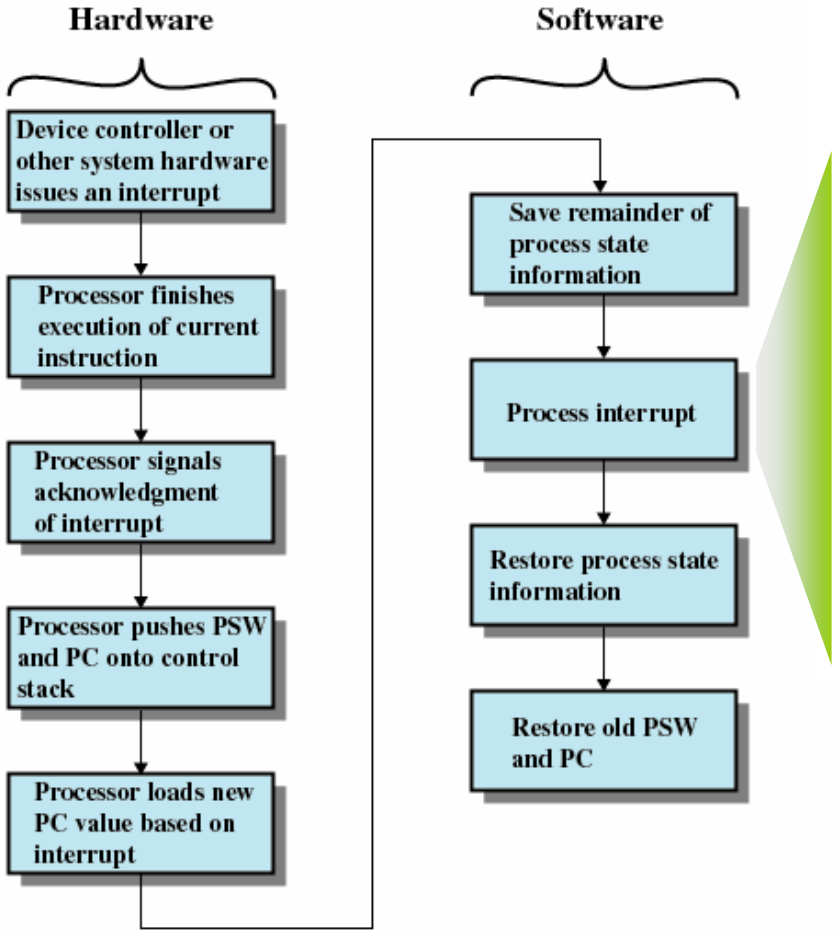
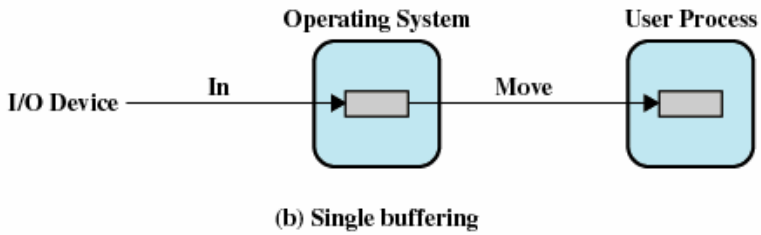


Figure 1.10 Simple Interrupt Processing

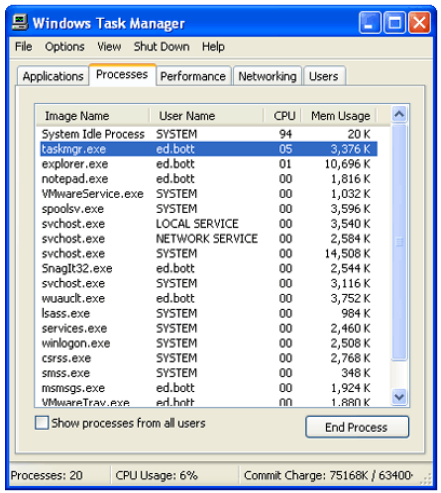
- ISR processes input from keyboard
  1. Clears interrupt flag
  2. Transfers data from device into buffer
  3. Establishes owner of device
  4. Triggers notification of user process



(b) Single buffering

In the meantime…

- Web browser is one of many processes running locally
- Other processes include
  - Other user processes (possibly of different users)
  - System processes implementing system services
  - Kernel processes



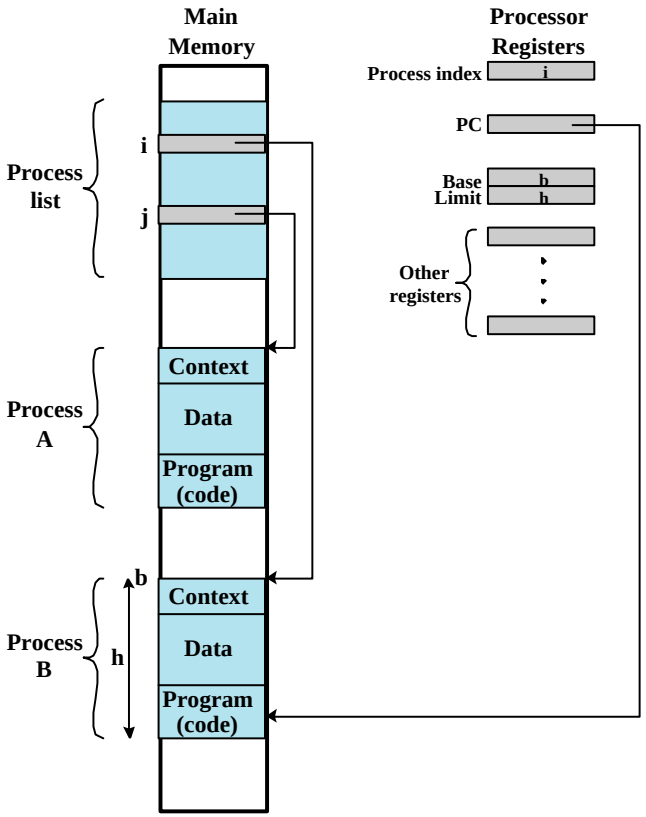
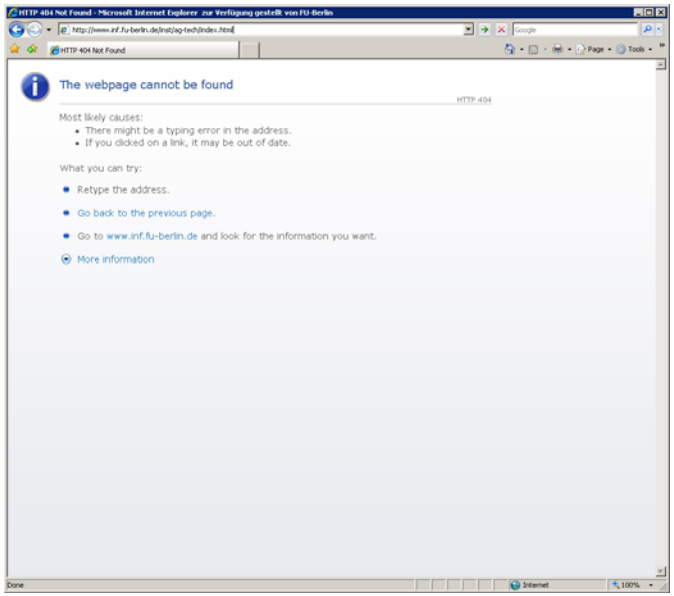


Figure 2.8 Typical Process Implementation

Web Browser Process in Detail

- Web browser processes
  - Currently waiting for input
    - e.g. using **`select()`**
  - Process state **blocked**



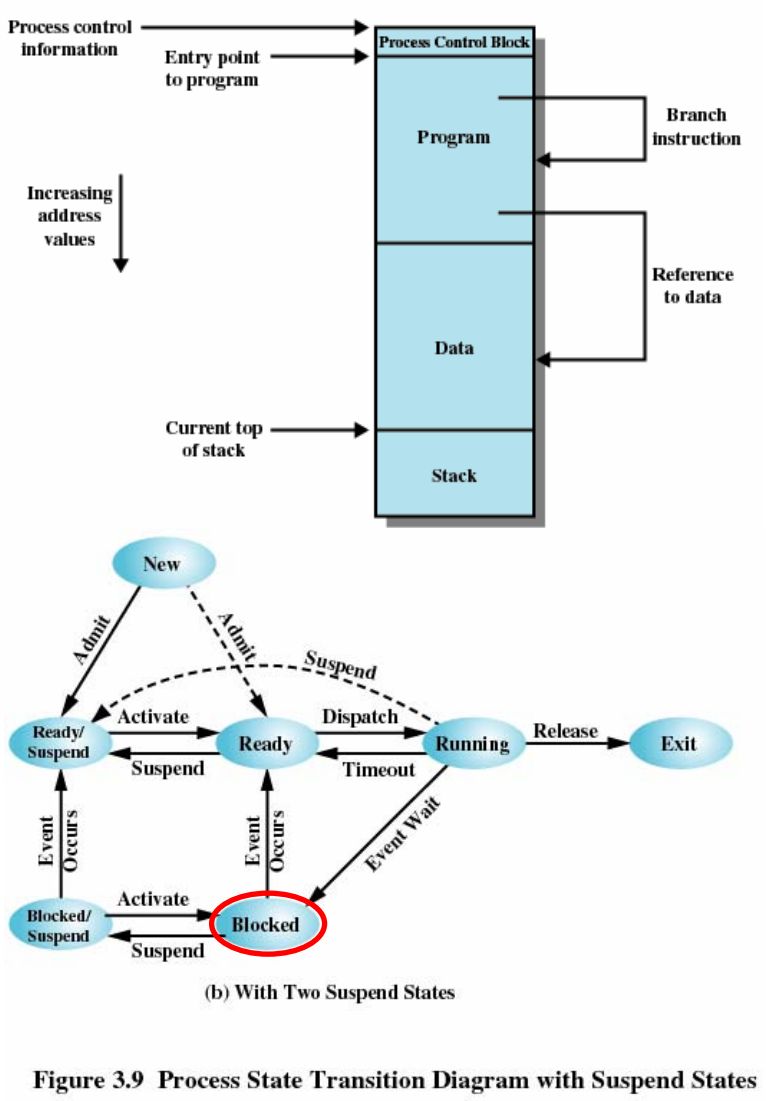
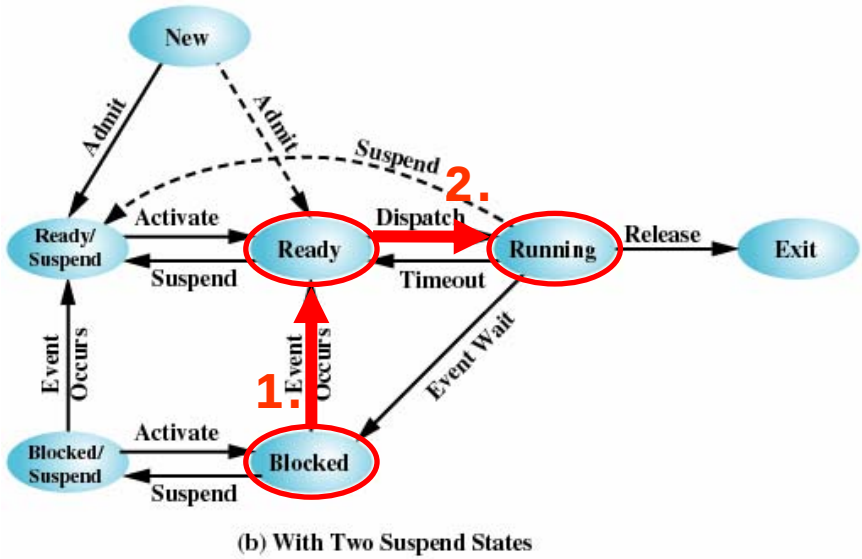


