3. Enabling Web and Software Technologies

3.1 Client / Server Architecture and the Internet

Client / Server Architecture, WWW, Internet Technologies

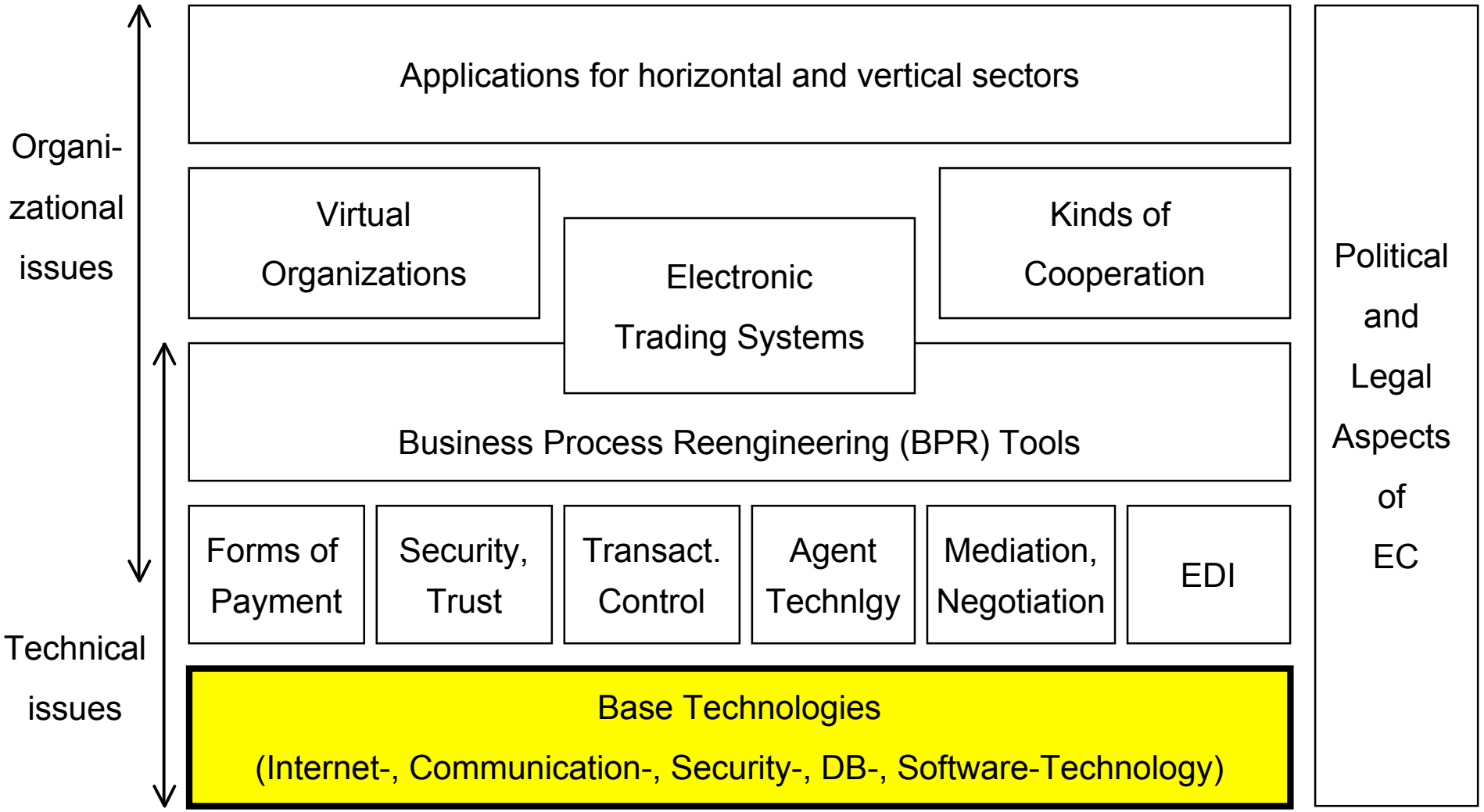
TCP/IP, Domain Naming Service Sockets, Future of the Internet (IPv6)

3.2 Internet Protocols and Internet Infrastructure

3.3 Multi-Tier Architectures

3.4 Platform Choices and Connectivity Options

ECommerce Reference Model



[MeTuLa99]

References & Further Reading


[Wilde99] Erik Wilde: World Wide Web: Technische Grundlagen, Springer-Verlag 1999, ISBN 3-540-64700-7


[CDK95] George Coulouris, Jean Dollimore, Tim Kindberg, Distributed Systems: Concepts and Design, Addison-Wesley 1995, ISBN 0-201-62433-8


[Pohlm00] Norbert Pohlmann, Firewall-Systeme: Sicherheit für Internet und Intranet, MIT Press 2000, 3-8266-4075-6


[Schm98] Klaus Schmeh: Safer Net, Kryptographie im Internet und Intranet, iX Edition, dpunkt Verlag 1998, ISBN 3-932588-23-1


[CommerceNet00] CommerceNet, eCommerce Resources, Internet Glossary, http://www.commerce.net/resources/glossary.html


[DCB00] Dot-Com Builder, Sun Developer Connection, Sun Microsystems, http://dcb.sun.com/

World Wide Web

Motivation: Developing a global distributed hypermedia system.

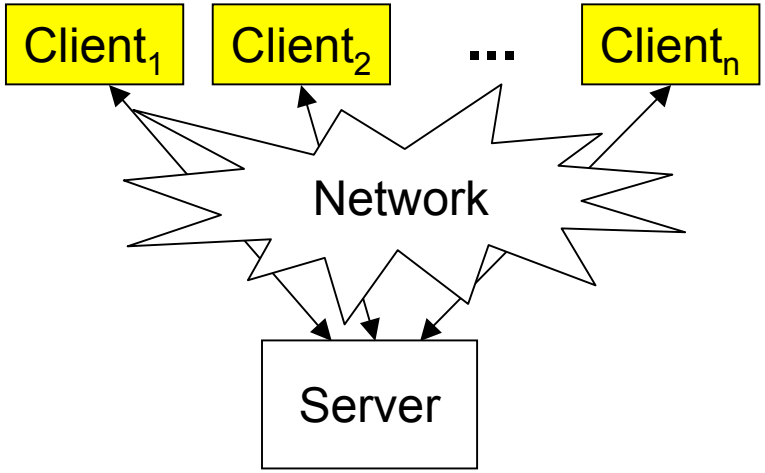
- Started 1989 by a research paper issued by Tim Berners-Lee who worked at the CERN.
- 1993: First usable browser (MOSAIC) issued.
- 1994: Foundation of World Wide Web Consortium (W3C).
- W3C then started developing HTML, HTTP and Style Sheets.

