



POLITECNICO
DI TORINO



e-Lite



Web Information Systems

Sistemi Informativi Aziendali – A.A. 2012/2013

Sommario

1. Definizione
2. Ambiti applicativi
3. Architetture di riferimento



Definizione

Web Information Systems

Definizione

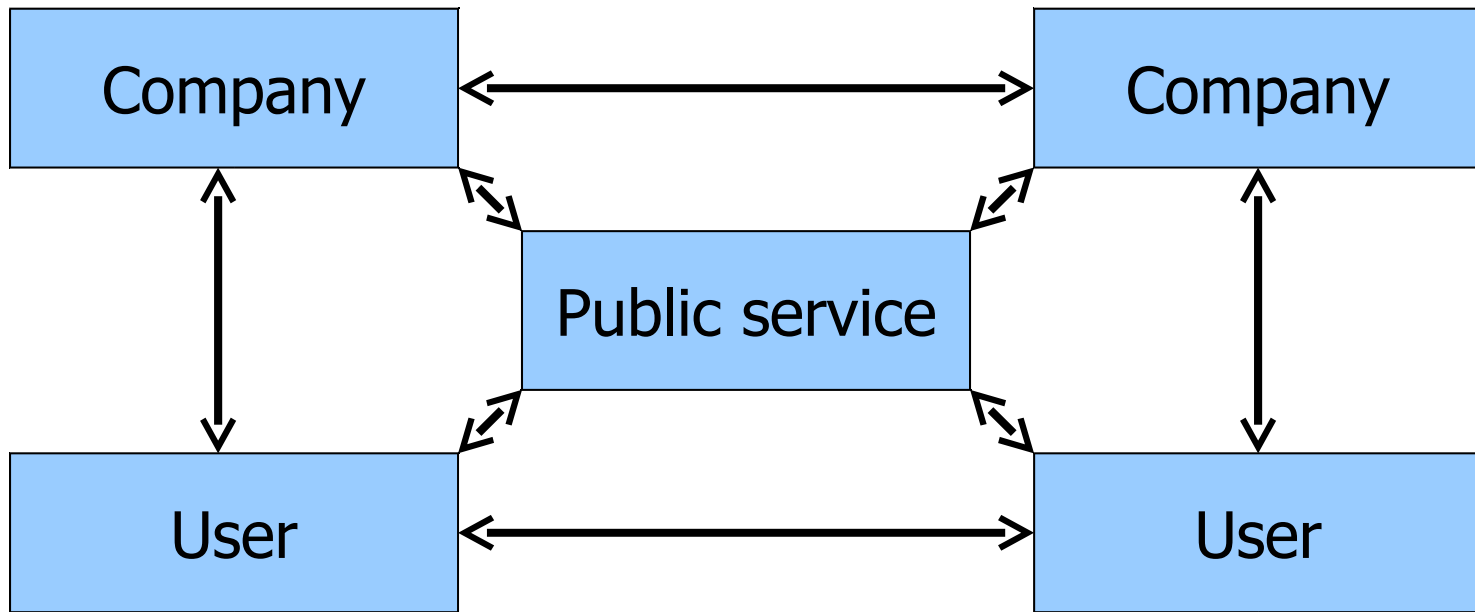
▶ Web Information System (WIS)

- ▶ La comunicazione tra macchine e utenti avviene sulla rete Internet pubblica o su una rete privata basata sugli standard Internet (VPN)
- ▶ L'accesso alle informazioni e ai servizi è supportato da programmi per la gestione dell'interfaccia utente noti come browser.



Cap. 3
Pag. 93

Attori



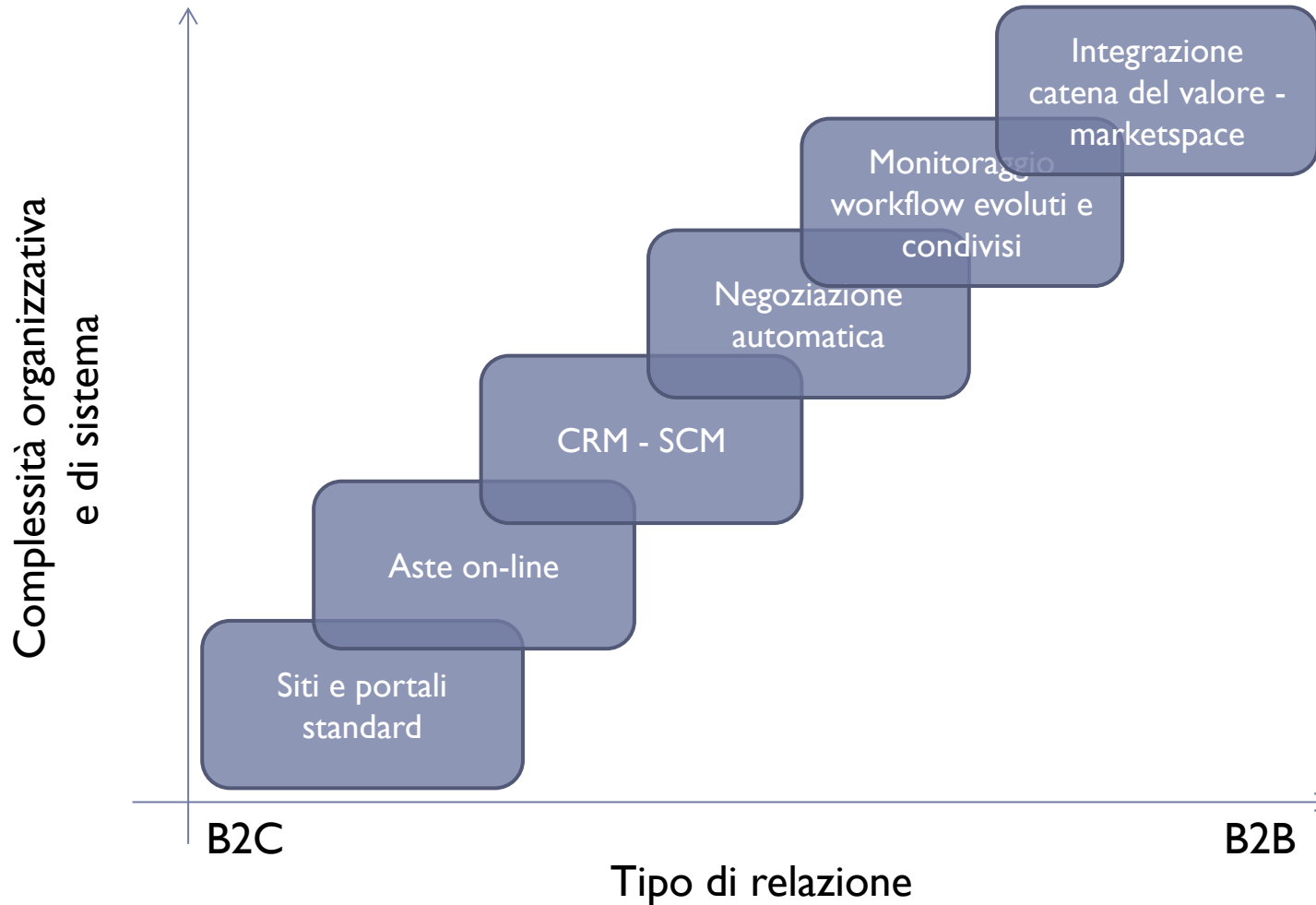
Modelli di collaborazione

- ▶ B2B (business to business): collaboration among companies
- ▶ B2C (business to consumer): on-line shops
- ▶ C2C (consumer to consumer): auctions, buy-sell notices
- ▶ Government to business : on-line taxes, services to companies
- ▶ Government to citizens : on-line taxes

Esempi

- ▶ On-line shops of consumer goods
- ▶ On-line auctions
- ▶ Thematic portal (links, user community, latest news)
- ▶ Distribution of components or raw materials
- ▶ Services (bank, finance, insurance, travel, consultancy, ...)
- ▶ Publications (newspapers, encyclopedias, press agencies, ...)