Figure 3.9 Process State Transition Diagram with Suspend States

Reaction to External Event

1. ISR changes process state to **ready**
2. Scheduling algorithm eventually changes process state to **running**



(b) With Two Suspend States

Figure 3.9 Process State Transition Diagram with Suspend States

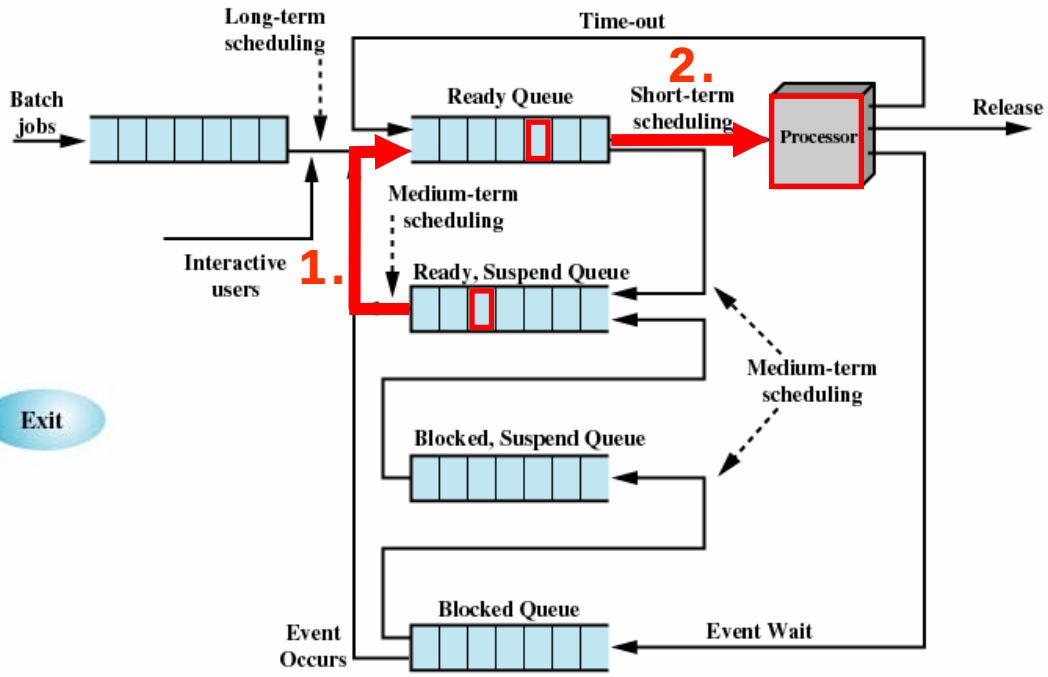
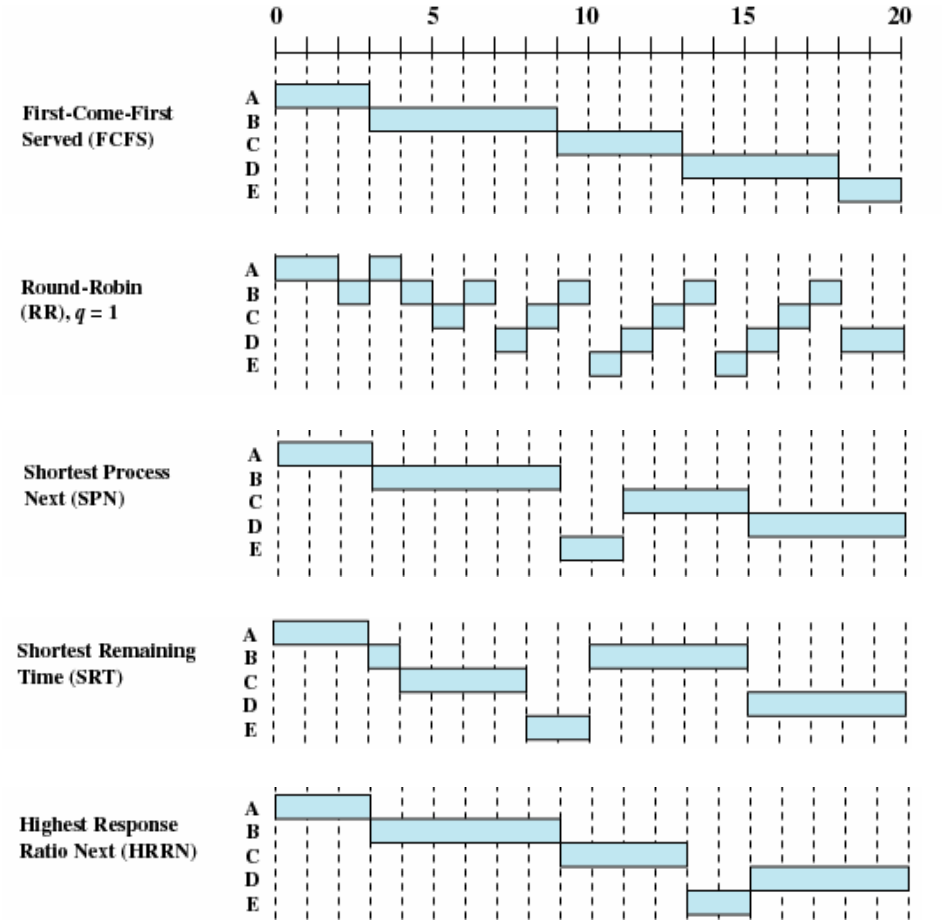


Figure 9.3 Queuing Diagram for Scheduling

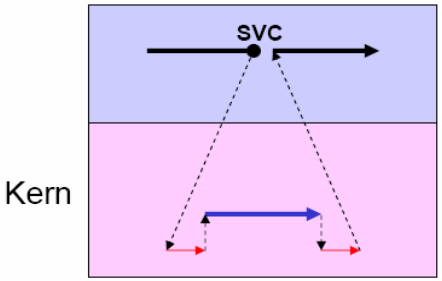
Process Scheduling

- Scheduling is handled by variety of scheduling algorithms
  - Non-preemptive / preemptive
  - Maximize throughput, responsiveness, etc...
- Processes may have priorities
  - Priority inversion due to lock on shared resources
  - Priority inheritance



Web Browser Processes Event

- Assume input requires web browser to display a web page with a given URL
  1. String processing (user space)
  2. Connect to server and retrieve necessary data (system calls)



  3. Render web page (user space)
  4. Update user interface (system calls)

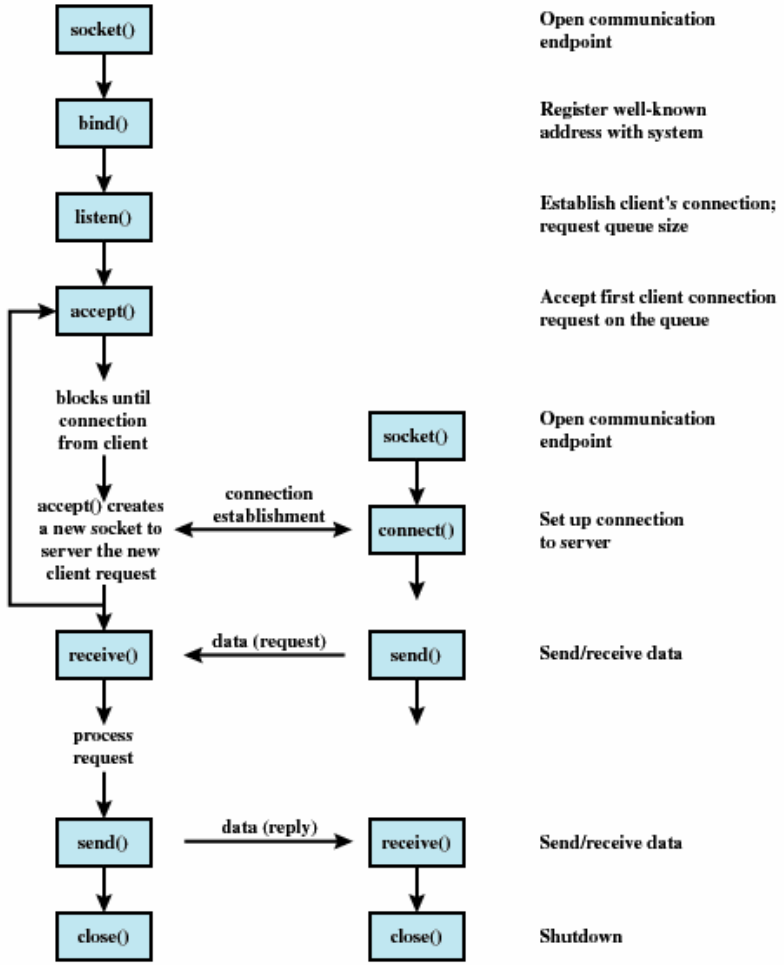
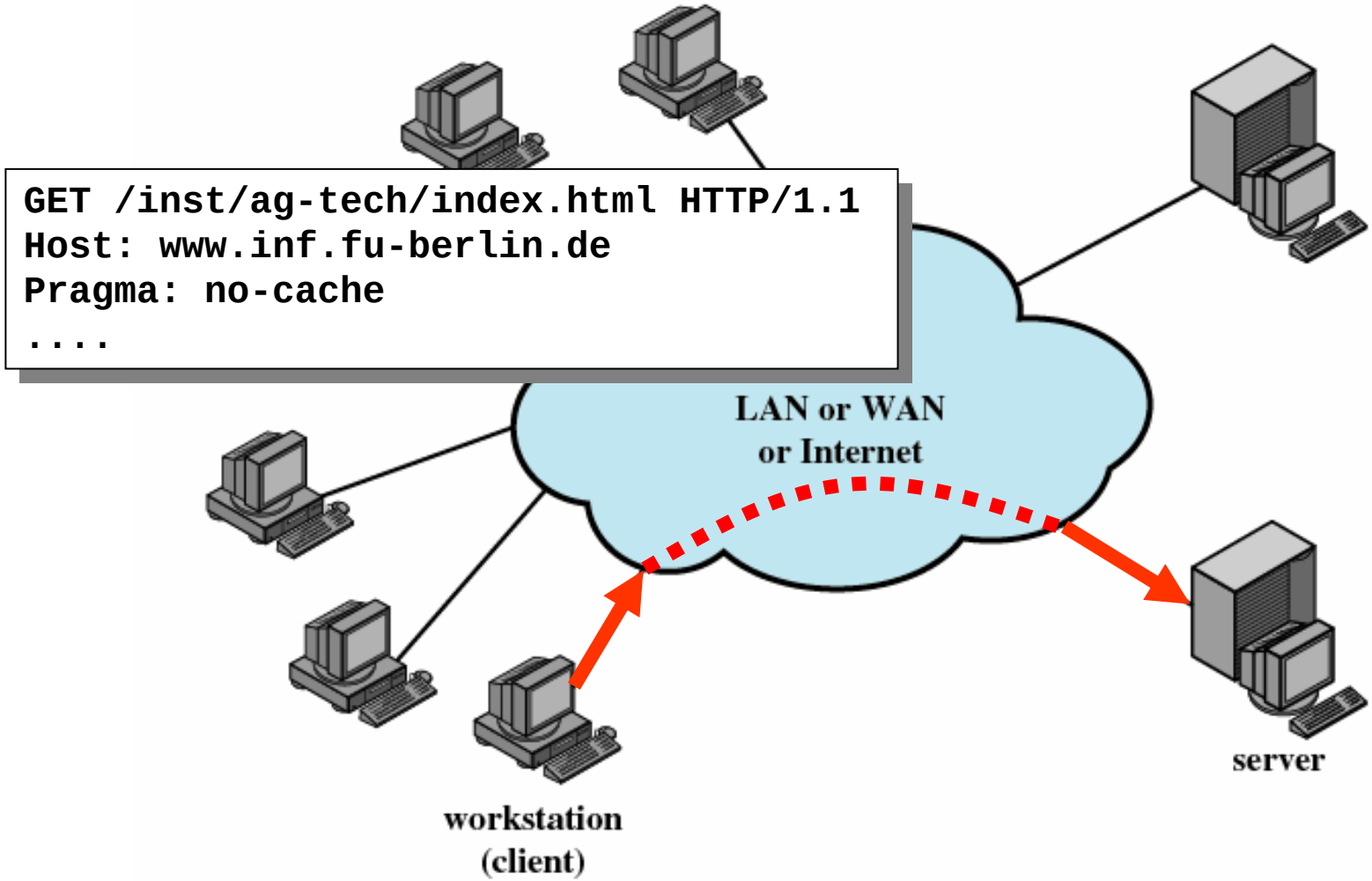