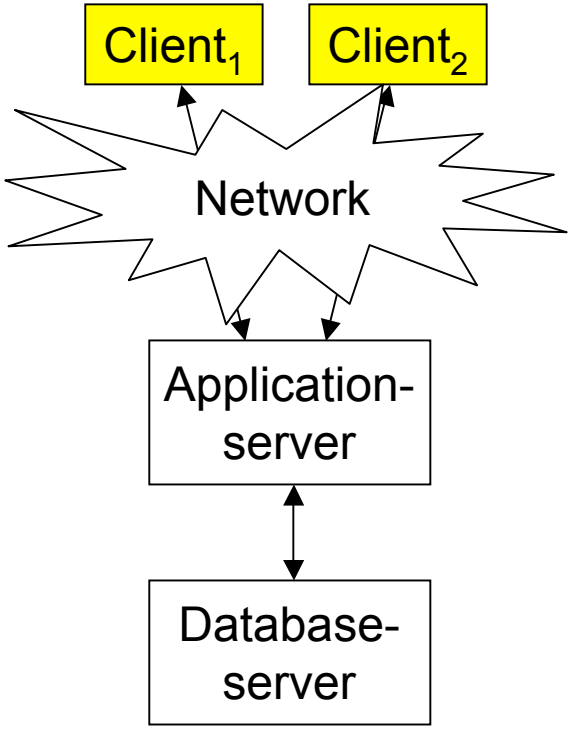
Internet Technologies

Relevant Internet Technologies

- Distributed Client / Server Architecture
- World Wide Web (WWW, The Web)
- Domain Name System (DNS)
- TCP/IP, Sockets

Distributed Architectures: Client / Server





Example: 2–tier Architecture:

- Browser
- Web-Server

Example: 3-tier architecture:

- Employee / customer PC
- Enterprise department server
- Enterprise database

Distribution Abstraction

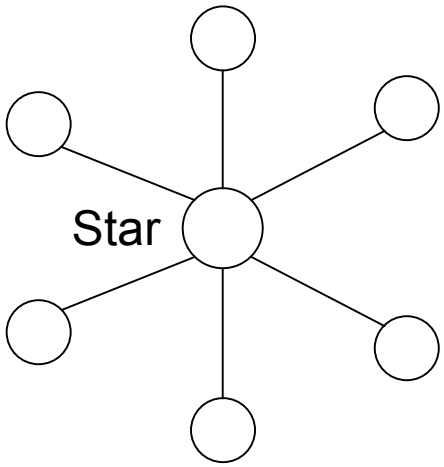
Systems on the Internet are modeled and realized in a way that allows for equal access to local and remote resources.

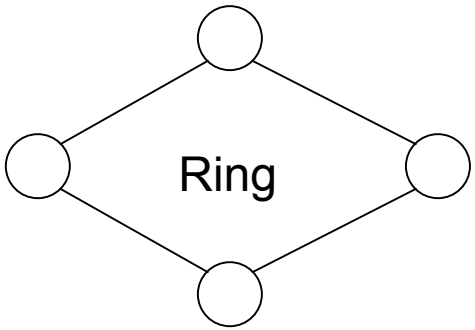
Examples

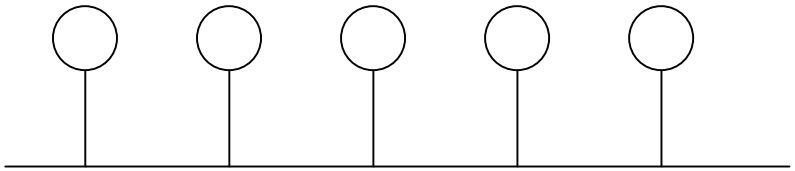
- Network File System (NFS)
- X-Window System
- Web - Protocol (HTTP)

Distribution is usually only noticed in case of failure (independent system failures)

Abstracting from a net topology







Bus

Client / Server Architecture (1)

Definition

A **server** acts as a resource manager for a collection of resources of a given type [CDK95].

Definition

A **client** performs a task that requires access to to some shared hardware and software resources [CDK95].

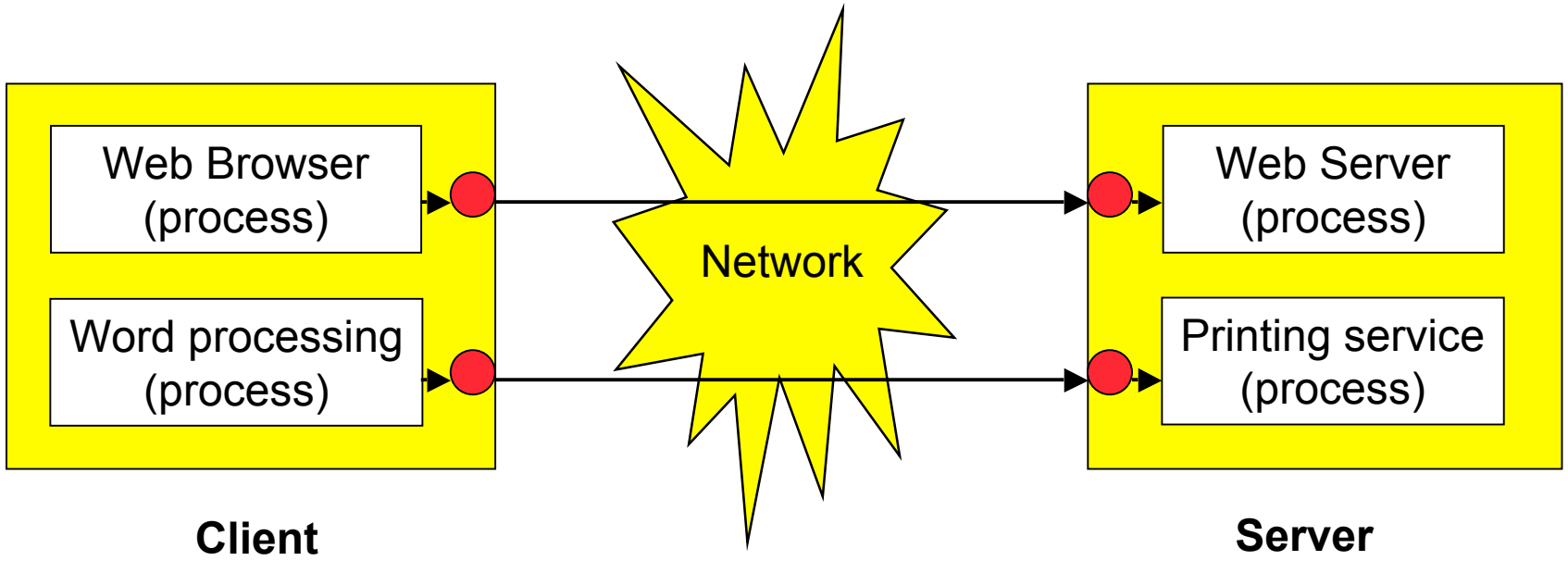
In the client / server model, all resources are held by servers. Clients issue *requests* whenever they need to access one of the resources [CDK95].

Examples

- A web server (HTTP daemon, http) manages a collection of web (HTML) pages
- A web client (HTTP client, a web browser) requests web pages.