Una possibile classificazione



Livelli di complessità

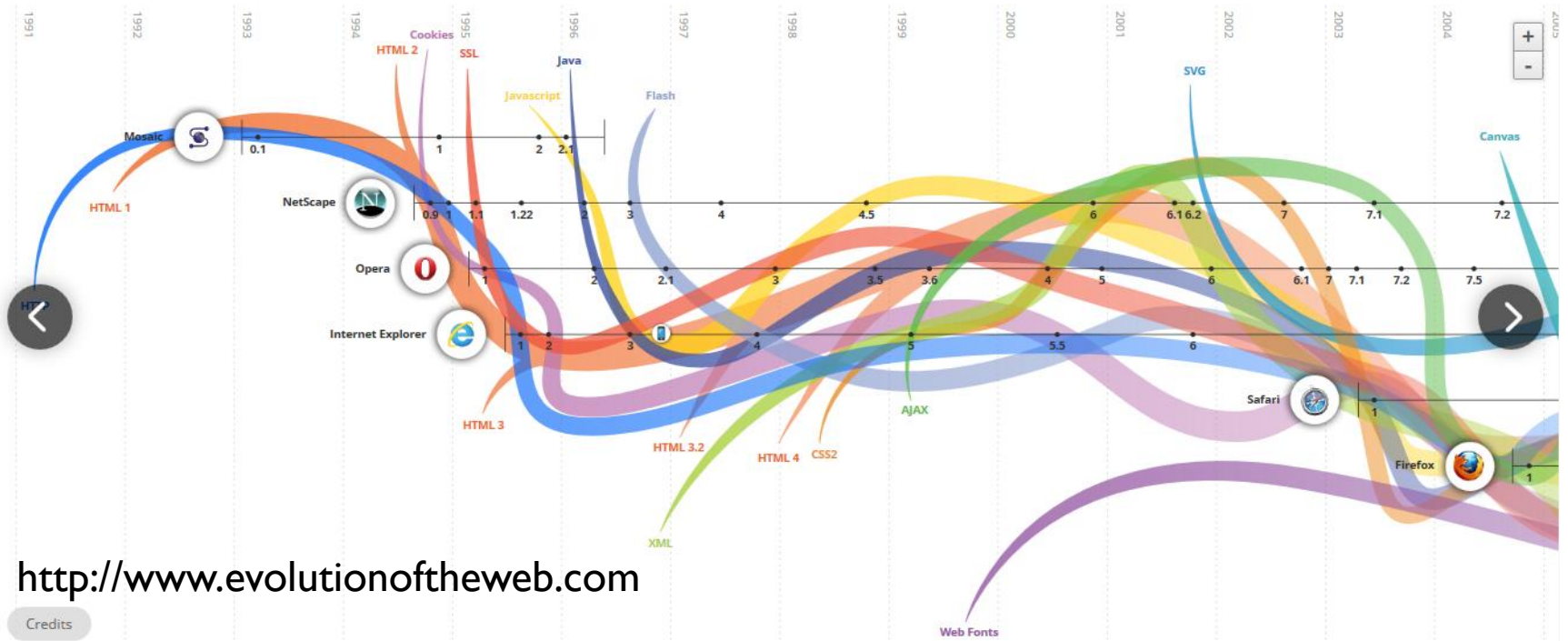
- ▶ **Siti informativi**
 - ▶ Chi siamo / Prodotti / Servizi / Contatti
 - ▶ Newsletter, Giornale, Blog, ...
- ▶ **Siti dispositivi**
 - ▶ Scelta prodotto, configurazione, acquisto
- ▶ **Sistemi gestionali**
 - ▶ CRM, SCM, ERP, MRP, ...
- ▶ **Sistemi autonomi**
 - ▶ Negoziazione, transazione, monitoraggio
- ▶ **Portali, marketplace, marketspace**
 - ▶ Aggregazione di più aziende/prodotti correlati



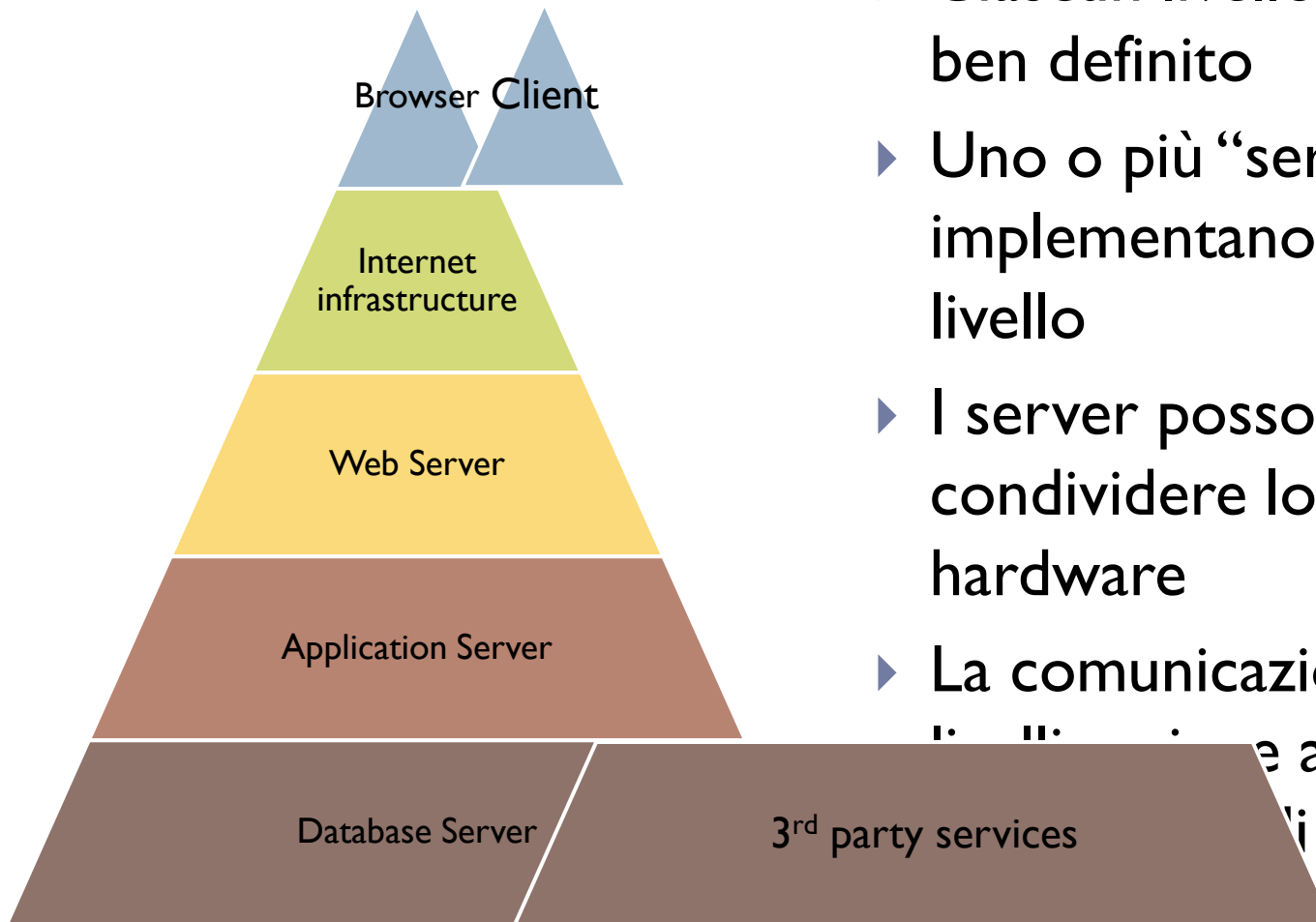
Architetture di riferimento

Web Information Systems

Evoluzione delle architetture web

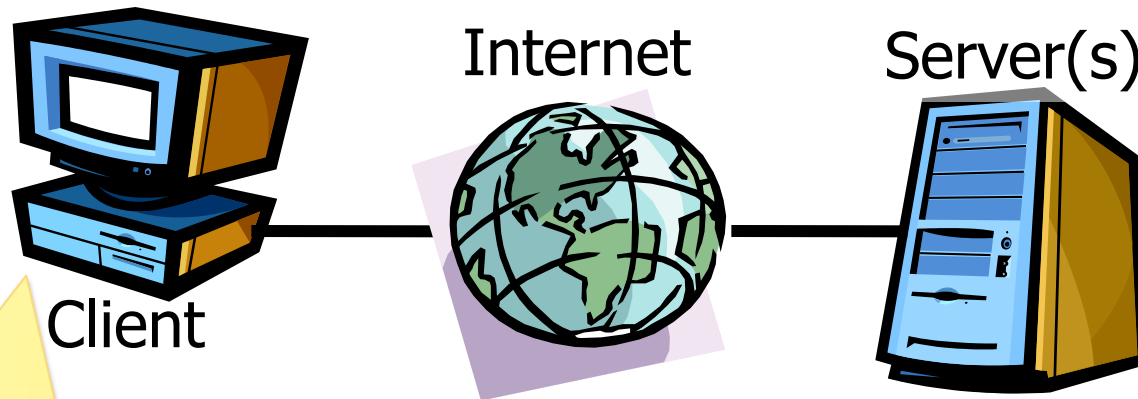


Architettura ad N livelli



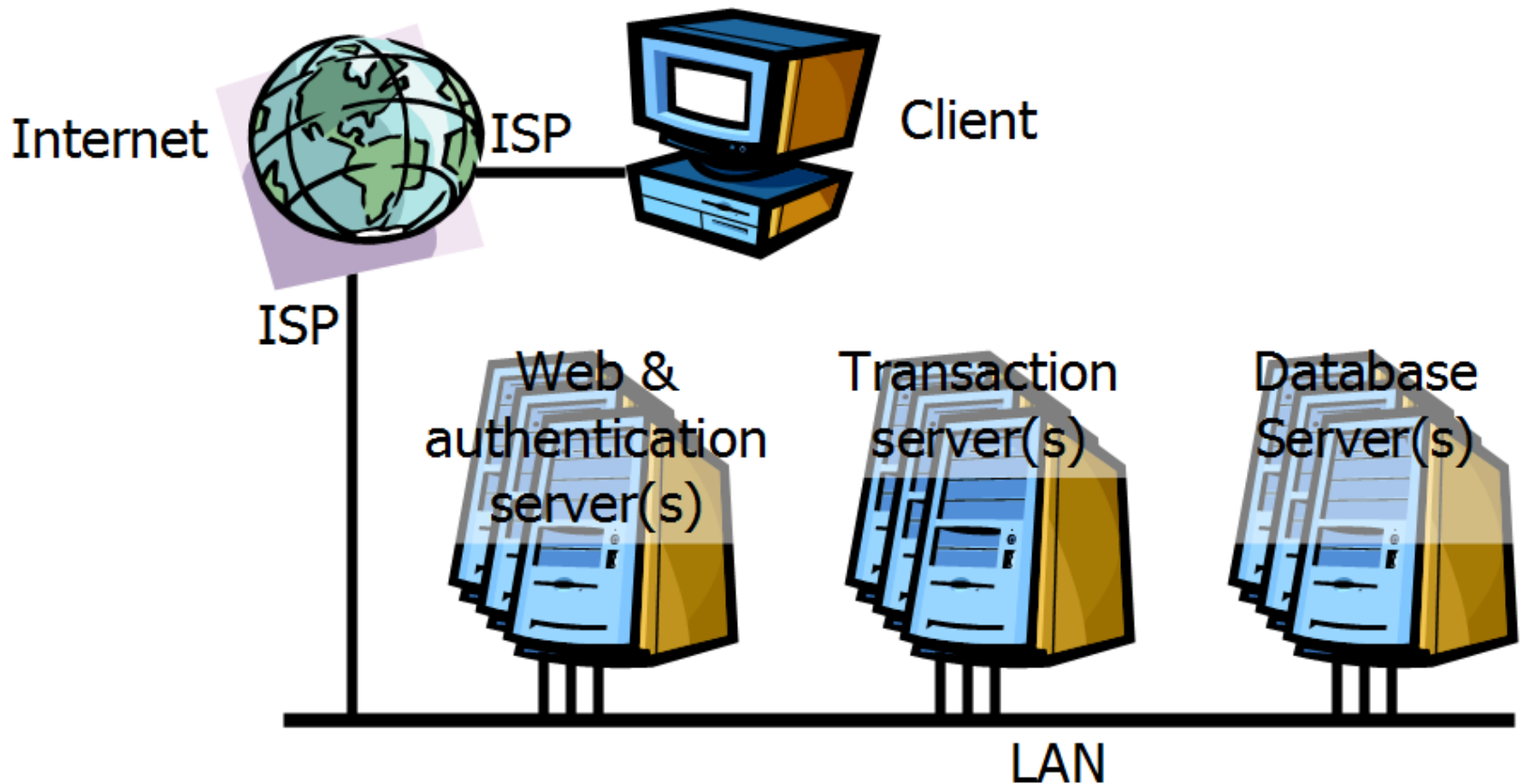
- ▶ Ciascun livello ha un ruolo ben definito
- ▶ Uno o più “server” implementano ciascun livello
- ▶ I server possono condividere lo stesso hardware
- ▶ La comunicazione tra i livelli avviene attraverso la rete