Figure 13.6 Socket System Calls for Connection-Oriented Protocol

Client/Server Communication



Layered Protocol Stack

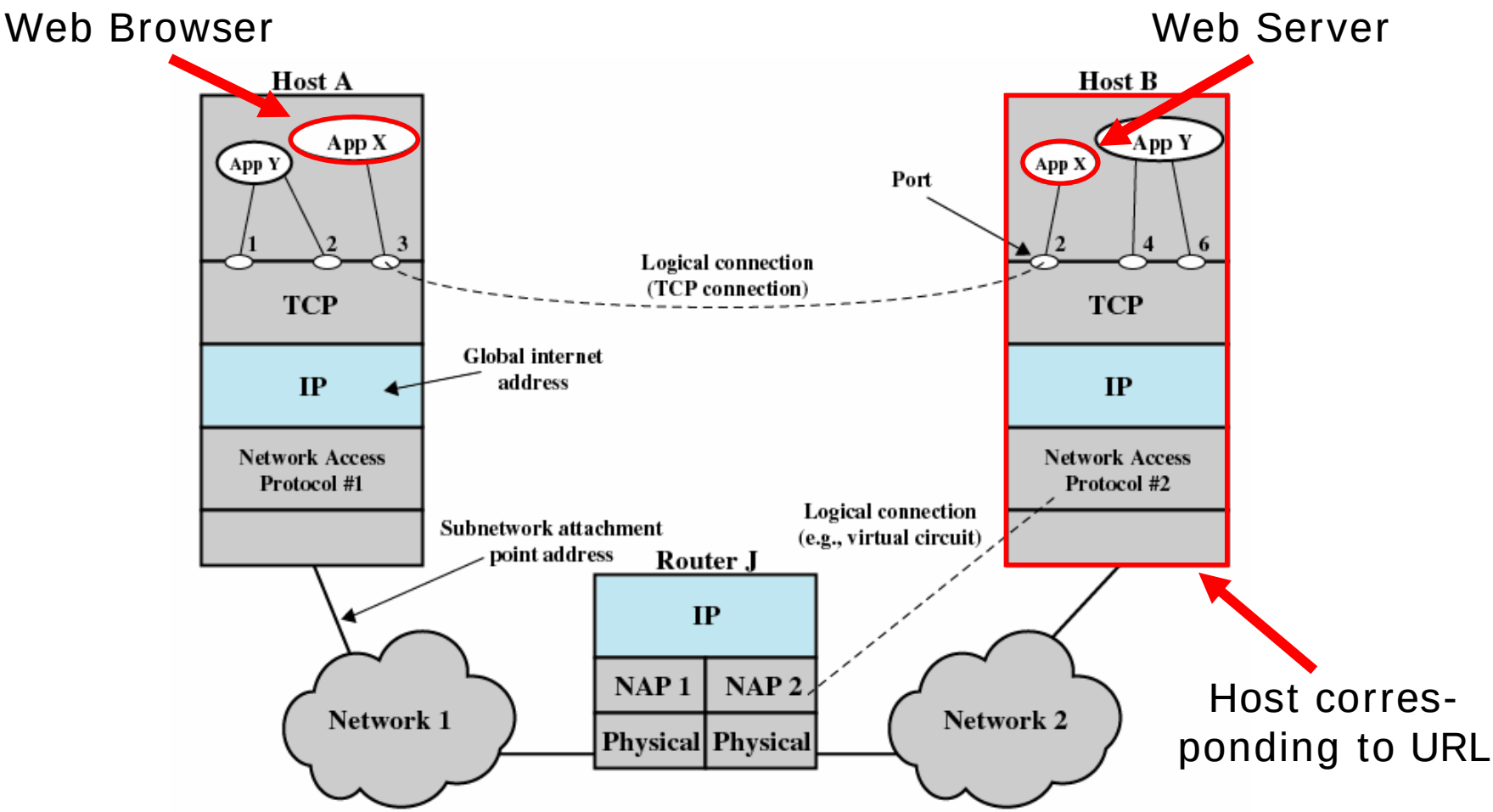
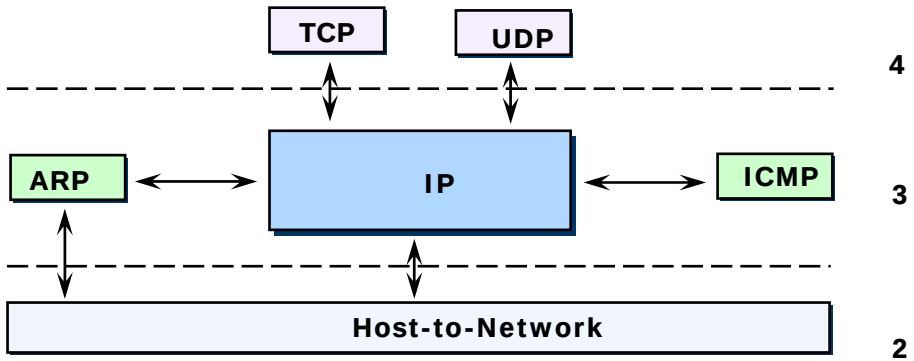


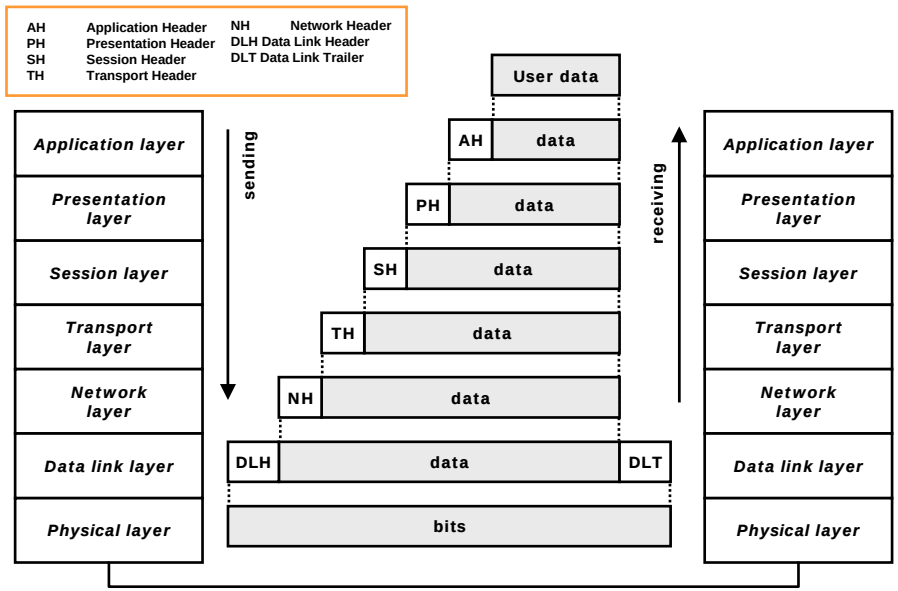
Figure 13.4 TCP/IP Concepts

Interaction Between Network Layers

- Layered protocol architecture
  - Each layer uses only services of layer directly below
  - Each layer provides services to layer directly above
  - ➢ Protocol independence
  - ➢ Modularity



- Data encapsulation
  - Lower layers treat upper layer packets as simple data
  - Headers contain control information for each layer
  - ➢ Repeated encapsulation causes overhead