Client / Server Architecture (2)

- The *performer* offers *services* (online shop, current time, stock markets) or resources (printers, files, ...) on the server.
- The *customer* uses the services, e.g., buys at online shops, requests the current time, or prints a document.



Internet and WWW

Definition

The **Internet** is the entirety of all connected computers that use the package of internet protocols at their network systems‘ topmost layer. The collection of internet protocols implements a packet-oriented Wide Area *Network for connecting networks* of diverse protocols and different connection characteristics.

Definition

The **World Wide Web** (WWW) is a distributed hypermedia system that relies on some of the internet‘s services. Most important are the naming service provided by the Domain Nam Service (DNS) and the - quite - reliable connection-oriented transmission service provided by the Transmission Control Protocol (TCP) [Wilde99].

Internet Addressing

Global identification of computers

- Local naming within domains:

```
sts.tu-harburg.de
tu-harburg.de
hamburg.de
marinfo.net
```

  structured logically, stable

- Non-ambiguous Internet addresses

```
134.100.11.156
```

  compact, efficient, limited (32 bit)

Global identification of further resources (persons, information)

- Users (by email addresses)

```
pa.hupe@tu-harburg.de
```

- Services (by URL)

```
http://www.tu-harburg.de
ftp://ftp.uni-hamburg.de
```

- Documents (by URL)

```
http://www.sts.tu-harburg.de/slides/1998-deutsch/10-98-Matt.ppt
```

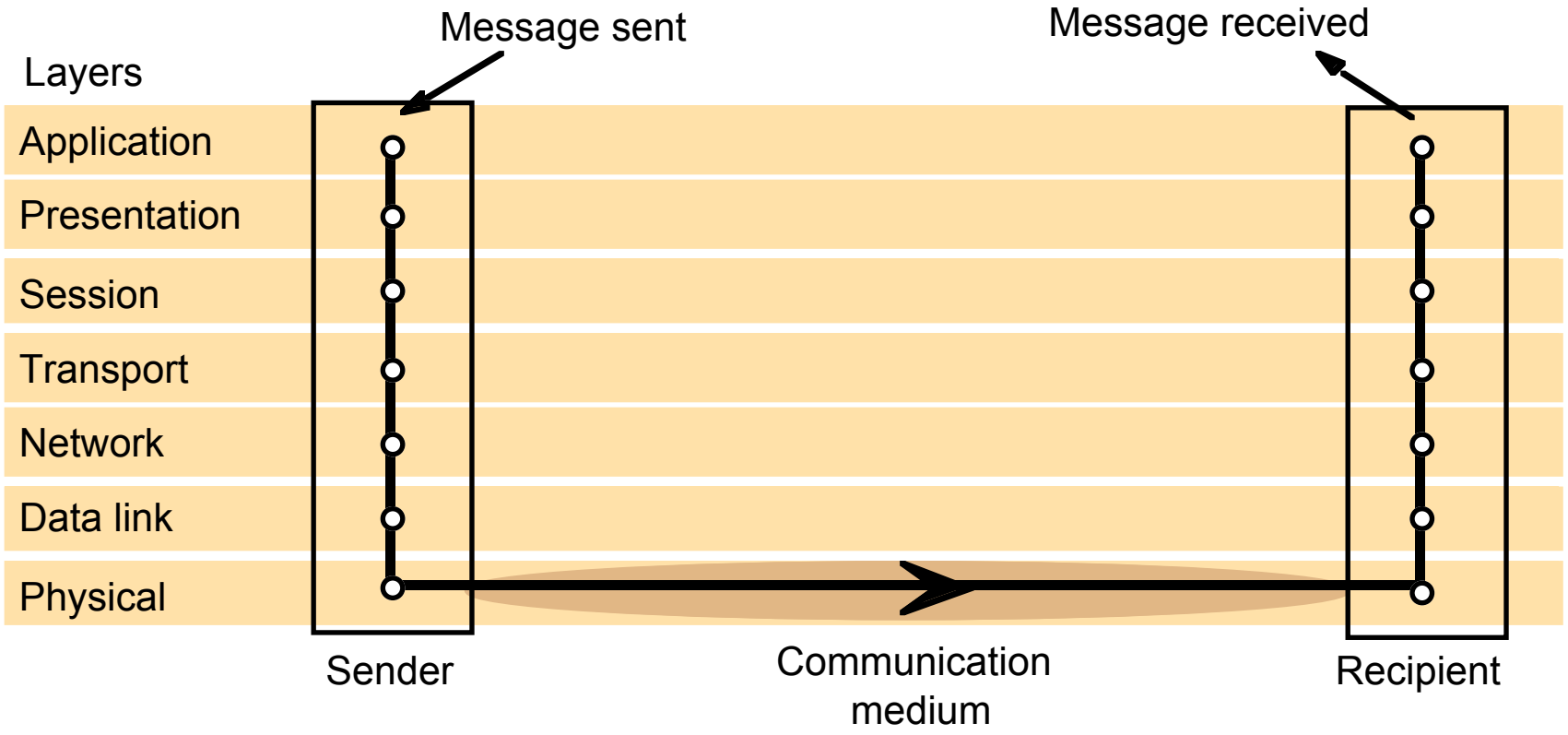
- Data, Information, Concepts, Knowledge, ...

Connecting to the Internet

- **Dial-up Connection:** Computers that are serving only as clients need not be connected to the internet permanently. Computers connected to the internet via a dial-up connection usually are assigned a dynamic IP address by their ISP (Internet Service Provider).

- **Leased Line Connection**: Servers must always be connected to the internet. No dial-up connection via modem is used, but a leased line. Costs vary depending on bandwidth, distance and supplementary services.

Open Systems Interconnection (1)



Open Systems Interconnection (2)

Layer	*Description*	*Examples*
Application	Protocols that are designed to meet the communication requirements of specific applications, often defining the interface to a service.	HTTP,FTP, SMTP, CORBA IIOP
Presentation	Protocols at this level transmit data in a network representation that is independent of the representations used in individual computers, which may differ. Encryption is also performed in this layer, if required.	Secure Sockets (SSL),CORBA Data Rep.
Session	At this level reliability and adaptation are performed, such as detection of failures and automatic recovery.	
Transport	This is the lowest level at which messages (rather than packets) are handled. Messages are addressed to communication ports attached to processes, Protocols in this layer may be connection-oriented or connectionless.	TCP, UDP
Network	Transfers data packets between computers in a specific network. In a WAN or an internetwork this involves the generation of a route passing through routers. In a single LAN no routing is required.	IP, ATM virtual circuits
Data link	Responsible for transmission of packets between nodes that are directly connected by a physical link. In a WAN transmission is between pairs of routers or between routers and hosts. In a LAN it is between any pair of hosts.	Ethernet MAC, ATM cell transfer, PPP
Physical	The circuits and hardware that drive the network. It transmits sequences of binary data by analogue signalling, using amplitude or frequency modulation of electrical signals (on cable circuits), light signals (on fibre optic circuits) or other electromagnetic signals (on radio and microwave circuits).	Ethernet base- band signalling, ISDN

Internet Protocol, IP (v4)

The **Internet Protocol (v4)** is connection-less, datagram-oriented, packet-oriented. Packets in IP may be sent several times, lost, and reordered.