Architettura generale



- Storicamente, un browser web
- Ma anche:
 - Applicazione Mobile
 - Applicazione Desktop
 - Altra applicazione server

General Architecture



Components

- ▶ One or more connections to the Internet by means of an Internet Service Provider (ISP).
- ▶ One or more servers implementing each tier/level of the architecture.
- ▶ One or more physical networks for interconnecting the servers.
- ▶ One or more network devices (router, firewall, switch) which implement communication and security policies.

Definition

- ▶ “Server” may be defined as:

- ▶ Logical definition:

- A **process** that runs on a host that relays information to a **client** upon the client sending it a **request**.

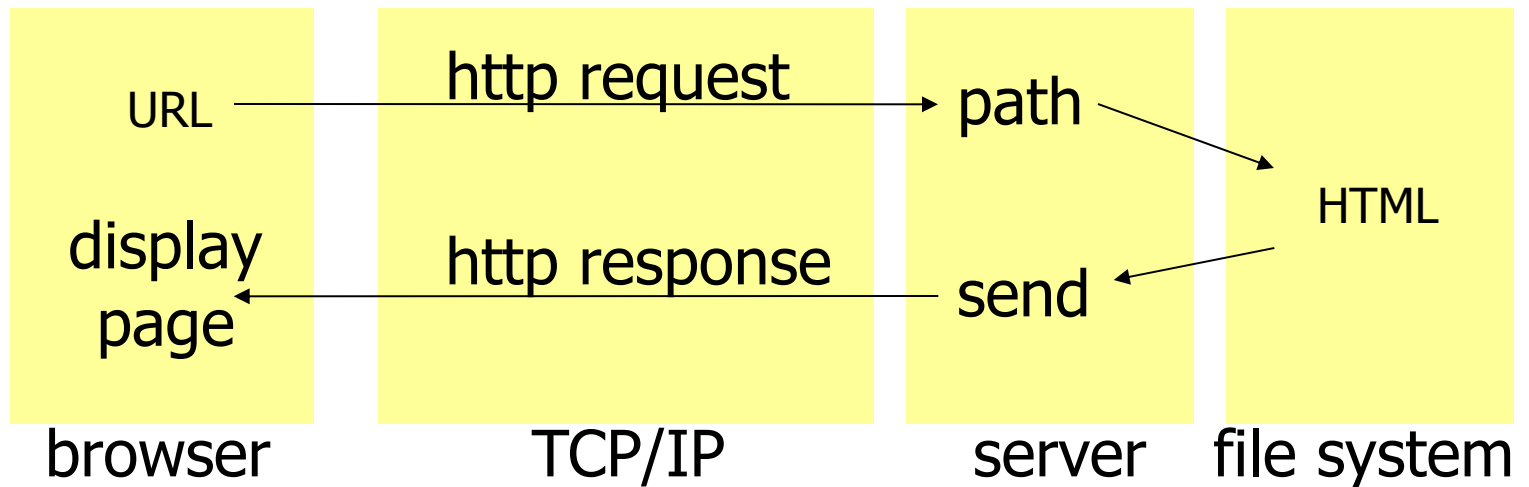
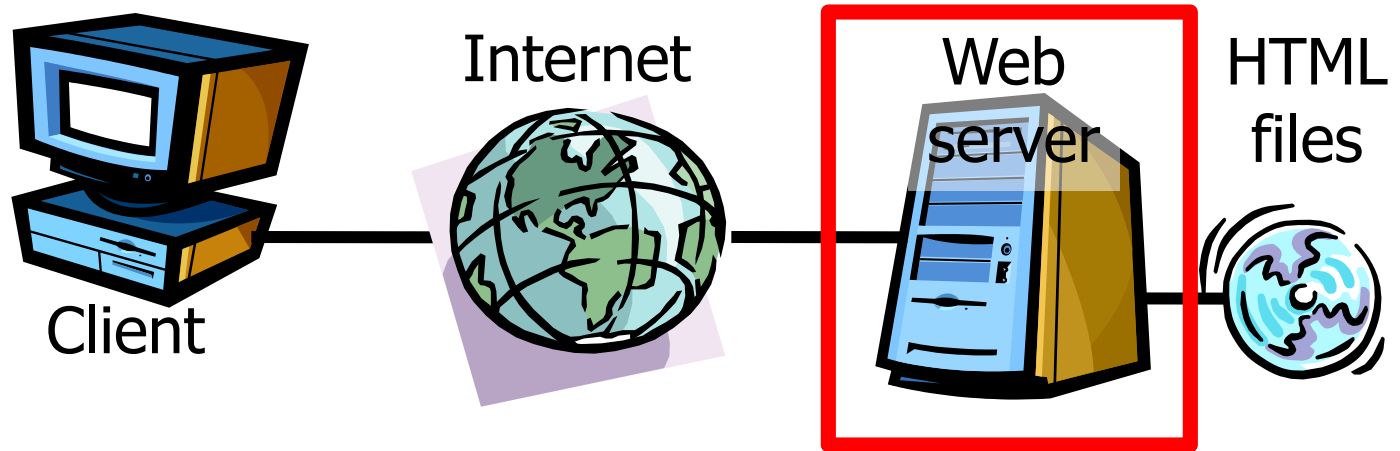
- ▶ Physical definition:

- A **host computer** on a network that holds information (eg, Web sites) and responds to requests for information

Web server

- ▶ Manages the HTTP protocol (handles requests and provides responses)
 - ▶ Receives client requests
 - ▶ Reads static pages from the filesystem
 - ▶ Activates the application server for dynamic pages (server-side)
 - ▶ Provides an HTML file back to the client
- ▶ One HTTP connection for each request
- ▶ Multi-process, Multi-threaded or Process pool