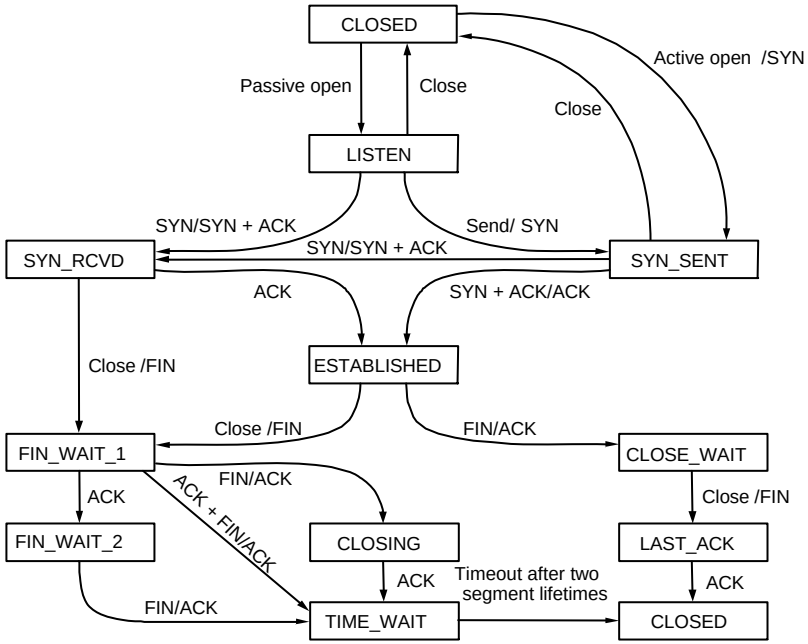
Uniform Resource Locator (URL)

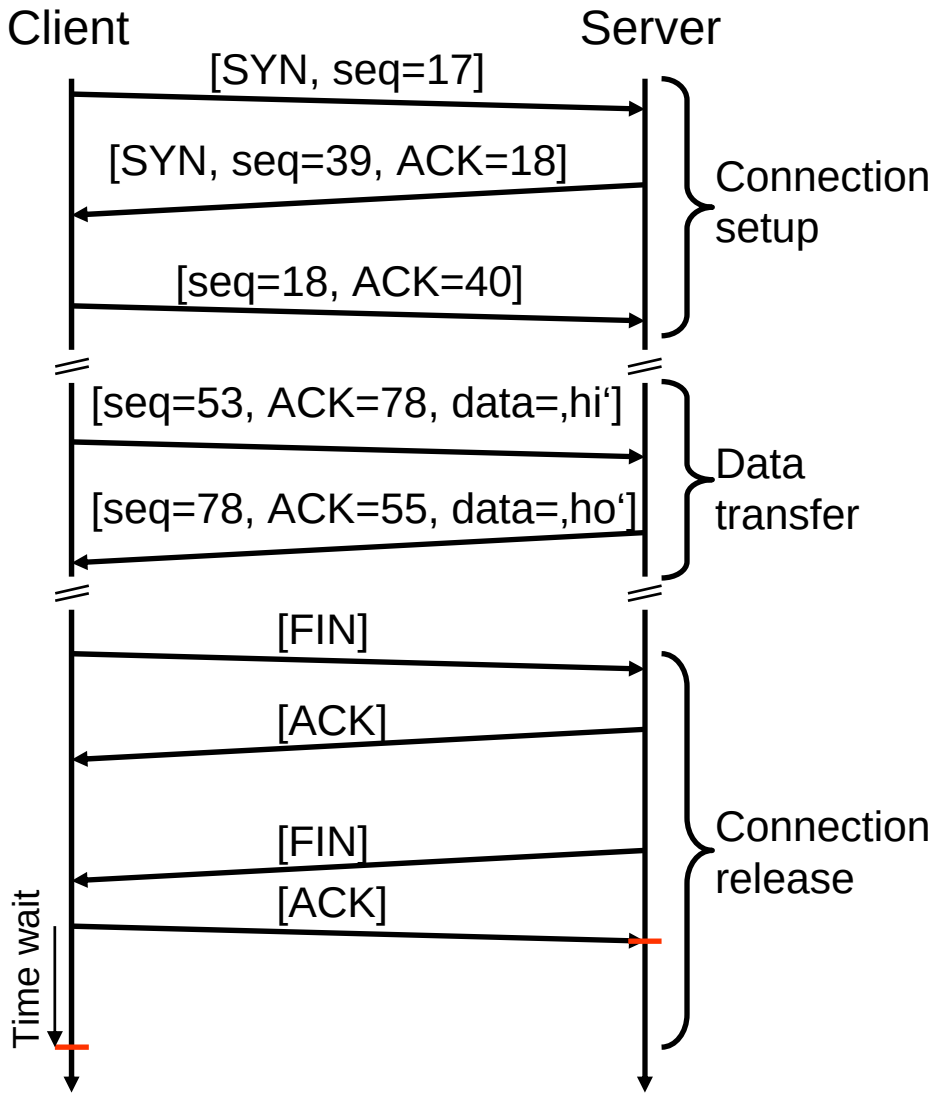
http://www.inf.fu-berlin.de/inst/ag-tech/index.html

- **http**: Hypertext Transfer Protocol (HTTP)
  - Protocol for accessing web pages and related content
  - Implies communication over port 80 (unless other port given in URL)
- **www.inf.fu-berlin.de**: Host name
  - Resolved to IP address via Domain Name System (DNS)
  - www.inf.fu-berlin.de -> 160.45.117.8
- **inst/ag-tech/index.html**: Local resource name
  - Protocol specific parameter
  - Handled by web server

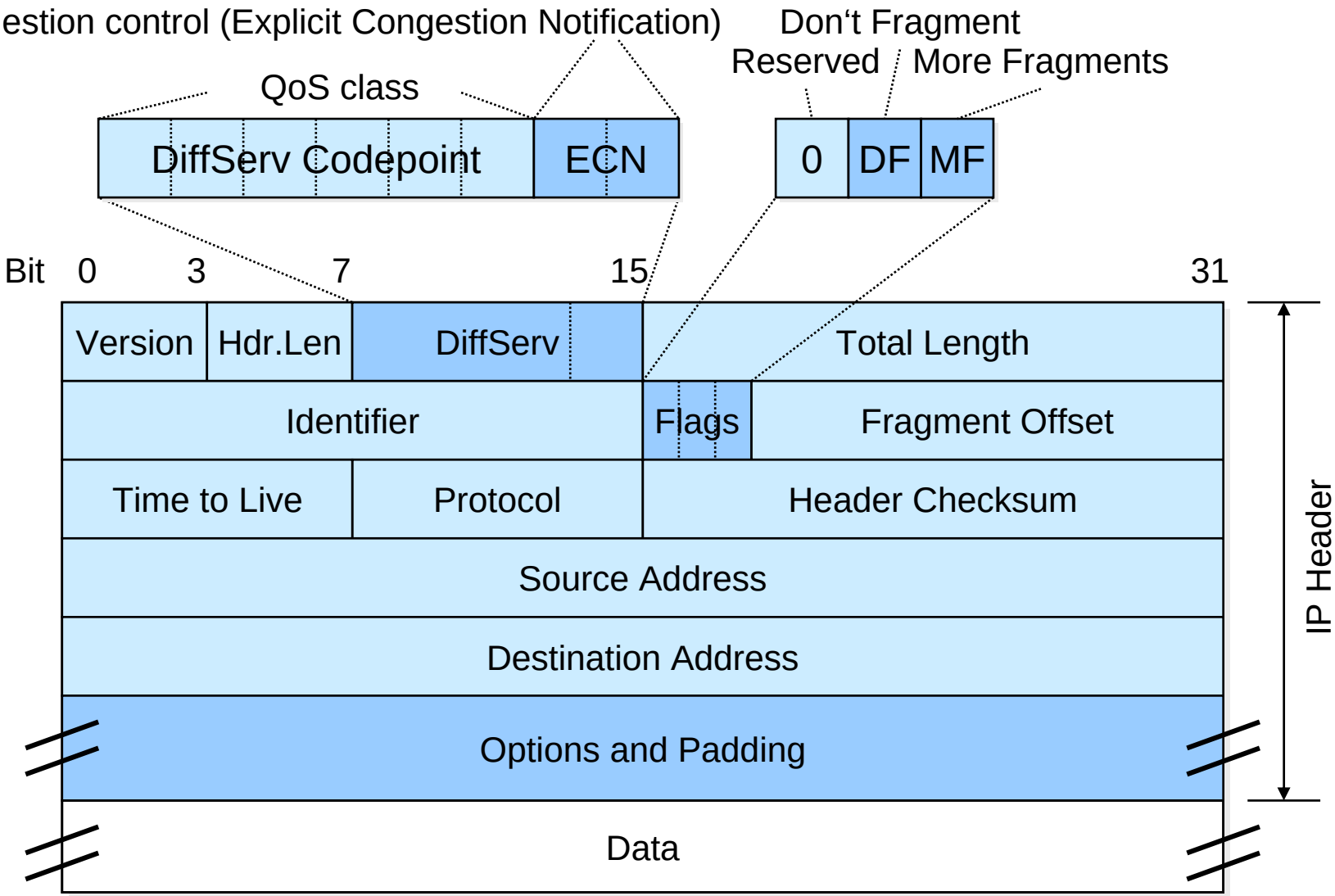
Connection Setup / Transport Layer

- Reliable end-to-end connection between processes
- Call to **connect()** initiates connection setup
  - TCP 3-way handshake
  - Connection parameters