Disadvantages of IPv4:

- Address space is limited to 4 billion hosts in 16,7 million networks. The limitation is severed furthermore by classification of IP addresses into A-. B-, C-, D- and E-Class nets. The net classes define the ratio of subnets (e.g. enterprise networks, university networks) to hosts in the subnets.

  Example: The TUHH has a Class B network (Class B = 16 bit for network prefix and 16 bit for host identification). TUHH hosts are in the IP range 134.28.x.x).

- No resource (bandwidth) reservation (for time-critical data transmissions as audio and video).
- Missing support for mobile servers. Mobile servers change IP address every time they connect to the internet.

Solutions

- IP next generation: New protocols for the internet. Most important of these protocols is IPv6.

Problem:

- Migration & Upward Compatibility: All hosts between the client and the server have to support IPv6

IP Addresses and Ports

Definition

The IP protocol defines **IP addresses**. An IP address specifies a single computer. A computer can have several IP addresses, depending on its network connection (modem, network card, multiple network cards, …).
An IP address is 32 bit long and usually written as 4 8 bit numbers separated by periods. (Example: 134.28.70.1).

Definition

A **port** is an *endpoint to a logical connection* on a computer. Ports are used by applications to transfer information through the logical connection. Every computer has 65536 (2^{16}) ports.

Some *well-known* port numbers are associated with well-known services (such as FTP, HTTP) that use specific higher-level protocols.

Examples:

Server programs are assigned to fixed, well-known ports. Example: A server serving the HTTP protocol (usually referred to as a web server) is based on port 80, FTP on ports 20 and 21. HTTP and FTP clients are assigned a dynamic port number.

Naming (1): IP Addresses and Domain Names

Every computer on the internet is identified by one or many IP addresses. Computers can be identified using their IP address, e.g., 134.28.70.1.

Easier and more convenient are **domain names** (e.g., www.sts.tu-harburg.de). Computer names on the internet follow the Domain Name System (DNS) format.

The **Domain Name System** (DNS) is a global naming service that translates names into IP addresses. Example: www.sts.tu-harburg.de is translated into 134.28.70.1.

Advantages:

- Ease of use (for humans): www.yahoo.com is more memorizable than 216.32.74.52.
- When moving the web server (e.g., to a computer with better performance), only the DNS lookup entry needs to be changed.

Naming (2): Domain Name Format

Domain names are structured *hierarchically* . Each domain name consists of domains that are separated by periods. Domain names are read from right to left. Example: www.sts.tu-harburg.de.

Top-Level Domains (*TLD*) are defined at the topmost level. Top level domains can be

- **Country-Code Top-Level Domains** (*ccTLD*). Examples: de, fr, ch, etc.
- **Generic Top-Level Domain** (*gTLD*). Examples: com, edu, org, net, mil, gov.

Top-level domains are issued by the *ICANN* (Internet Corporation for Assigned Names and Numbers). Currently, new top-level domains are being issued (e.g., biz, info).

Further structuring of domain names is done by the organization assigned to the domain. Examples:

- TLDs: *InterNIC* for North America TLDs (.com, .edu, .org), DENIC for German TLD (.de).
- Corporate Domains: *Microsoft* (microsoft.com), *TUHH* (tu-harburg.de).

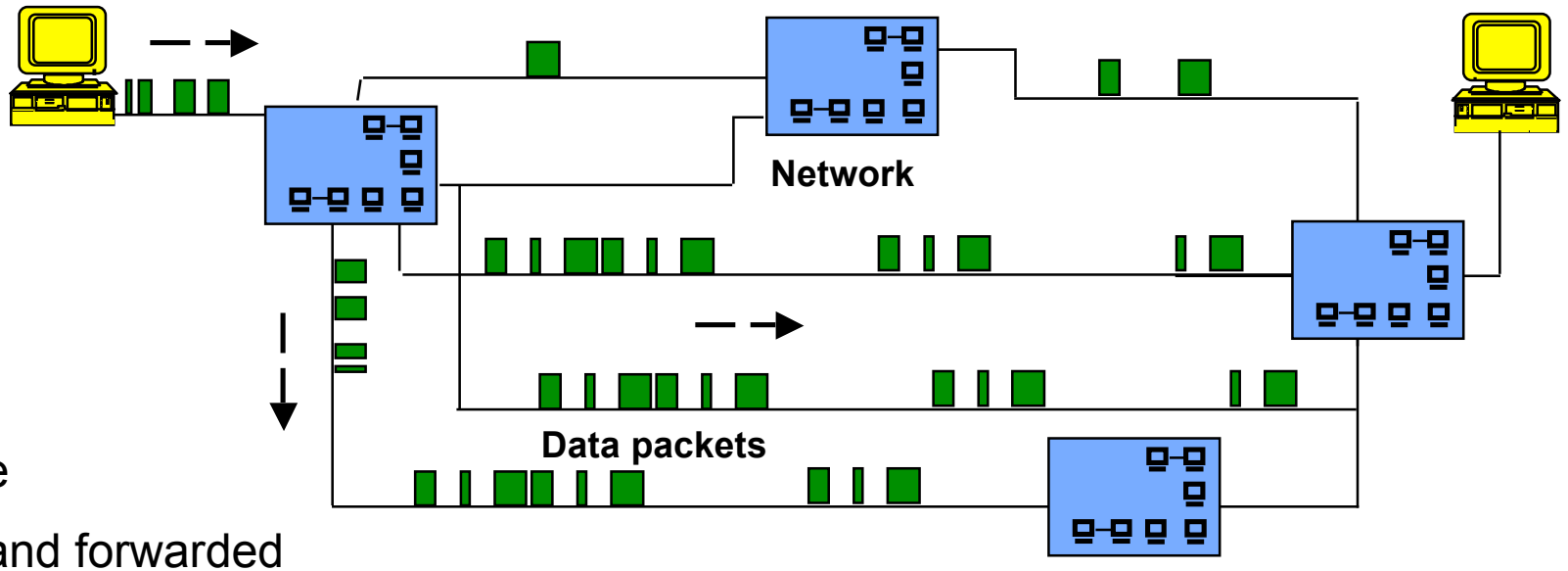
Transmission Control Protocol, TCP

Characteristics

TCP is synchronous, finite buffered, reliable, bytestream-oriented, connection-oriented, indirect addressing, bi-directional, untyped. TCP lies on top of IP.

A TCP address is an IP address plus a port (Example: 134.28.70:80).

TCP achieves *reliable connections* by sorting IP packets and re-requesting lost packets.



Packets are

- stored and forwarded
- redirected if required

Successful Communication Abstraction

Properties of a successful communication abstraction:

- Expressiveness
- Efficiency
- Availability on different hardware platforms (PC, Mac, Sun,...)
- Broad market support

We will introduce one successful communication abstraction: **Sockets**.