Example



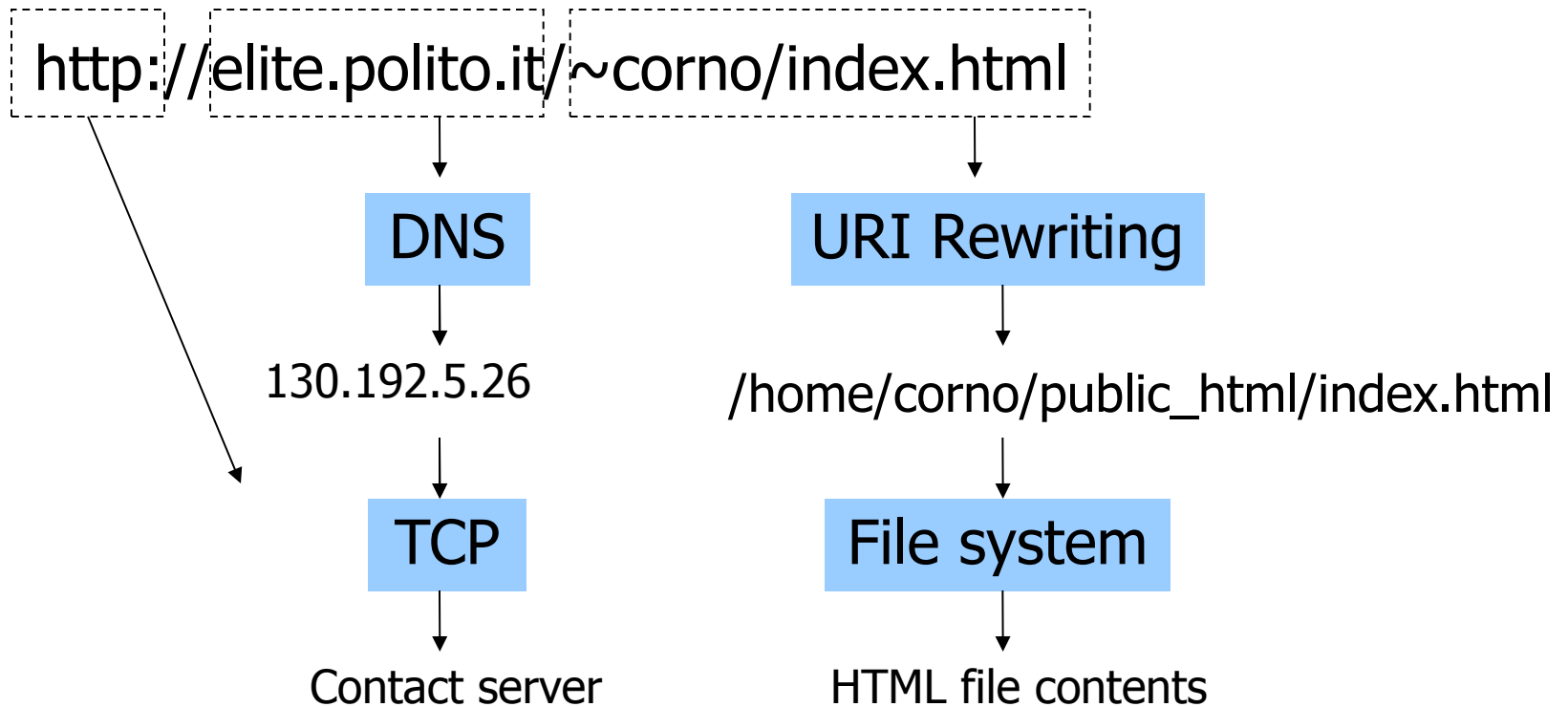
Adopted standards

- ▶ URL (uniform resource locator) for finding web pages
- ▶ HTML (hyper text markup language) for writing web pages
- ▶ GIF (graphics interchange format) for images
- ▶ HTTP (hyper text transfer protocol) for client-server interaction
- ▶ TCP/IP (transmission control protocol over internet protocol) for data transfer

URL

RFC 2396

<http://www.w3.org/Addressing/>



URI Basics

▶ **Scheme** **Hostname** **Query**
`http://www.sadev.co.za/users/1/contact`

▶ **Scheme** **Hostname** **Query**
`http://www.sadev.co.za?user=1&action=contact`

▶ **Scheme** **Userinfo** **Hostname** **Port**
`http://rob:pass@bbd.co.za:8044`

▶ **Scheme** **Hostname** **Query** **Fragment**
`https://bbd.co.za/index.html#about`



HTTP protocol

RFC 2616, RFC 2617
<http://www.w3.org/Protocols>

```
GET /~corno/index.html HTTP/1.0
Accept: text/html
Accept: image/gif
User-Agent: CornoSoft SuperBrowser 9.45
```

```
HTTP/1.0 200 OK
Date: Monday, 01-Jan-2001 00:00:00 GMT
Server: Apache 1.3.0
MIME-Version: 1.0
Last-Modified: 31-Dec-2000
Content-type: text/html
Content-length: 3021

<HTML> . . .
```

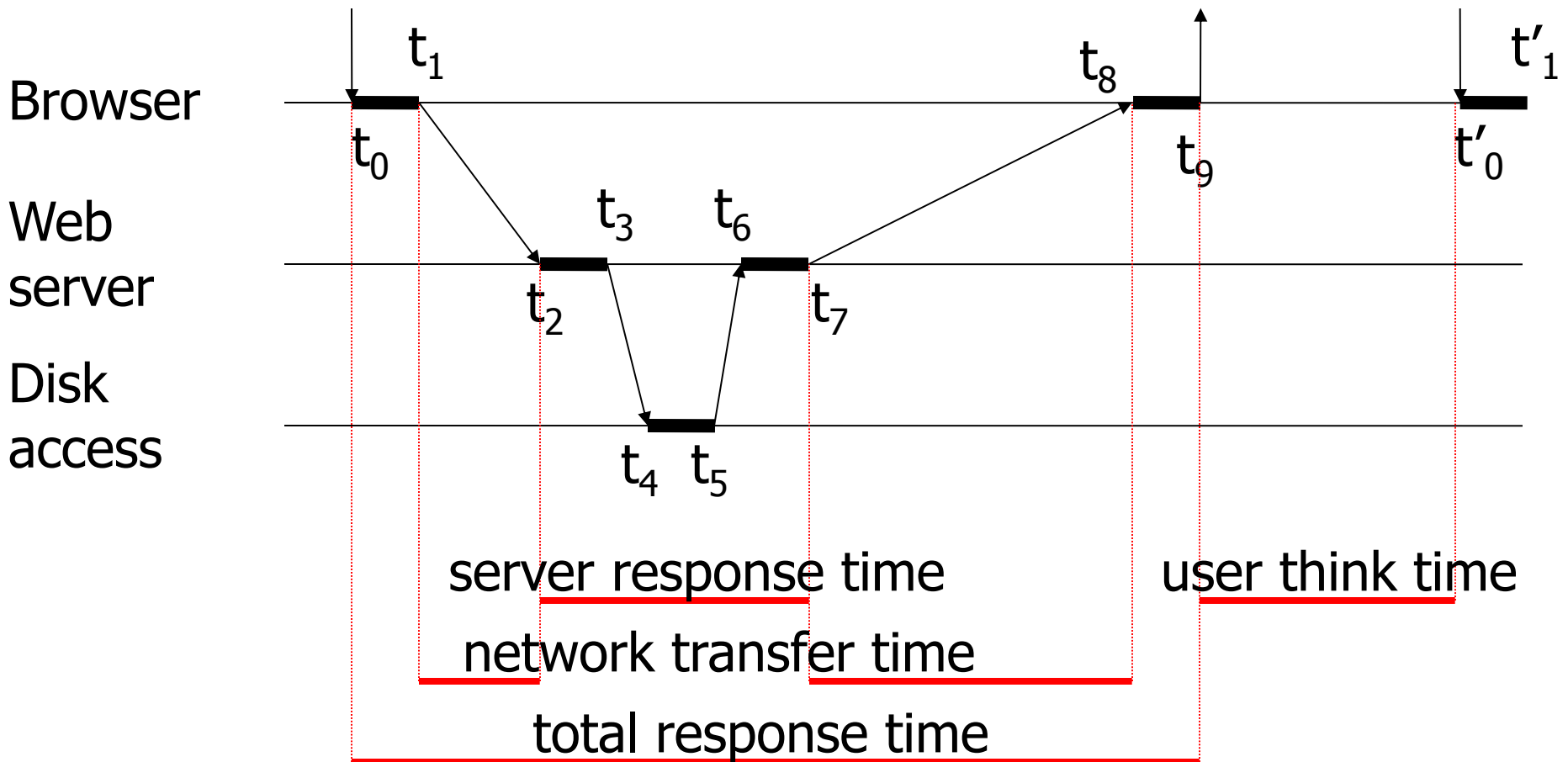
Performance measures

- ▶ Latency: time required for providing a 0 byte http page. Includes the server activation time, the request decoding time, the file access time, the transmission time and the time for closing the connection.
 - ▶ Unit of measure: http/s or s/http
- ▶ Throughput: maximum speed at which infinite-sized pages can be sent.
 - ▶ Unit of measure: Bytes (Mbytes)/s
- ▶ #Requests / s