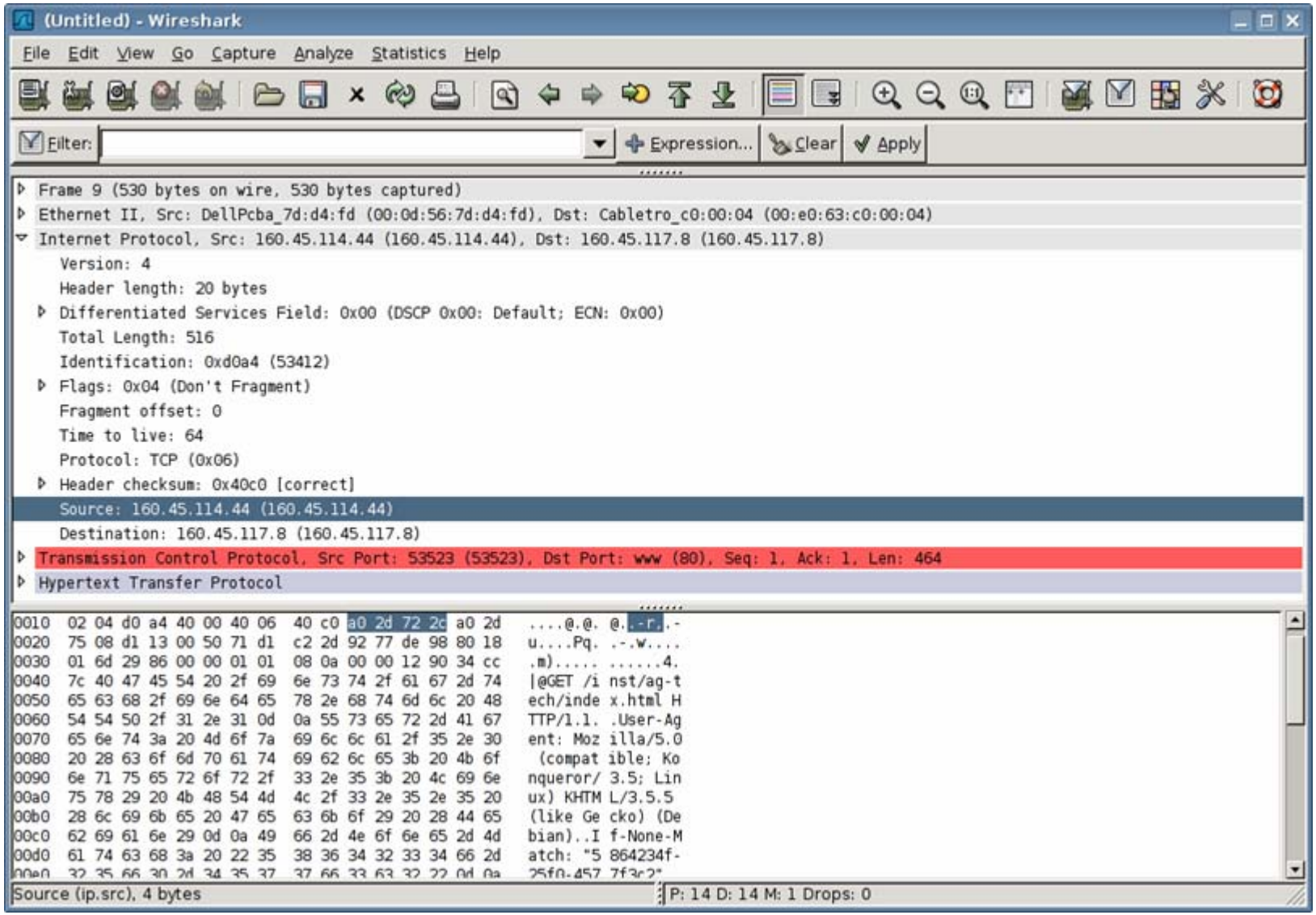




Structure of Network Layer IP-Packet

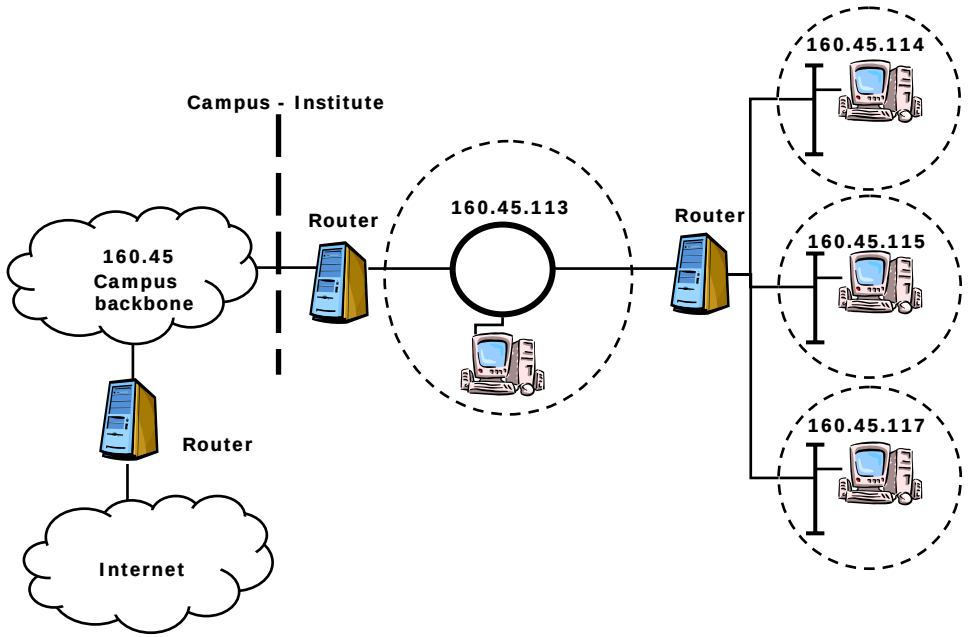


HTTP Data Encapsulation



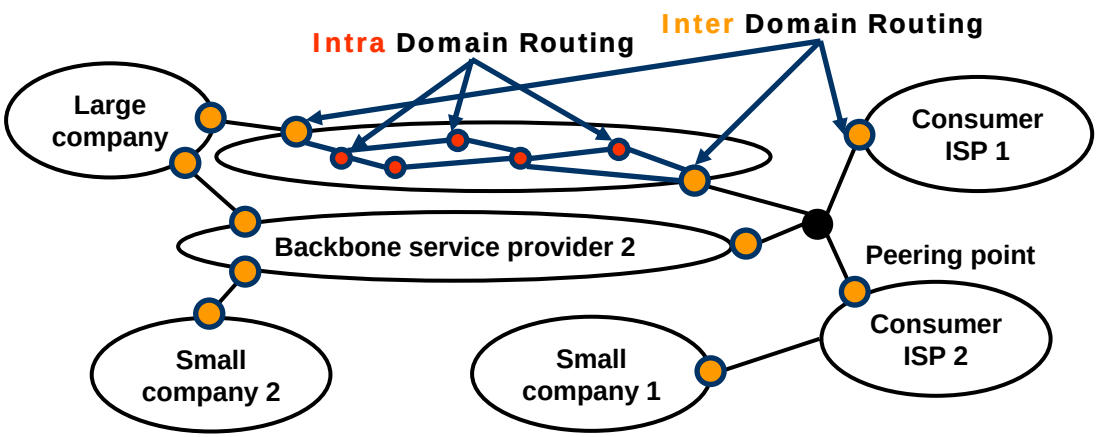
Network Layer Routing (Local Scope)

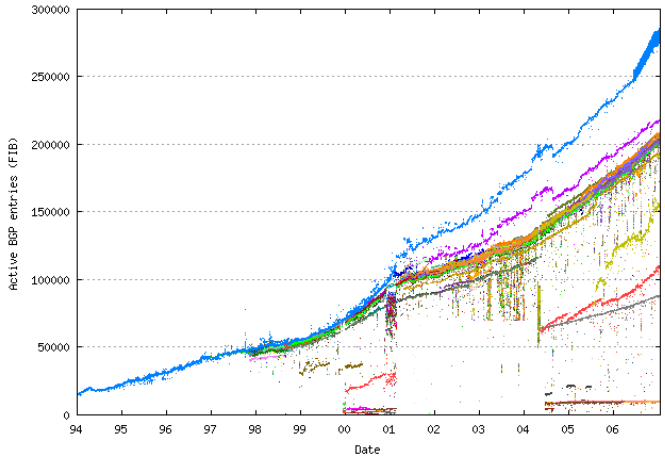
- Globally unique per host addressing
- Routers maintain tables of known networks
  - Optional route to default gateway
- Subnetting implements logical structure
  - Subnet mask builds hierarchy using host part of IP address
  - Limits broadcasts
  - More efficient routing
- Network topology may be part of security concept



Network Layer Routing (Global Scope)

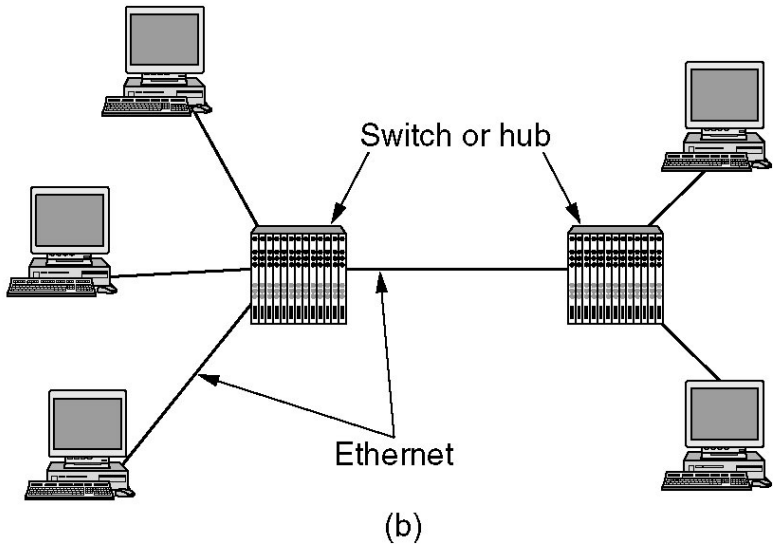
- Internet organized into autonomous systems (AS)
  - Commonly, one AS per major organization
  - Peering points to exchange data between ASs
- Intra-domain routing: OSPF, link state algorithm
- Inter-domain routing: BGPv4, distance vector protocol
  - ➢ May involve non-technical routing choices

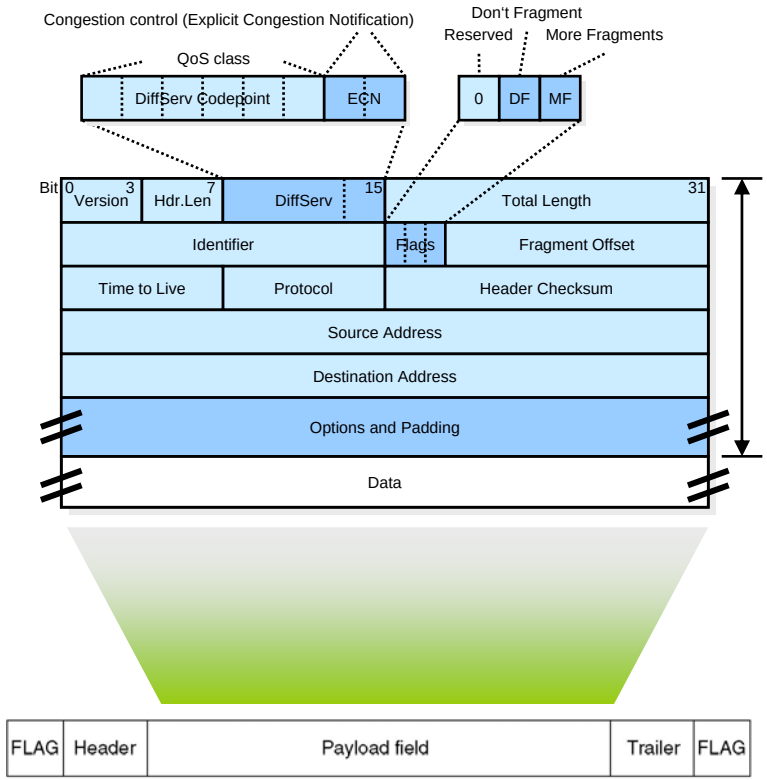




Data Link Layer Comm. (Local Scope)