- Sockets are de-facto standard for *inter-process communication (IPC)* in a network.
- 1982 introduced in BSD 4.2-Unix; which led to the name *Berkeley Sockets*.
- Socket implementations are available for most Operating Systems, e.g., Unix, Windows (NT/2000 and 98/ME), MacOS, AmigaOS, BeOS, etc.

Sockets (1)

A **socket** is a network communication endpoint. It abstracts from the IP address, the protocol and the port number.

There are active sockets and passive sockets:

• An **active socket** is connected to a remote active socket via an open data connection. Closing the connection closes and removes the active sockets at each endpoint.

• A **passive socket** is not connected, but rather waits for incoming connection, which will establish a new active connection (create an active socket to the socket from which requested the connection).

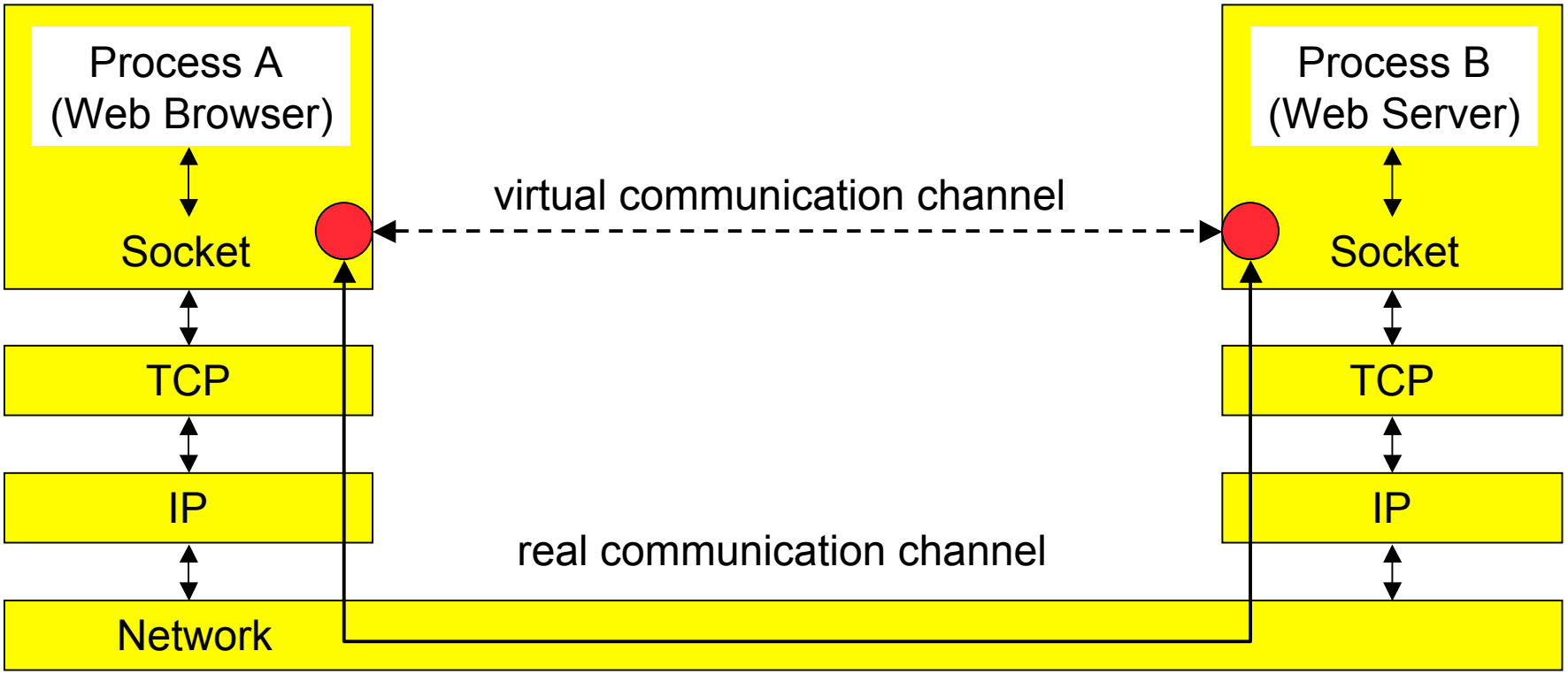
A socket is *associated with a port*. Each port can have a single passive socket, awaiting incoming connections, and multiple active sockets, each corresponding to an open connection on the port.

Several distributed systems are based on sockets, e.g.,

- Network File System (NFS),
- X-Window System,
- Remote Procedure Calls (RPC),
- DCE (Distributed Computing Environment),
- CORBA (Common Object Request Broker Architecture) Middleware

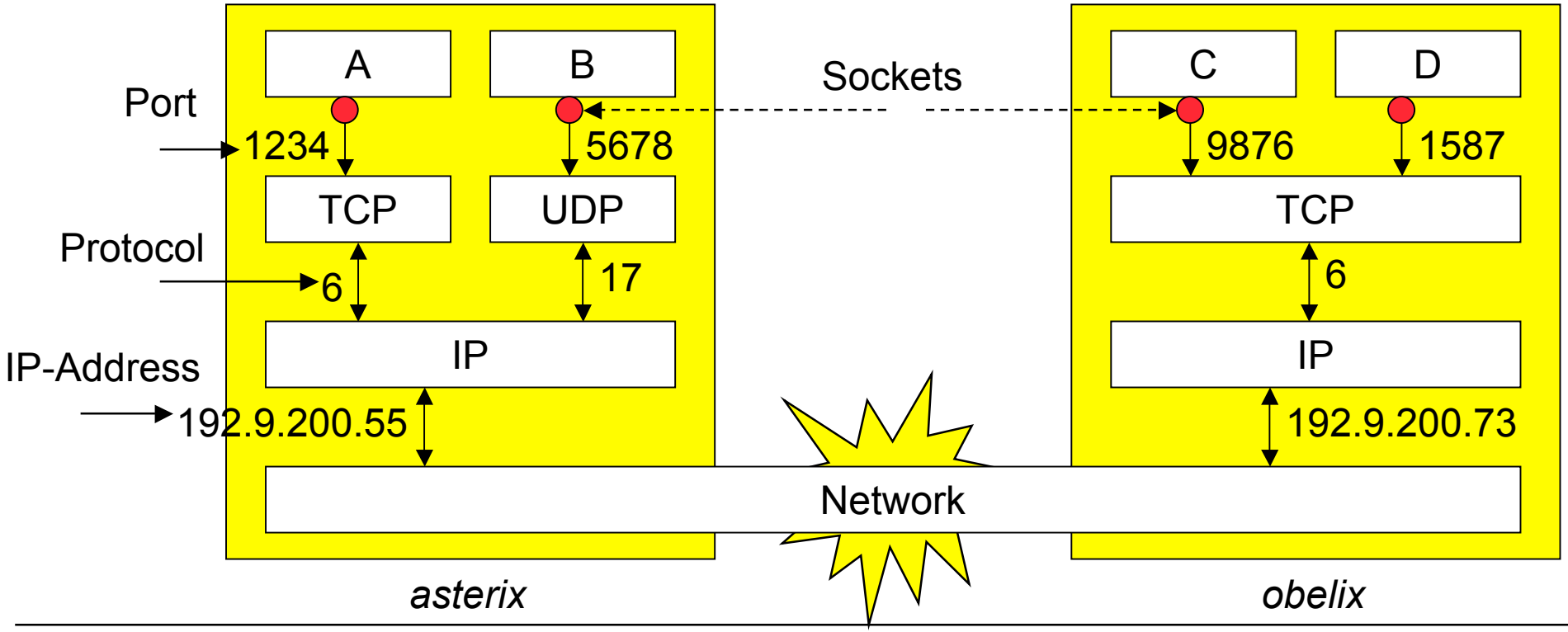
Sockets (2)