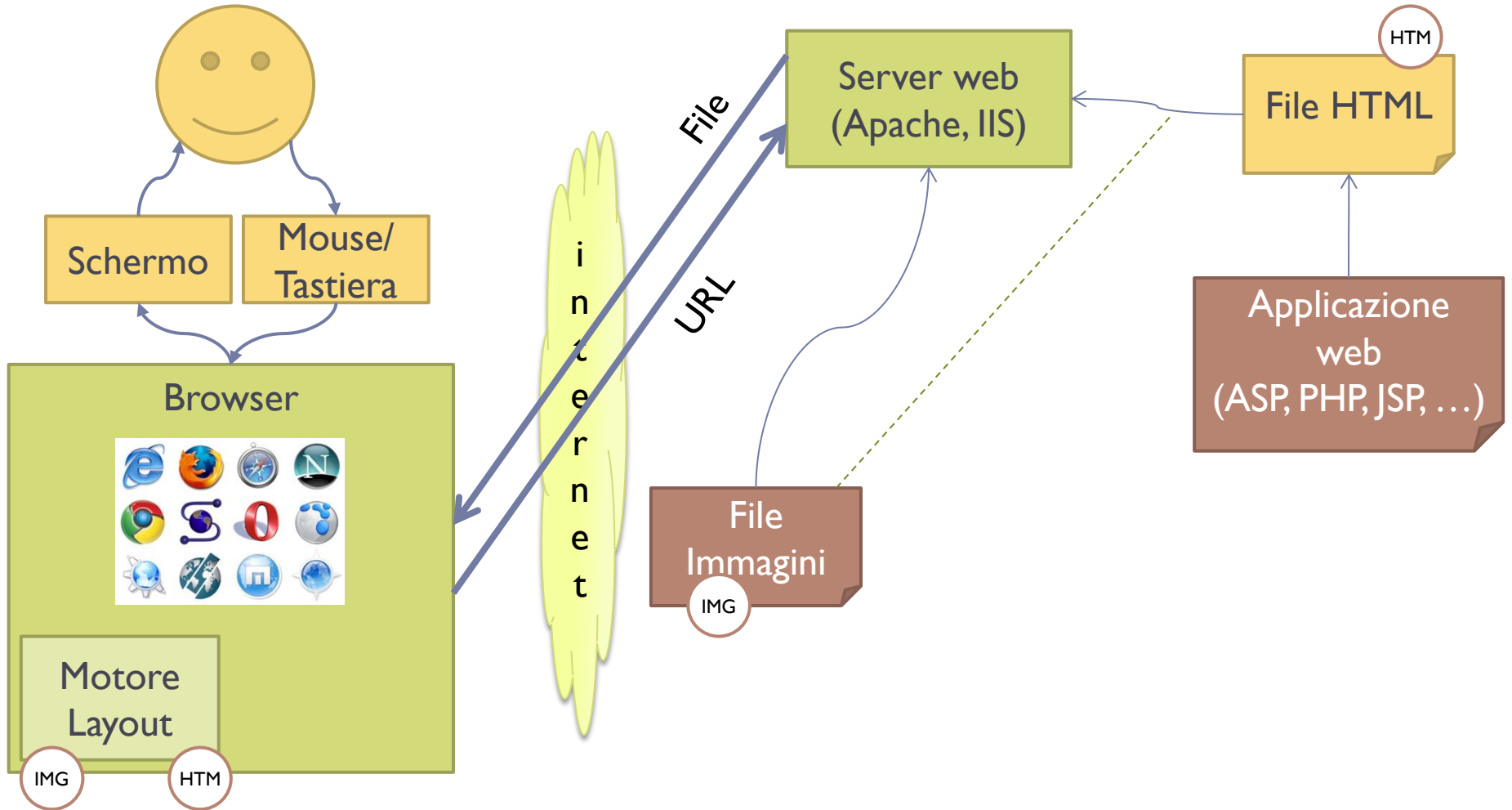
Delay time

- ▶ $T = \text{Latency} + \text{ByteHTML} / \text{Throughput}$
- ▶ This equation is valid if:
 - ▶ The other architecture elements (I/O subsystem, network, ...) are not overloaded
 - ▶ The web server has not yet reached its maximum workload
- ▶ Example:
 - ▶ Latency: 0,1s
 - ▶ ByteHTML: 100kBytes
 - ▶ Throughput: 800kBytes/s
 - ▶ $T = 0,1s + 100K\text{Bytes} / 800K\text{Bytes/s} = 0,225s$

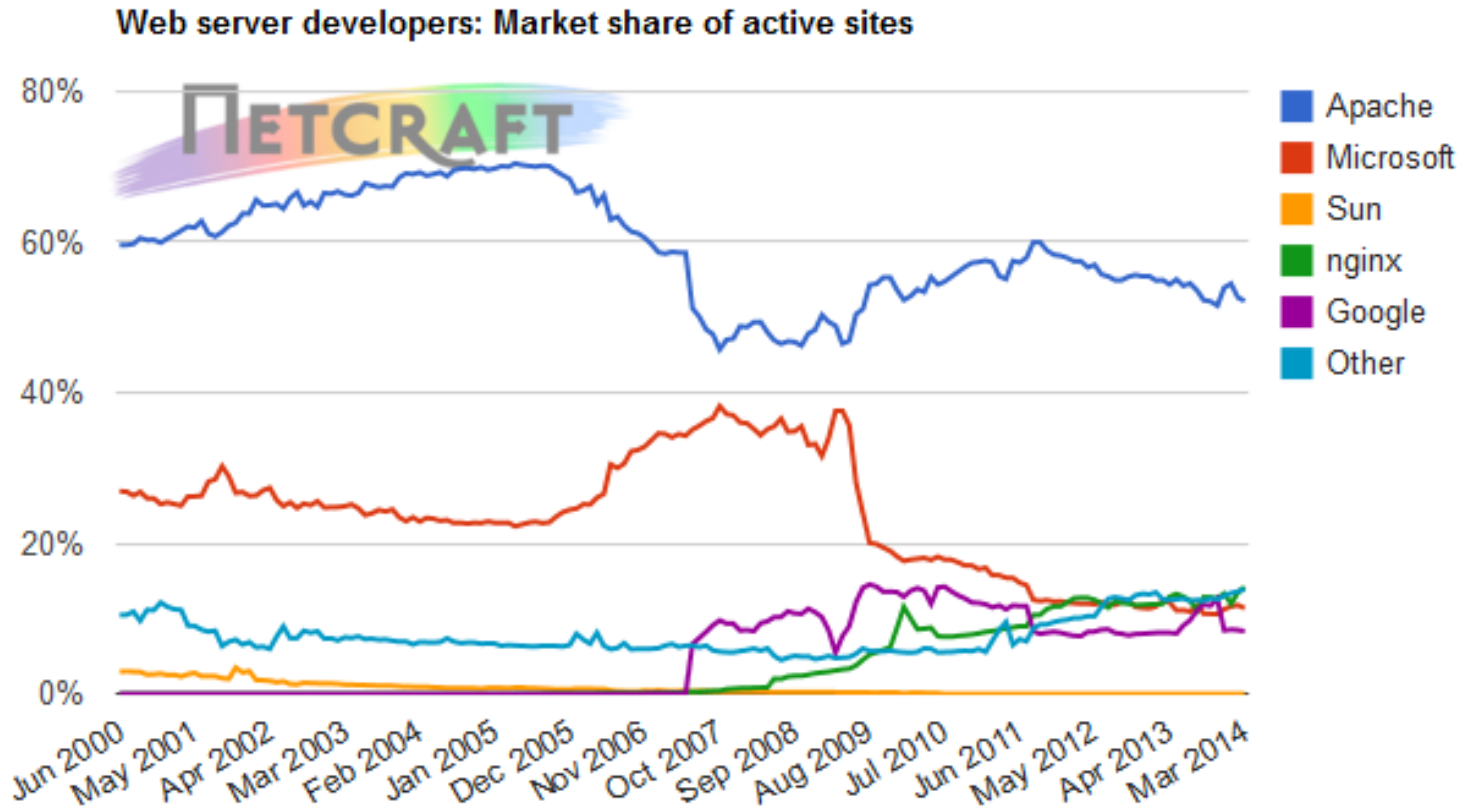
Static web transaction



Architettura generale del web



The most adopted web servers



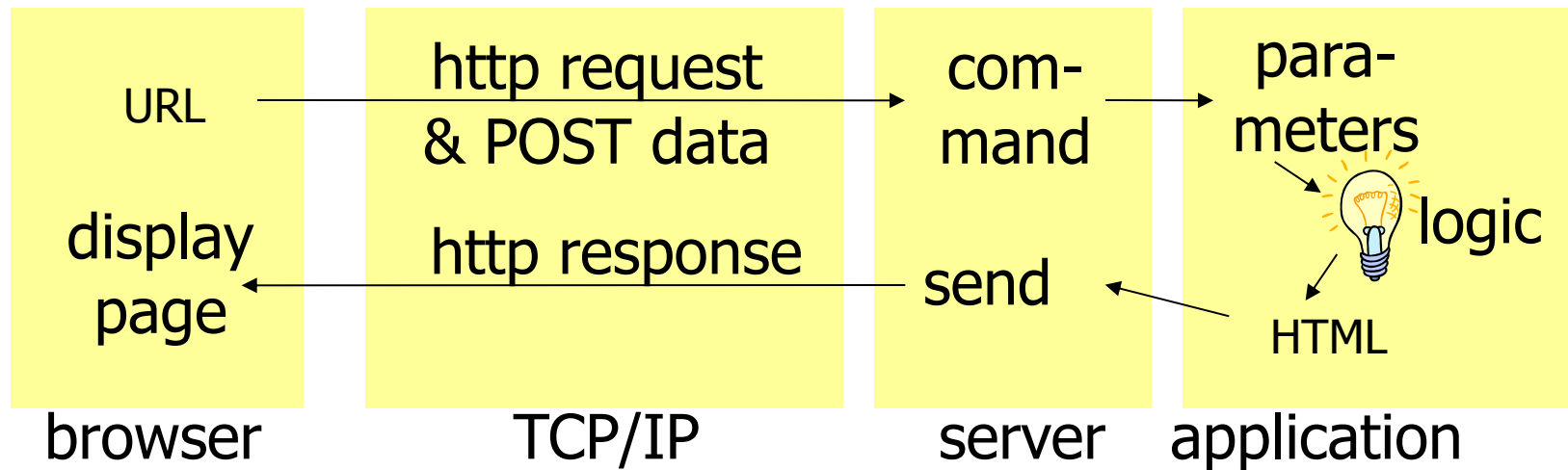
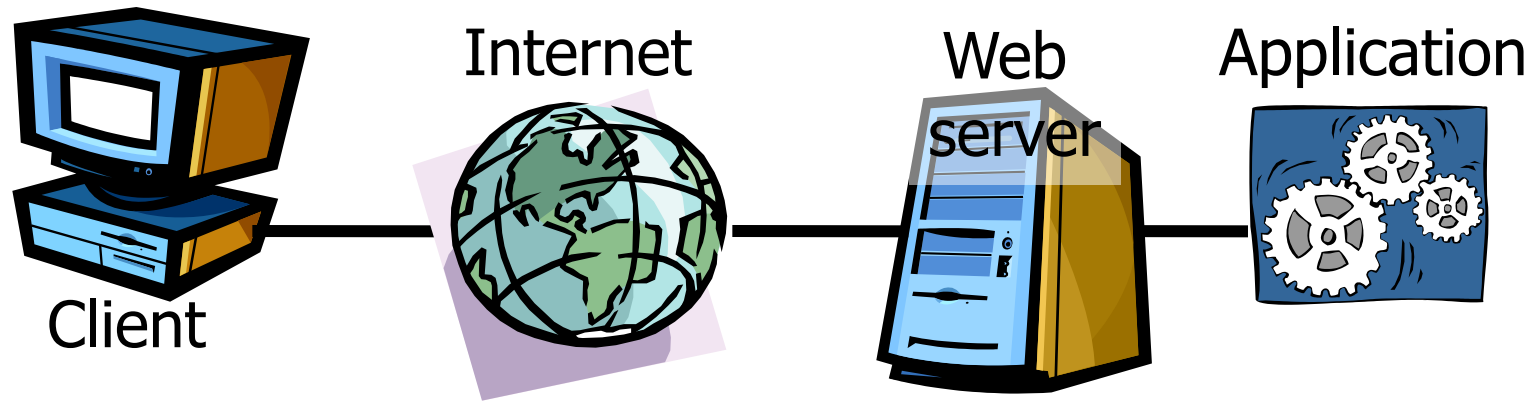
Source: <http://news.netcraft.com/>