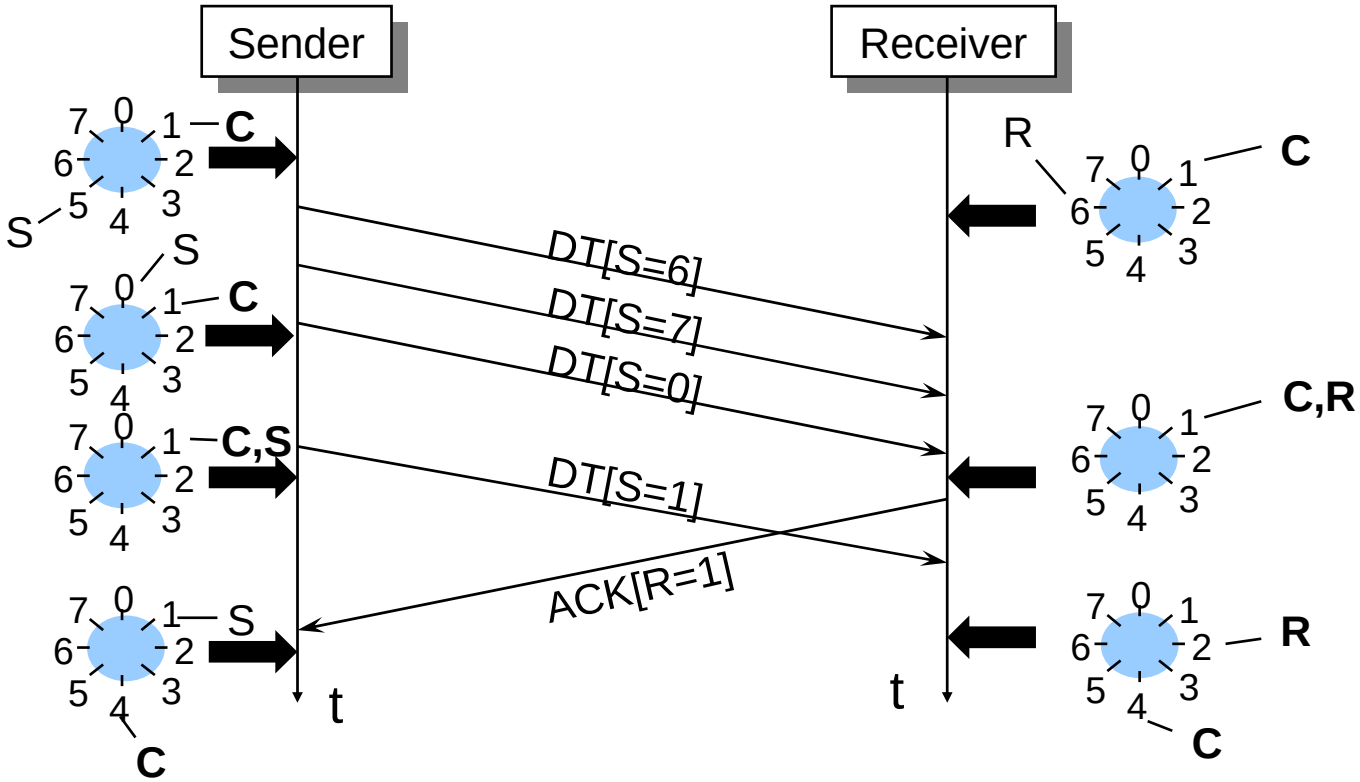
- Transparent communication between two directly connected nodes
- Services include: framing, error control, connection maintenance, acknow-ledgements, flow control



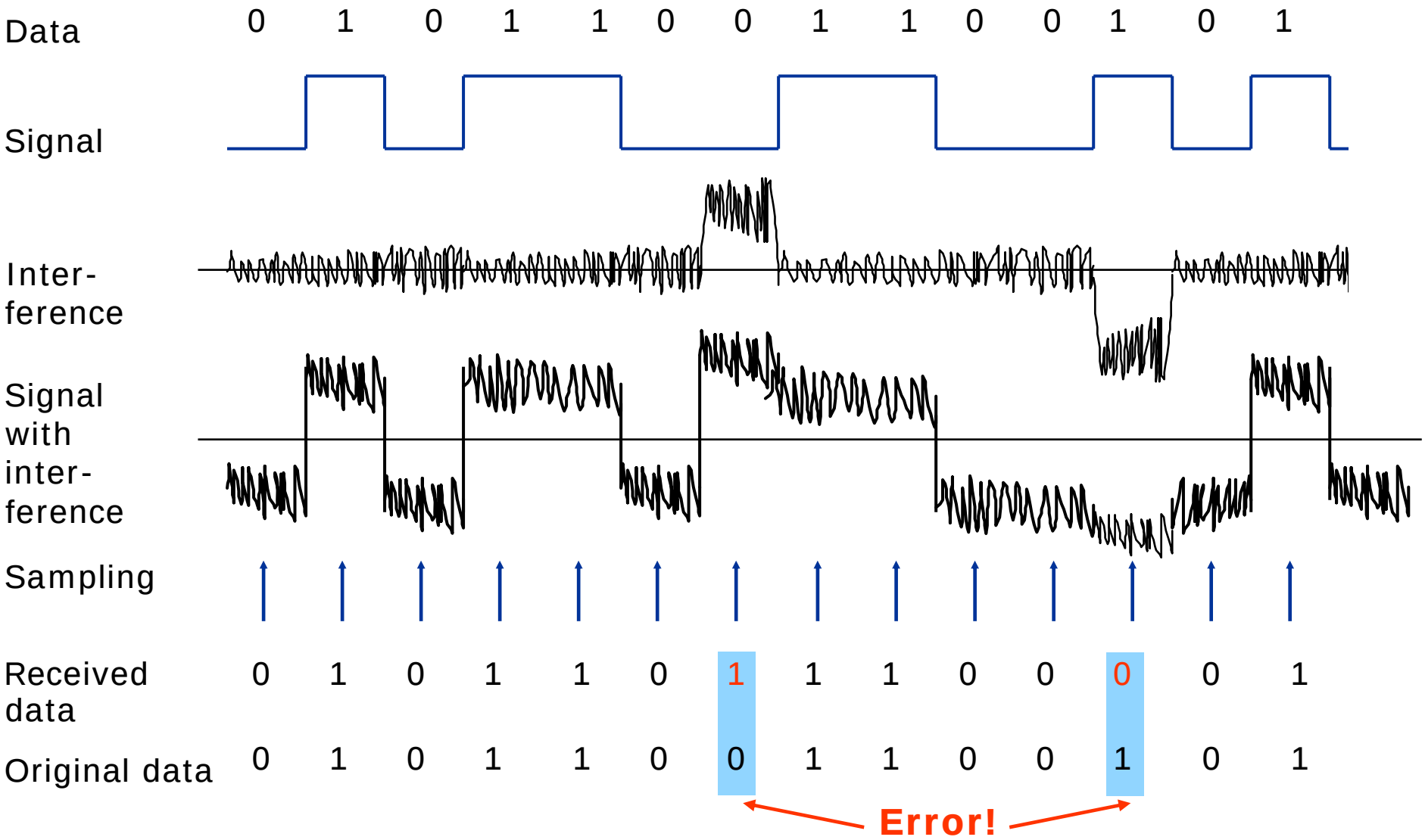


Flow Control

- Packets may be lost / Sender may flood receiver
- ➢ Need to acknowledge packets / control transmission rate
  - Credit-based schemes, e.g. sliding window



Errors During Transmission



Error Detection: Cyclic Redundancy Check (CRC)

- Reception of a correct bit sequence:

```
11 0011 1001 ÷ 1 1001 = 10 0001
11 001
00 0001 1001
      1 1001
      0 0000  = remainder
```

  - No remainder, thus the received bits *should* be error free

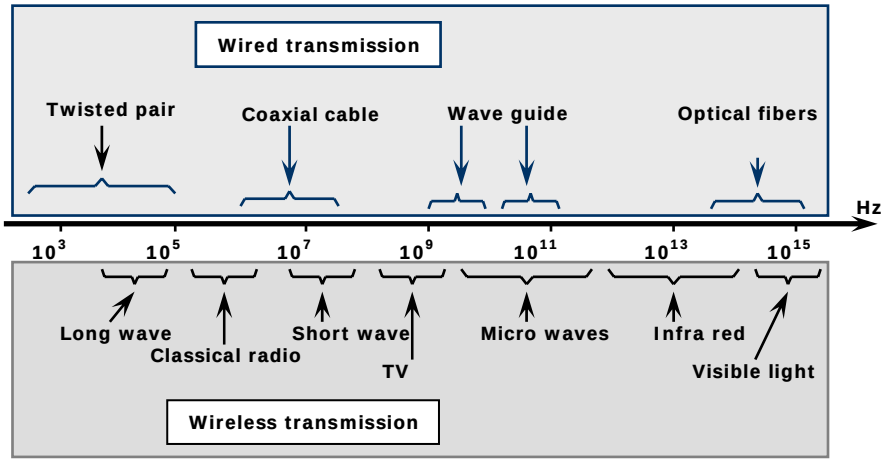
- Reception of a erroneous bit sequence:

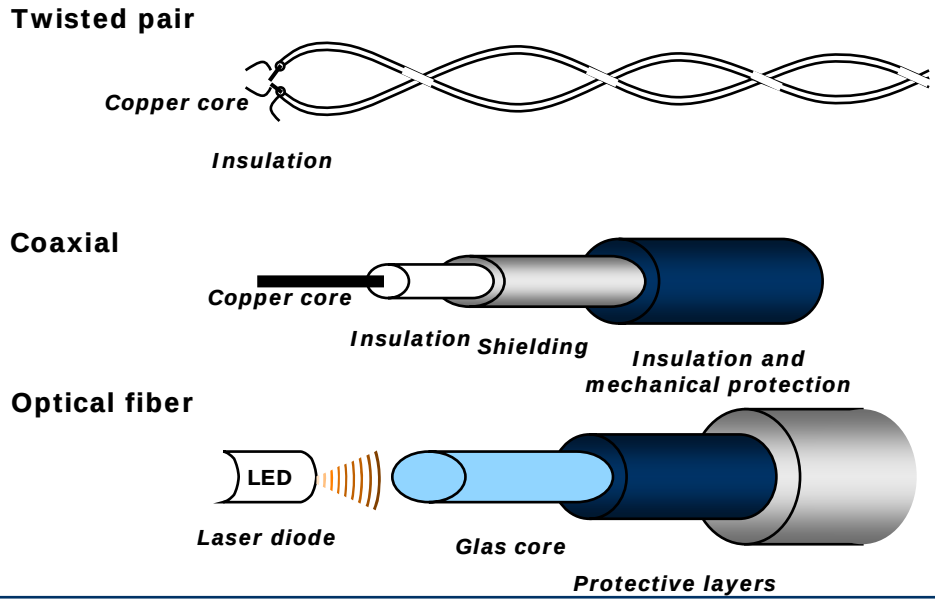
```
11 1111 1000 ÷ 1 1001 = 10 1001
11 001
00 1101 1
   1100 1
   0001 0000
      1 1001
      0 1001  = remainder ≠ 0
```