- Sockets can be realized using different protocols.
  - The common socket types use TCP or UDP (User Datagram Protocol) as Transport Protocol.
  - Sender and receiver must use the same protocol.



Example: Details of Communicating Processes

- Process A on computer *asterix* communicates with process D on computer *obelix*.
- Process A uses a local socket with port 1234 and process D uses a local socket on port 1587. The protocol is TCP (identified by the protocol number 6).
- Process A sends a message via its socket (the so-called **client socket**) to the so–called **server socket** of process D.



Future of the Internet Protocol (v6)

- Expansion of address space to 128 Bit (2^{128} addresses)
  - 15% utilization Hundreds of IP addresses per human
- Unification of addressing
- Improved Support for Quality-of-Service (QoS) Parameters:
  - Real-time data transmission support via prioritization (0-15)
  - Security mechanisms at the protocol level:
    Encapsulating Security Payload (ESP)
  - Selected Routing to commit packets to fixed paths uniform latency time
- Packet size > 64KB performance increase

Web Technologies (1)

HTTP: Hypertext Transfer Protocol. Purpose: Accessing resources on the internet (web documents). Clients (browsers) issue requests for resources to a server, the server sends the requested document back to the client as a response. Current version is HTTP/1.1.

HTTP Requirements:

1. **Infrastructure**

Proxies, Gateways, Tunnels, Mirrors, Firewalls: Important additional client- and server-side resources on the web used to enhance performance, availability, accessibility and to protect servers, etc.

2. **Target Identification**

URL: Uniform Resource Locator. Defines the location of a resource on the internet. Example: http://www.sts.tu-harburg.de/teaching/ is a URL.

3. **Service Messaging**

Requests, Responses, Headers, Extensions, Negotiation, etc. regulate the format of messages which communicate service details

HTTP Protocol Overview

HTTP Design Goals:

- Simplicity
  - Simple request / response protocol
  - Use few resources on client and server
- Speed

Existing HTTP versions:

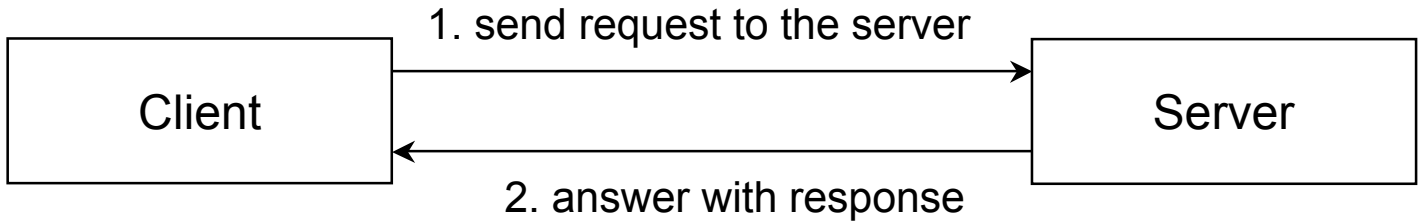
- 0.9
- 1.0
- 1.1 (current)

HTTP Protocol

Definition

HTTP is a simple request/response protocol that is built on top of a reliable, connection-oriented transport service. It makes use of computers in two roles: client and server. Client sends requests to the server, the server then sends responses to the client.

Simplest form of interaction:



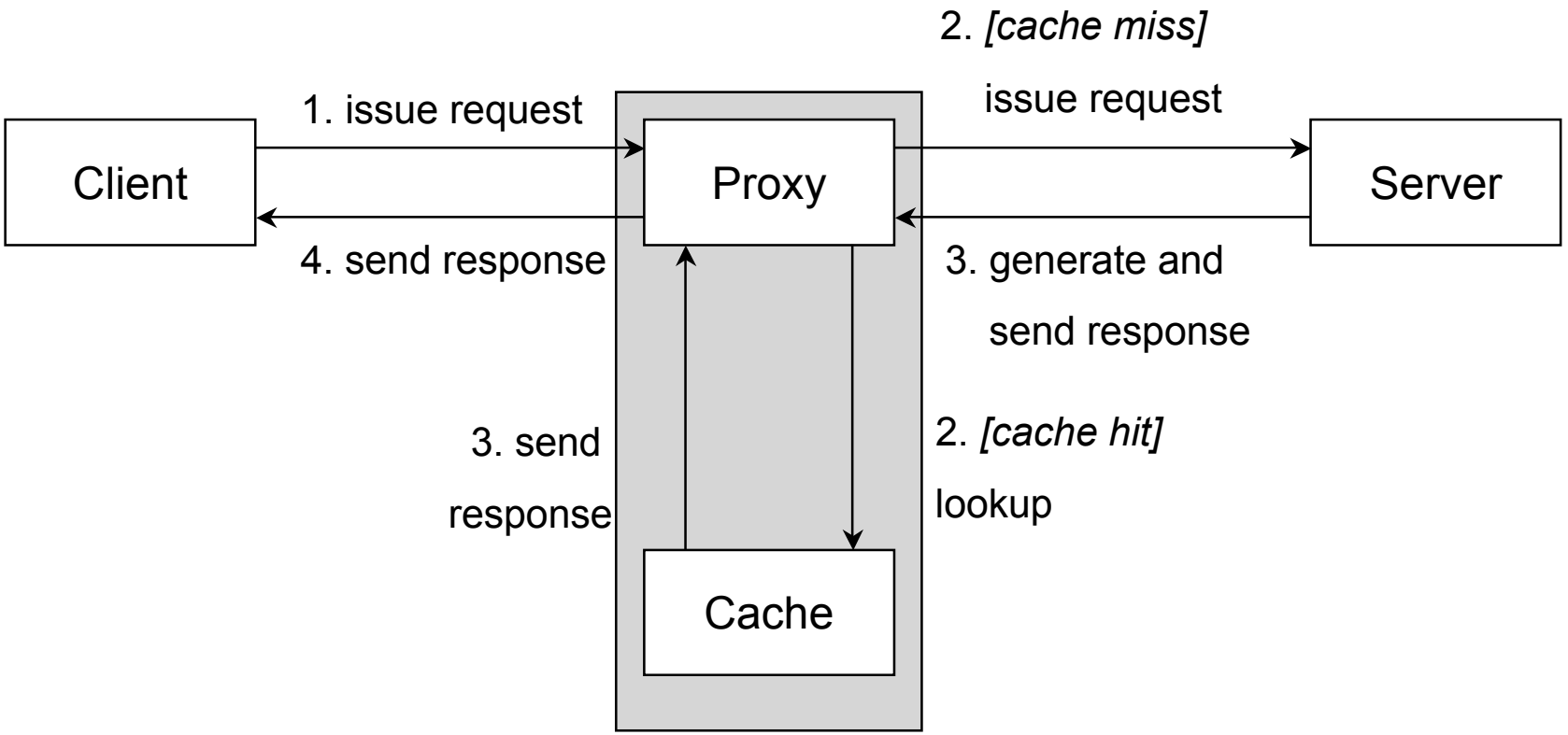
Connections may use intermediate stations. We take a deeper look at the following types of intermediate station:

- Proxies, Gateways, Tunnels
- Mirrors, Firewalls

HTTP: Proxies (1)

Definition

Proxies act as an intermediate station between client and server. They primarily cache requests in order to increase performance.



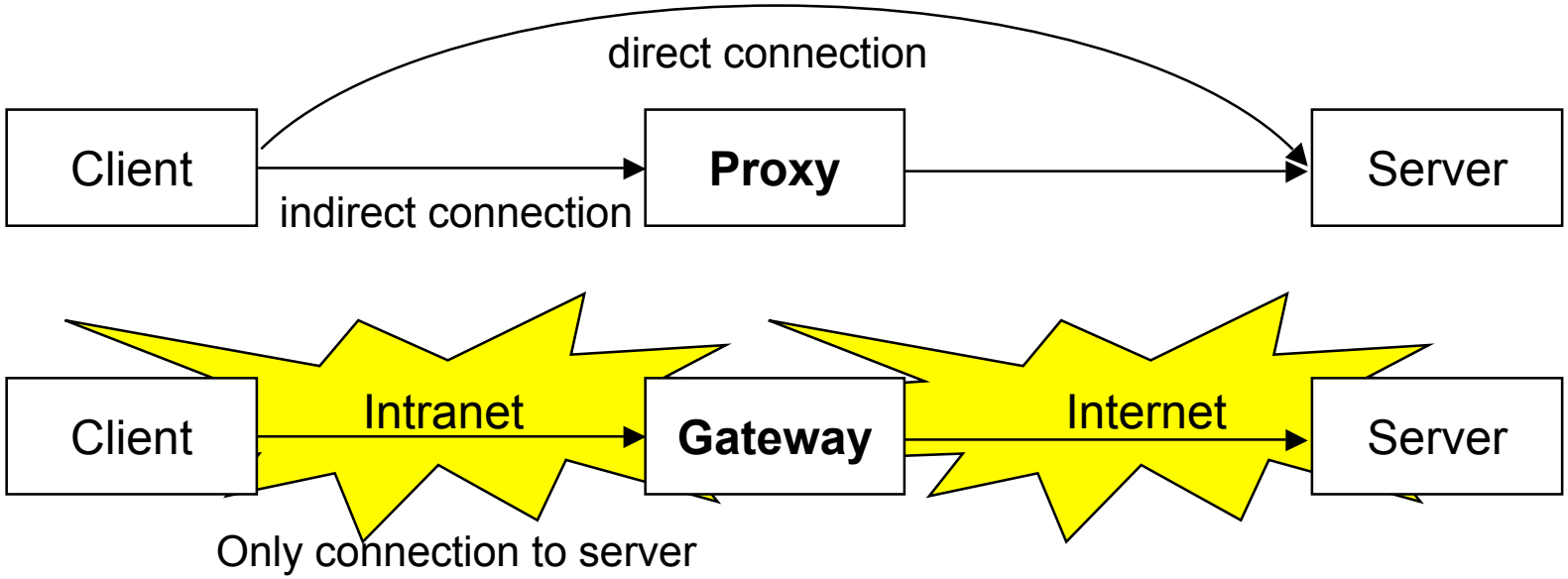
Proxies can act both as a client proxy and server proxy.

HTTP: Proxies and Gateways

Definition

Gateways act as an anonymous intermediate station on the way to the servers. They redirect requests from clients to an origin server. Clients connecting to a (server) proxy know of the proxy. Clients connecting to gateways do not know they do not connect to the origin server. In fact, they (usually) cannot connect directly to the server at all.

Differences between Proxies and Gateways: Proxy usage is not compulsory: Clients need not connect to a server proxy, they can also connect directly to the server. Gateways, on the other hand, cannot be passed by.

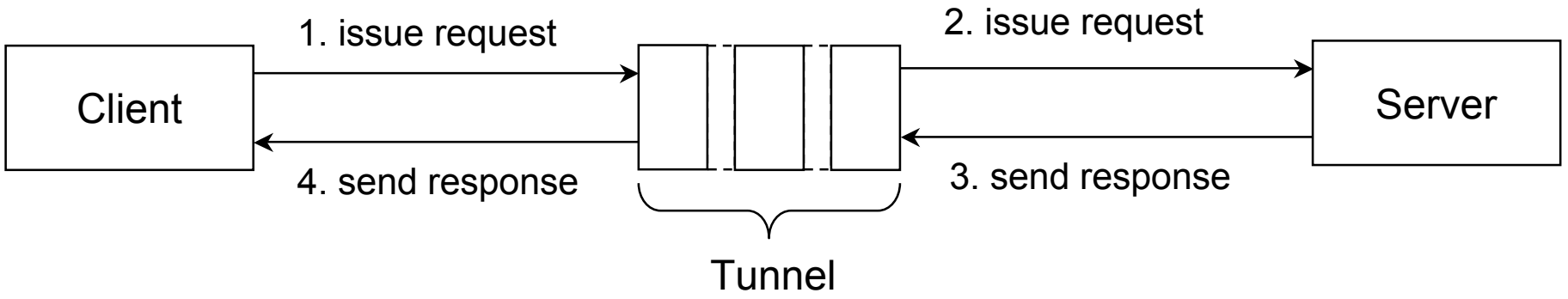


HTTP: Tunnels

Definition

Tunnels are “blind” intermediate stations, i.e., they do not cache requests or responses but only forward them. They may consist of several intermediate stations. They can be used to send requests/responses of one protocol over another protocol at the same level. In contrast to protocol layering, protocol tunneling is transparent (Examples: RAS tunneling, NetBEUI over TCP/IP).

Reasons for tunneling: Speed, availability, security, etc.