<http://news.netcraft.com/archives/2014/03/03/march-2014-web-server-survey.html>

Application server

- ▶ Dynamic page generation
- ▶ Manages the site business logic
- ▶ It's the middle tier between the client browser and the data residing on a database
- ▶ Implements the session mechanisms
- ▶ Different technologies and architectures are available

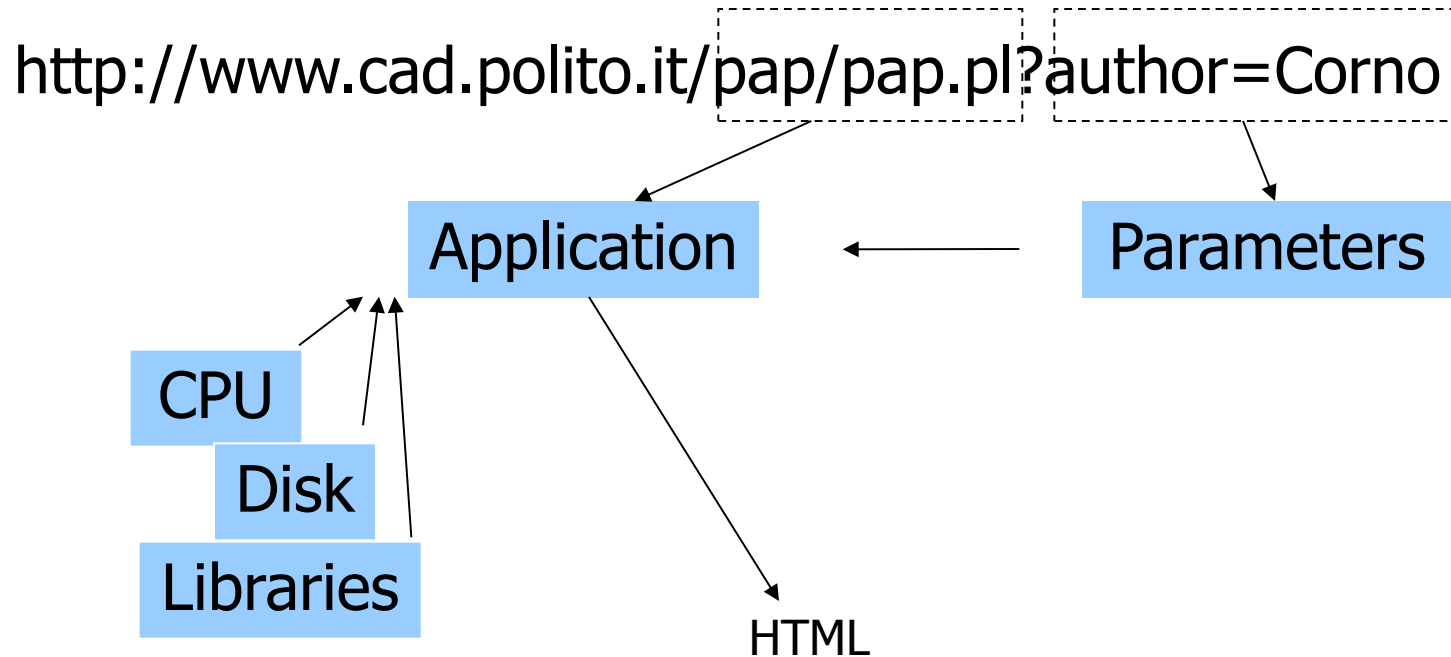
Dynamic web transaction



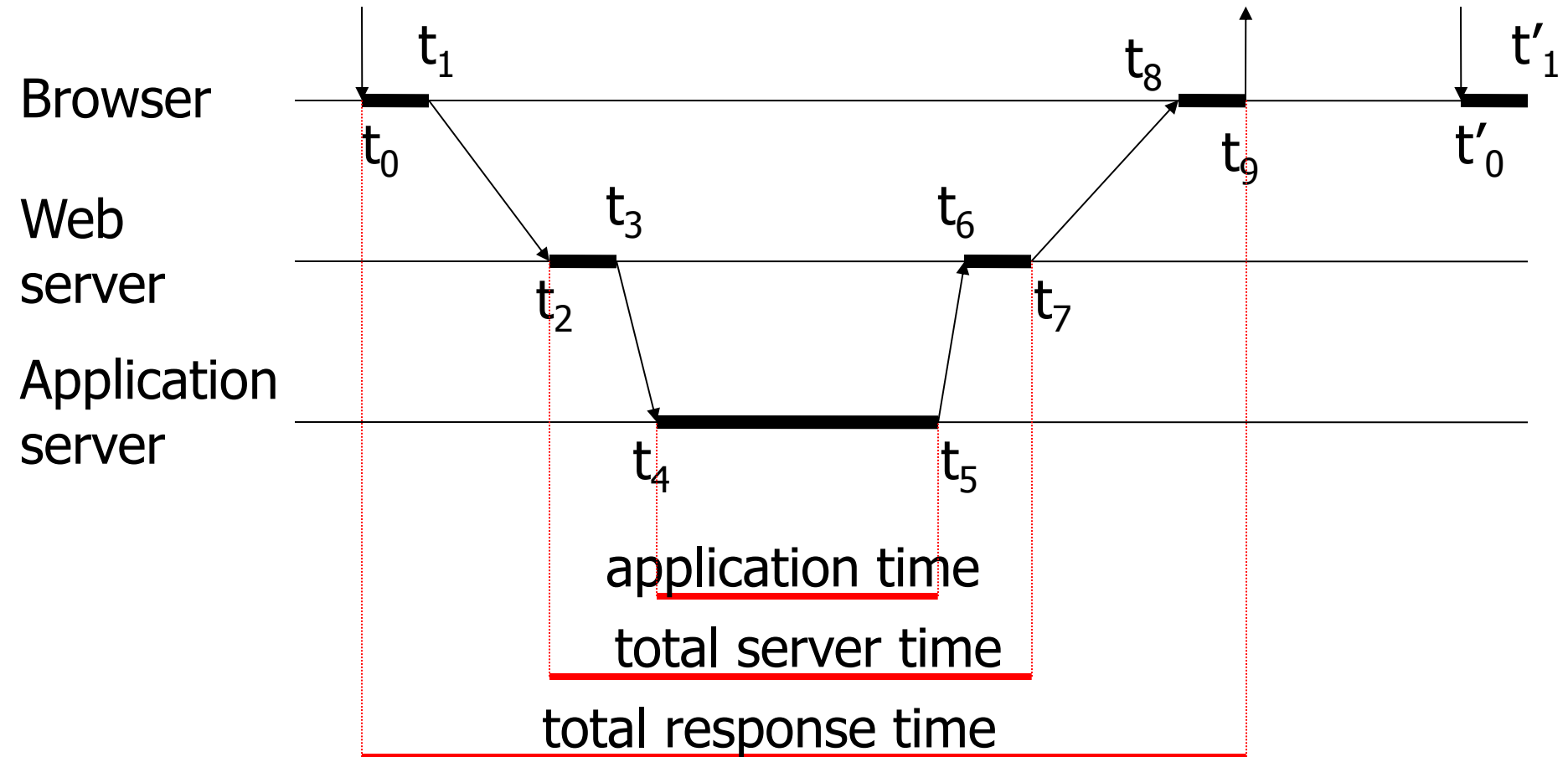
Adopted standards

- ▶ HTTP-POST for sending user-specified data
- ▶ CGI (common gateway interface), ISAPI (internet information server application programming interface), server-side script, java servlet for integrating application logic into web servers
- ▶ ASP (active server pages), PHP, PERL as new languages for application development

URL (HTTP GET)