  - There is a remainder unequal 0, thus there was definitely a transmission error

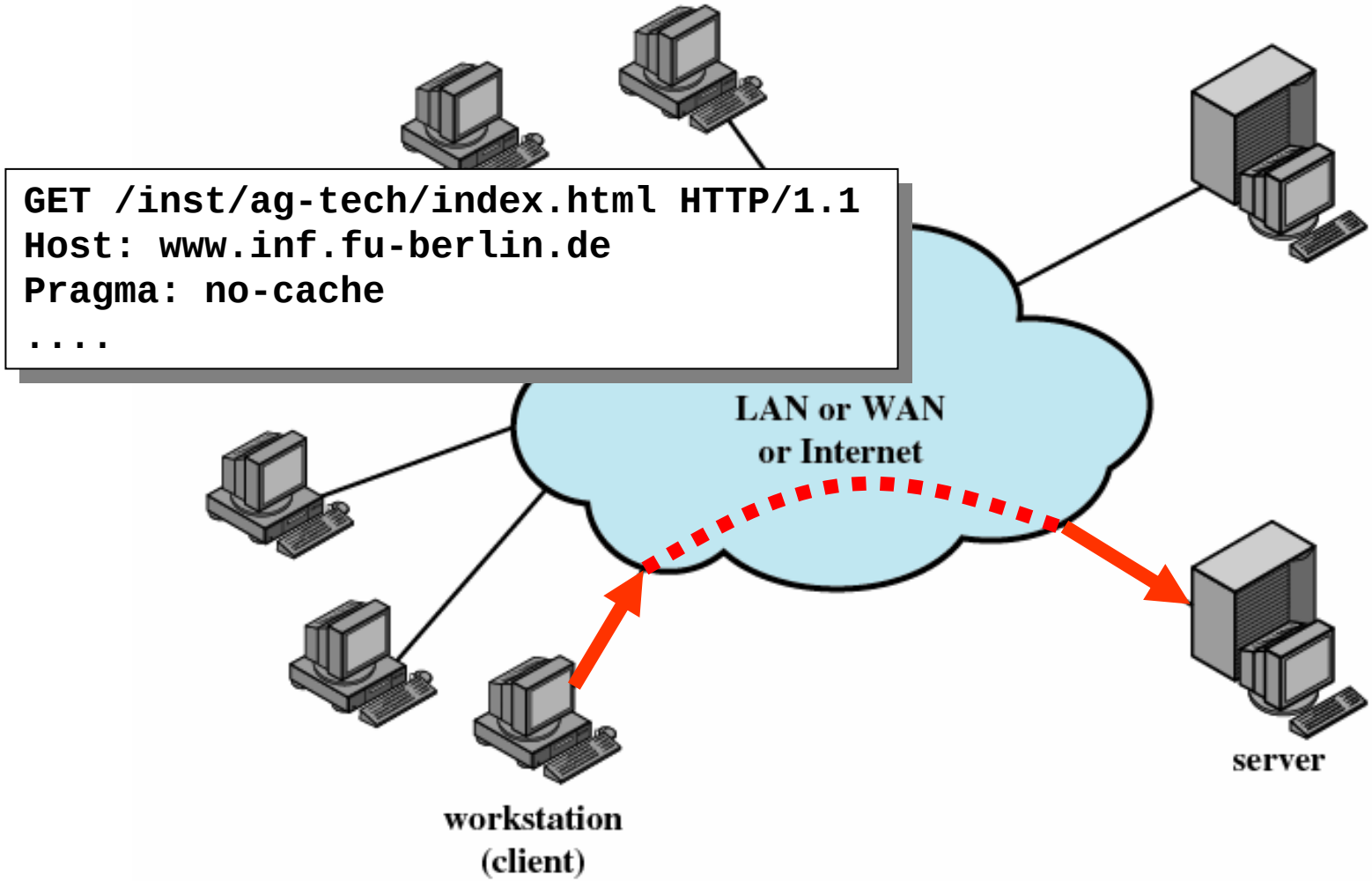
Physical Layer

- Packet / sequence of bits turned into physical signal
- Signal propagation depends on physical medium (limited bandwidth, attenuation, dispersion) and background noise
- Mapping between bits and (multi-valued) symbols
- Baseband transmission vs. modulation (broadband transmission)



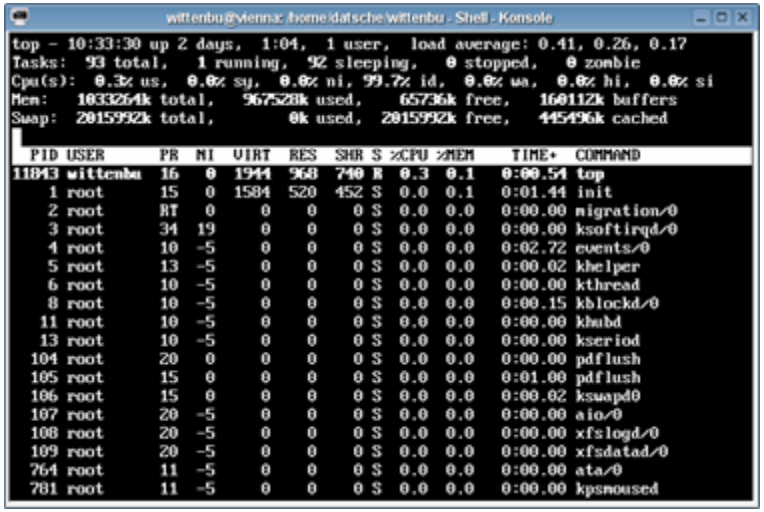


Client/Server Communication



At the server...

- Web server is one of many processes running locally



- Upon receiving packet, network interface controller (NIC) will raise interrupt
- Kernel will handle the packet and notify the web server process

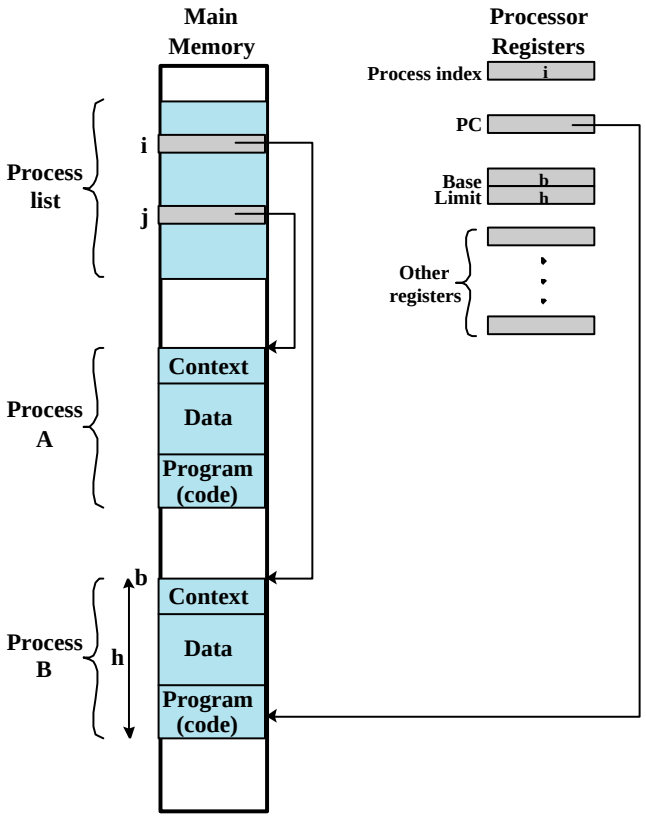


Figure 2.8 Typical Process Implementation

Processing of HTTP-GET Request

- Web server retrieves file **`inst/ag-tech/index.html`** from local file system
  - System calls to access secondary storage
  - Kernel maps file name to data layout on disk
- Web server sends data to client