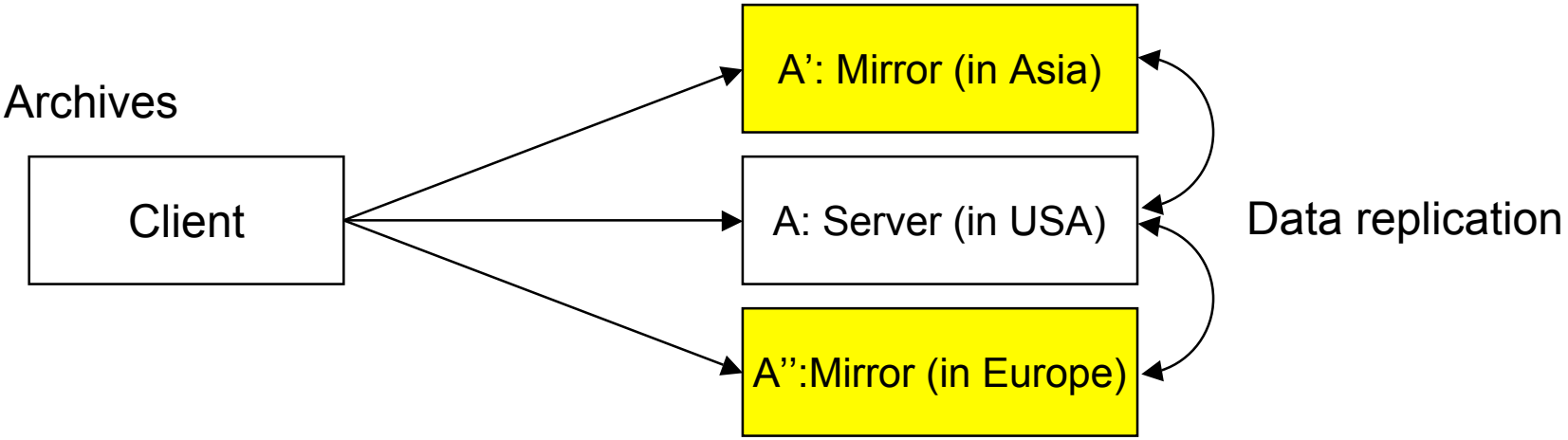
HTTP: Mirrors

A **Mirror** of a server *A* is a server A' that manages the same set of resources like the original server *A*. A mirror either provides read-only content or is responsible for synchronizing updates with the original server *A*.

- Client can choose whether to connect to the original server or to a mirror.
- Mirrors are typically located at remote locations (e.g. different continents).

Example

- FTP Archives

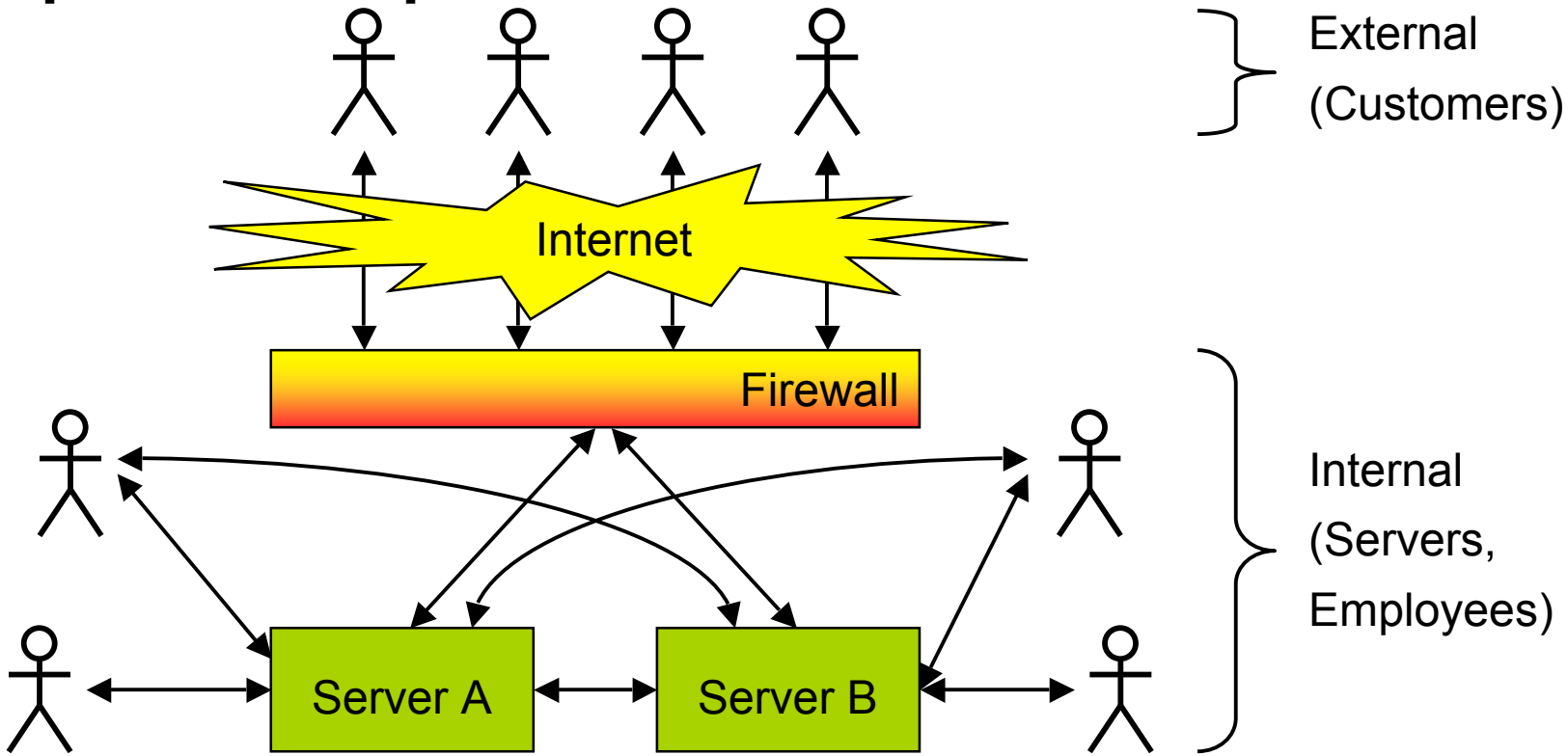


The concept is opposed to ***Replicated Servers*** (example: www222.yahoo.com), where one server balances load (i.e., distributes requests) to all replicated servers.

HTTP: Firewalls (1)

Definition

A **firewall** is a security system protecting a LAN or other networks. It performs monitoring and perhaps routing of traffic in and out of a network or at a bridge, perhaps limiting access to services. Firewalls are used to insulate sensitive data from the Internet without isolating the entire network [CommerceNet00].



HTTP: Firewalls (2)

Elements of a firewall system [Pohlm00]:

Active:

- Active element: Packet filter: low level control

  Analyzes and controls transmitted packets on network layer and internet layer. Stateful packet filters may control packets up to the application layer (HTTP).

- Active element Application Gateway: high level control

  Analyzes and controls packets on the application layer. It holds a proxy for every supported application service (telnet, ftp, smtp, http). The application gateway and its proxies can implement service-specific security measures and logging.

Administration:

- Security Management:

  Controls and administrates packet filter and application gateway.