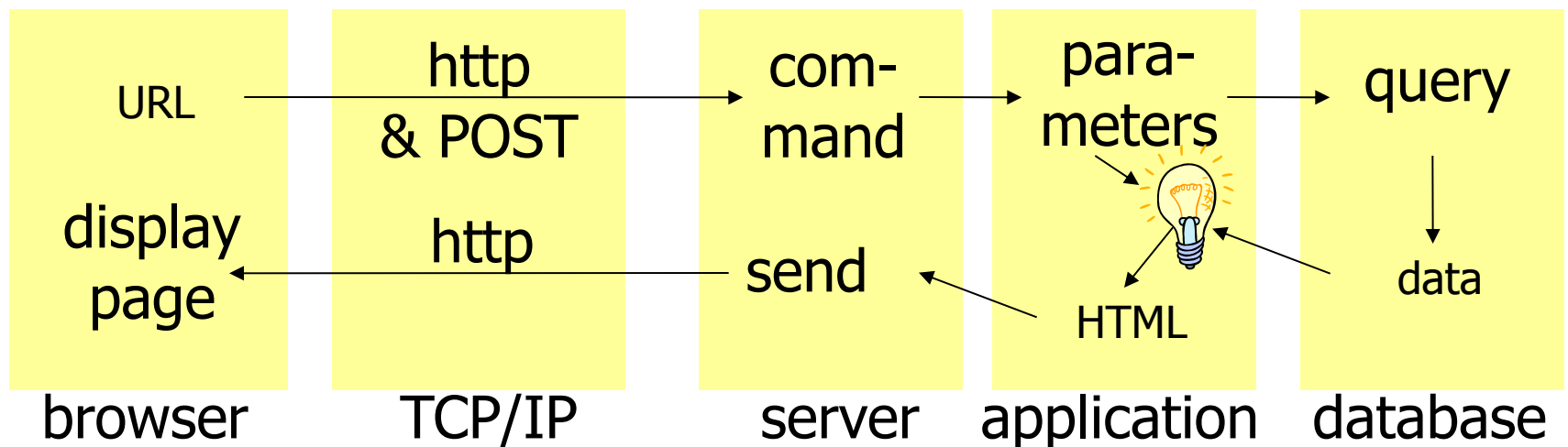
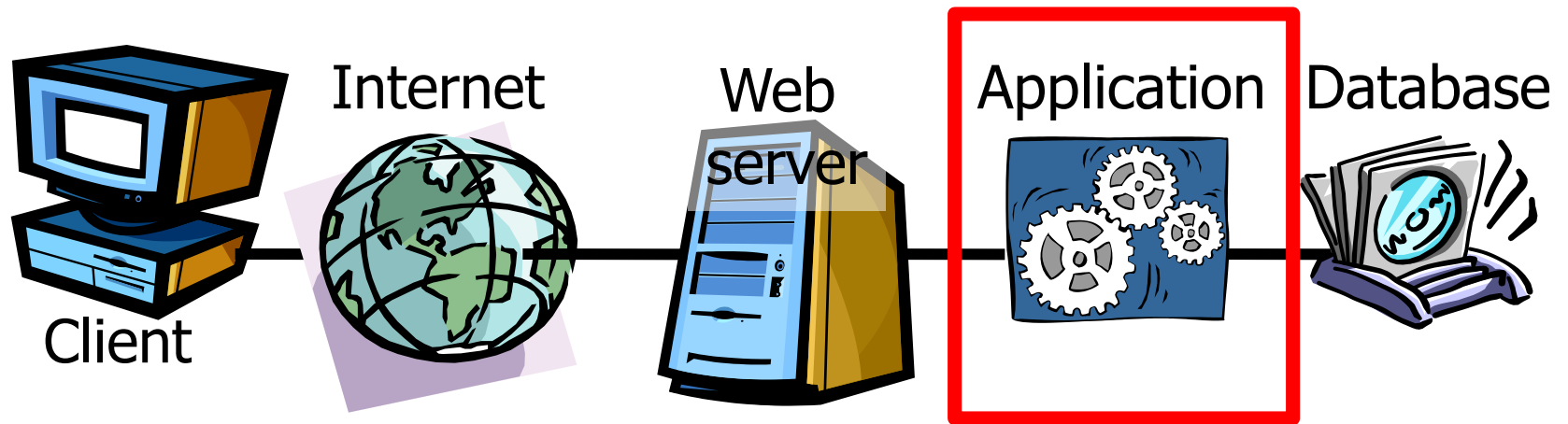
Dynamic web transaction



Database server

- ▶ Stores the data on which the application server works.
- ▶ Executes the queries issued by the application server:
 - ▶ Updates the stored data
 - ▶ Inserts new data
 - ▶ Provides back query results
- ▶ The most frequent/complex queries can be implemented internally as stored procedures (pre-compiled queries with parameters)

Example



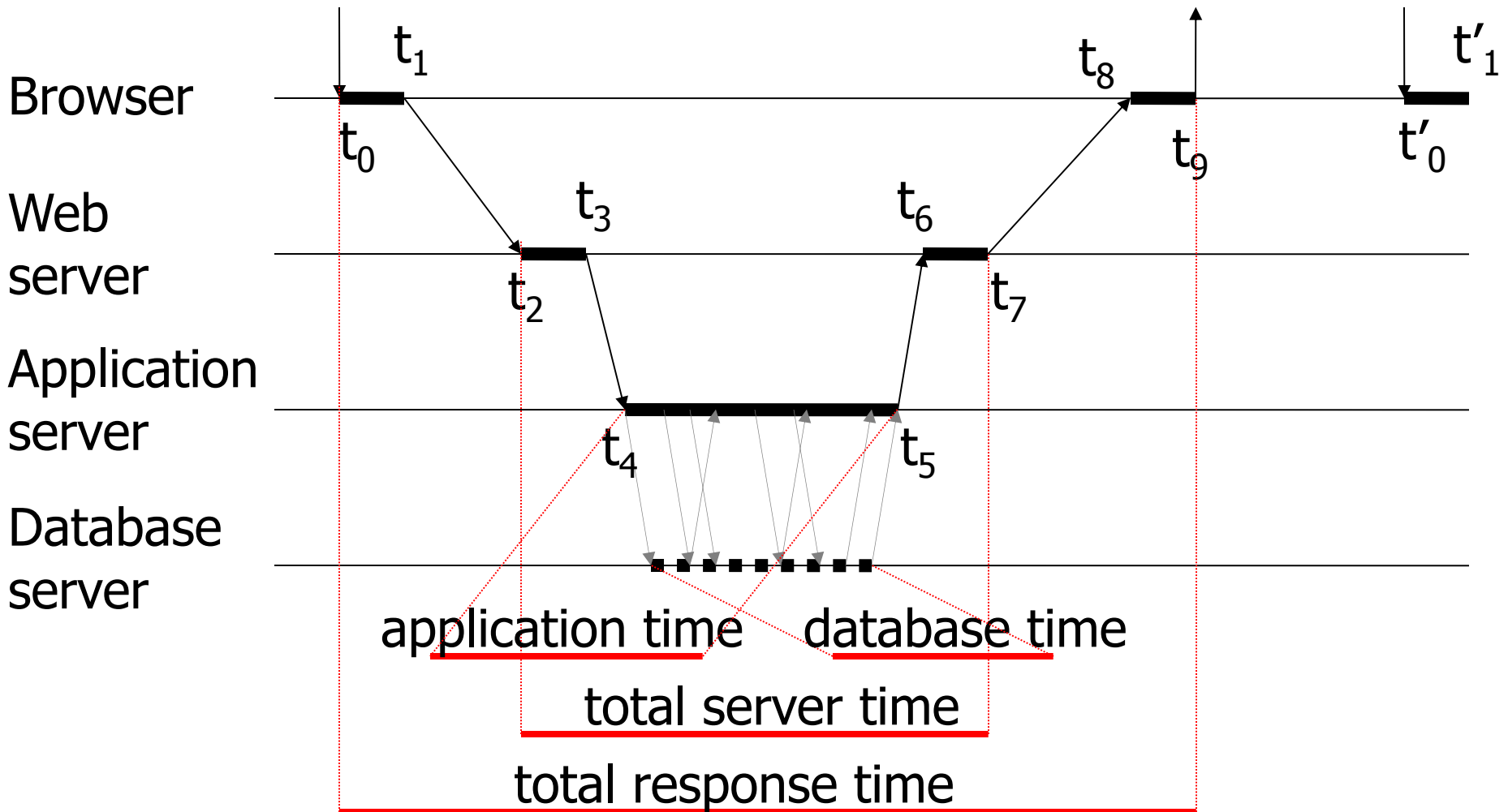
Adopted standards

- ▶ Cookies for storing the state of a session
- ▶ Java, JavaScript, ActiveX, Flash to program the user interface on the browser
- ▶ SQL (structured query language), ODBC (open database connectivity) to access data bases

Database server

- ▶ Queries are almost always in SQL
 - ▶ `SELECT * FROM table;`
 - ▶
- ▶ Often adopts the relational database model
 - ▶ Other models can be used
 - Object model
 - Triple model
- ▶ The most advanced/complete solutions are called Transaction servers

Database-driven transaction



Example (PHP)

The application composes the query

```
▶ <?php
▶ $query = "SELECT doc_id FROM key_doc_index, keywords WHERE
key_doc_index.key_id = keywords.id AND keywords.key =
$_REQUEST["query"];";
```

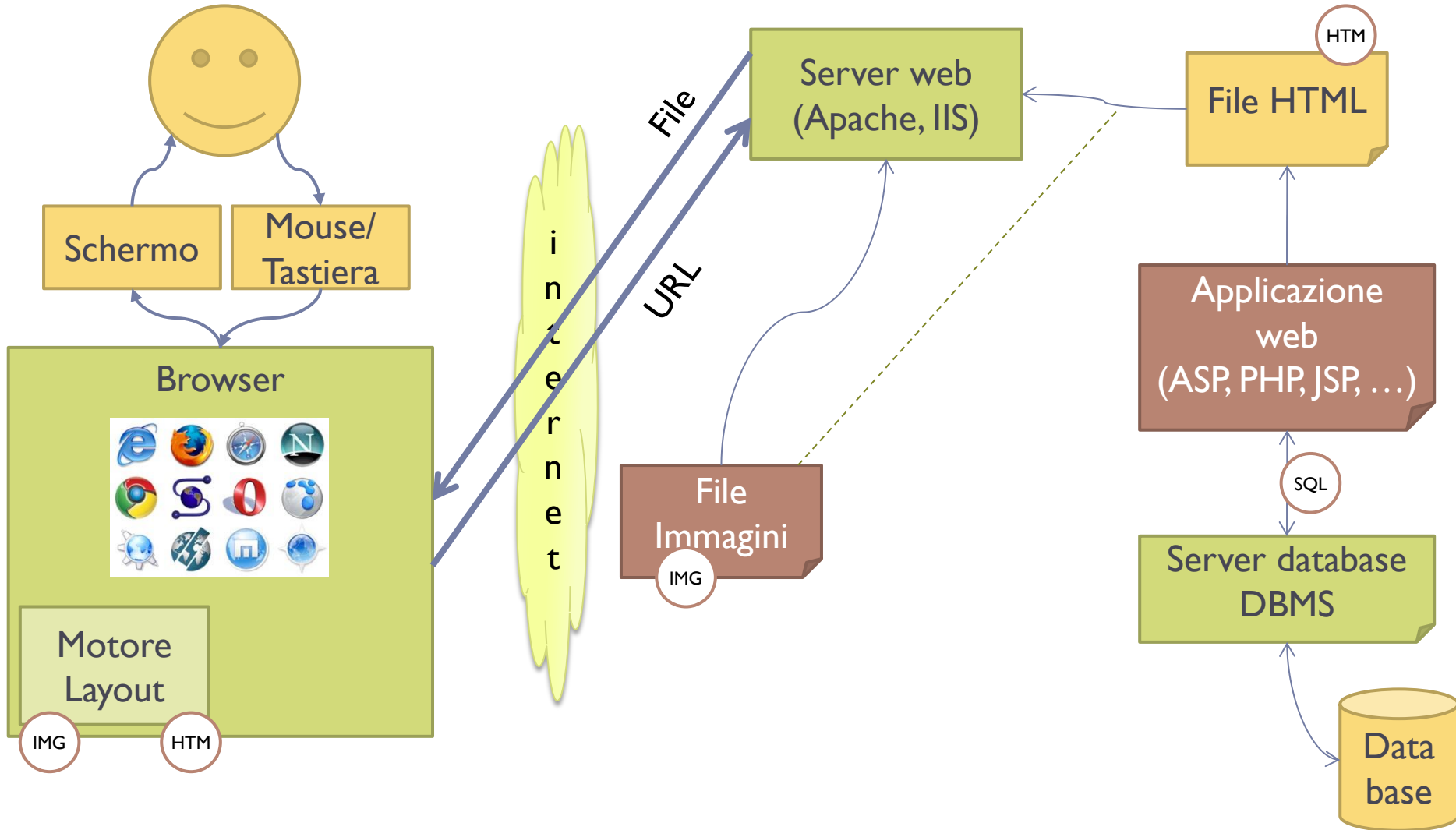
The query is sent to the db-server and a rowset containing the results is returned

```
▶ $rowset = mysql_query($query);
```