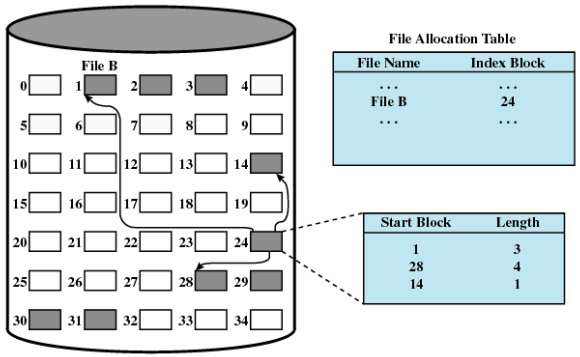


Figure 12.12 Indexed Allocation with Variable-Length Portions

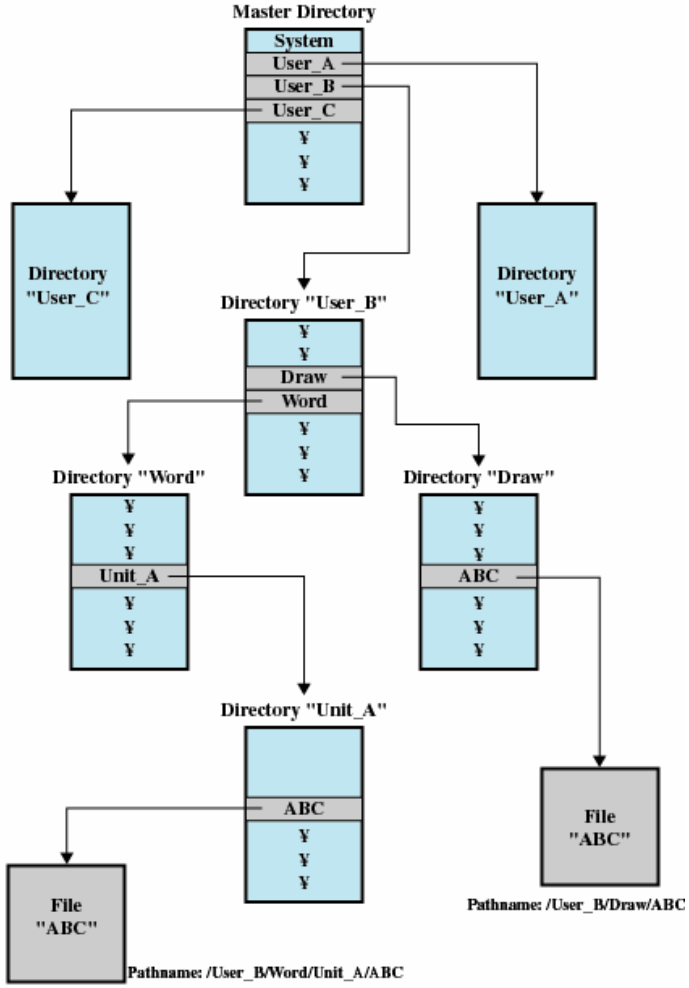
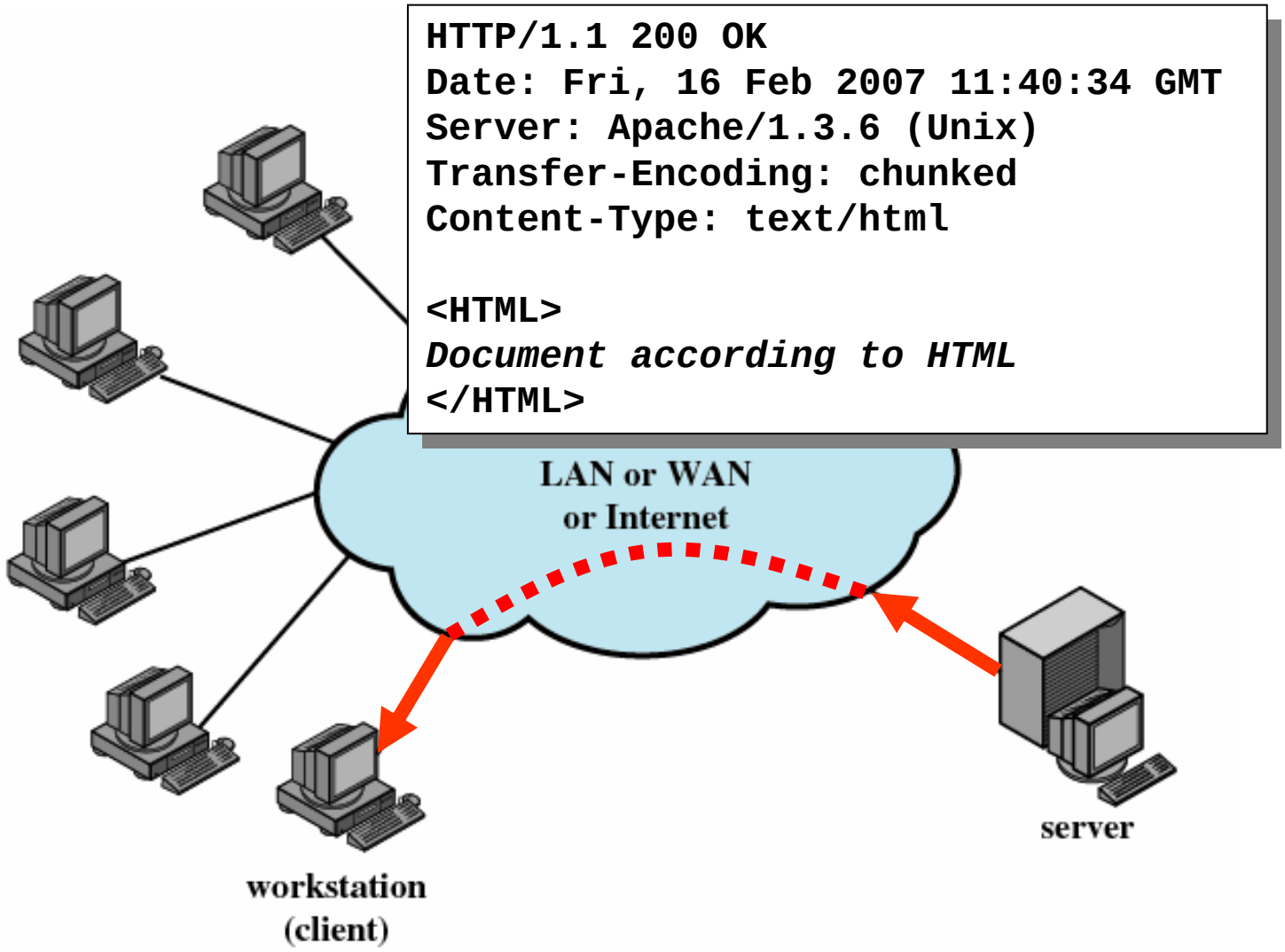


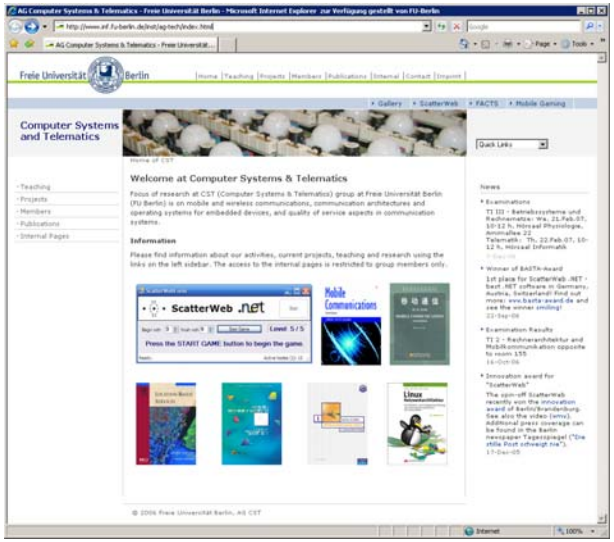
Figure 12.5 Example of Tree-Structured Directory

Server Replies to Client



Client Data Processing

- Client host receives packet
- Kernel hands data to web browser process
- Web browser renders page
  - May have to allocate memory in the process
- Finally, browser updates user interface via system call



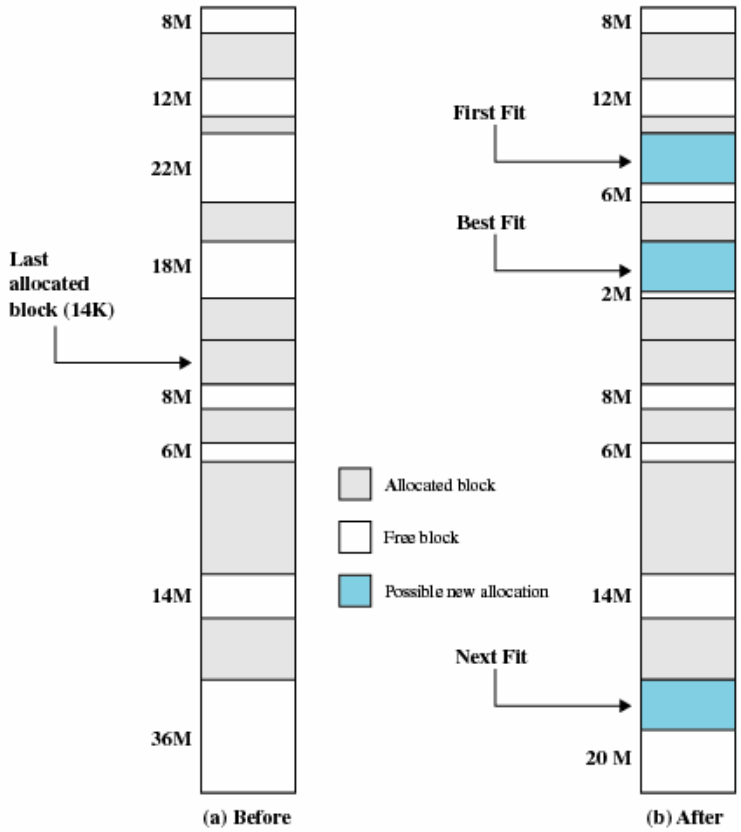
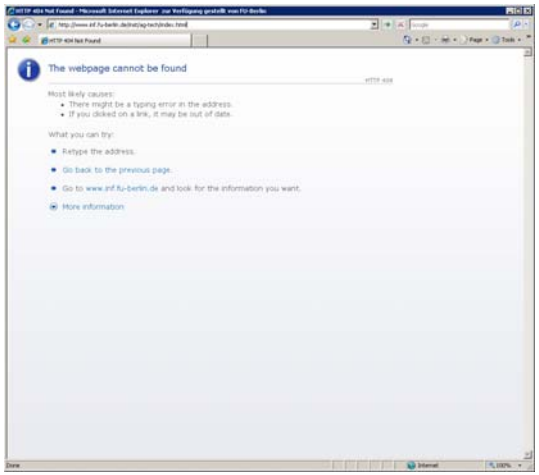
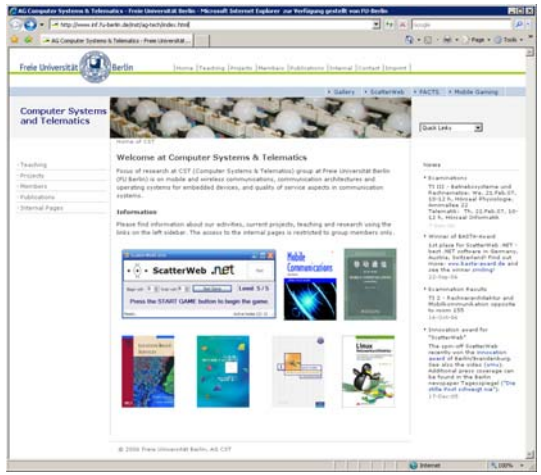


Figure 7.5 Example Memory Configuration Before and After Allocation of 16 Mbyte Block

A Comprehensive Example





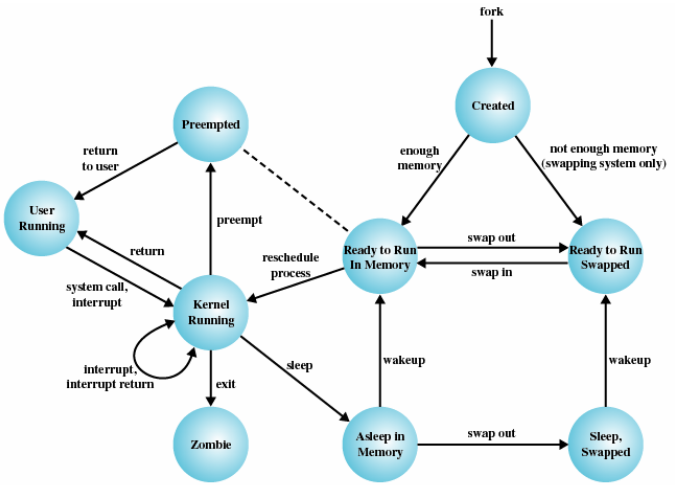
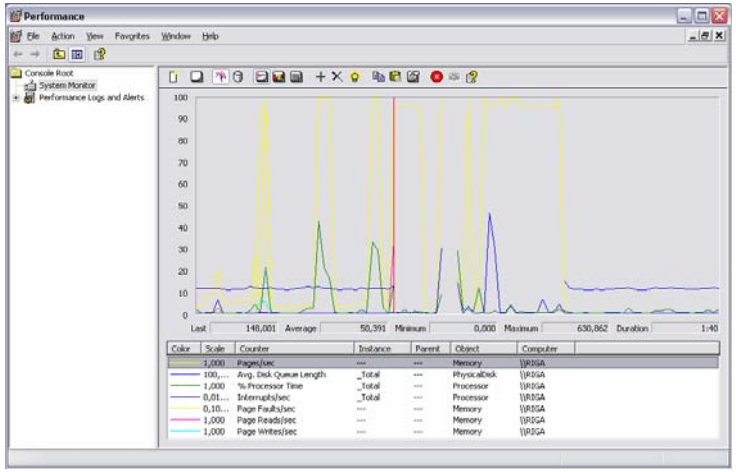


Figure 3.17 UNIX Process State Transition Diagram



Fin