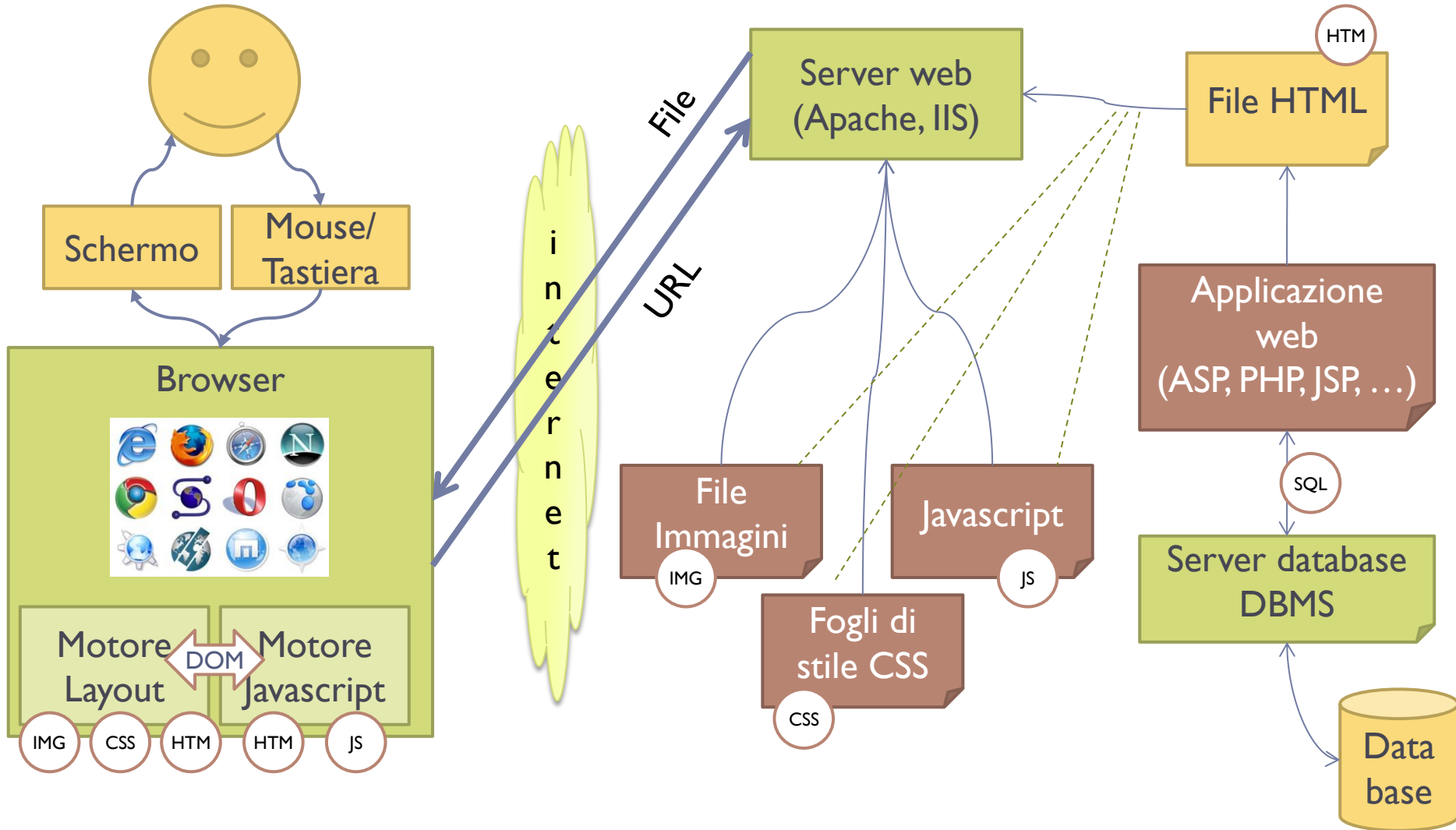
```
▶ while($row = mysql_fetch_row($rowset))
▶ {
▶ //elaborate data
▶ }
▶ ?>
```

The application elaborates the data

Architettura generale del web



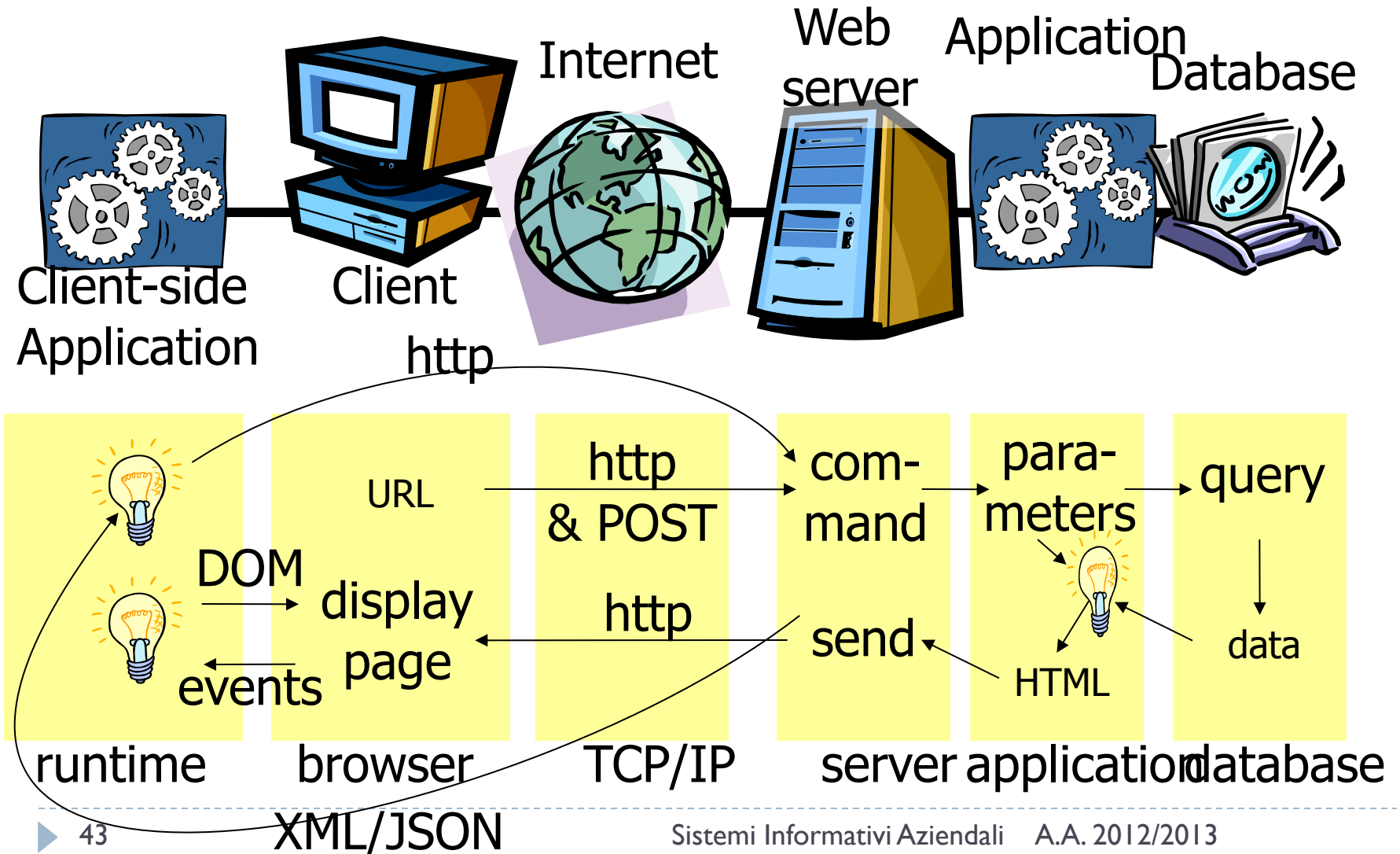
Architettura generale del web



Web 2.0

- ▶ Web applications support social interaction models
- ▶ Peer exchange and user-contributed content instead of rigid publisher/reader pattern
 - ▶ Online communities
- ▶ Rich, dynamic, interactive user interfaces
- ▶ Integration of contents across web sites (mashups)

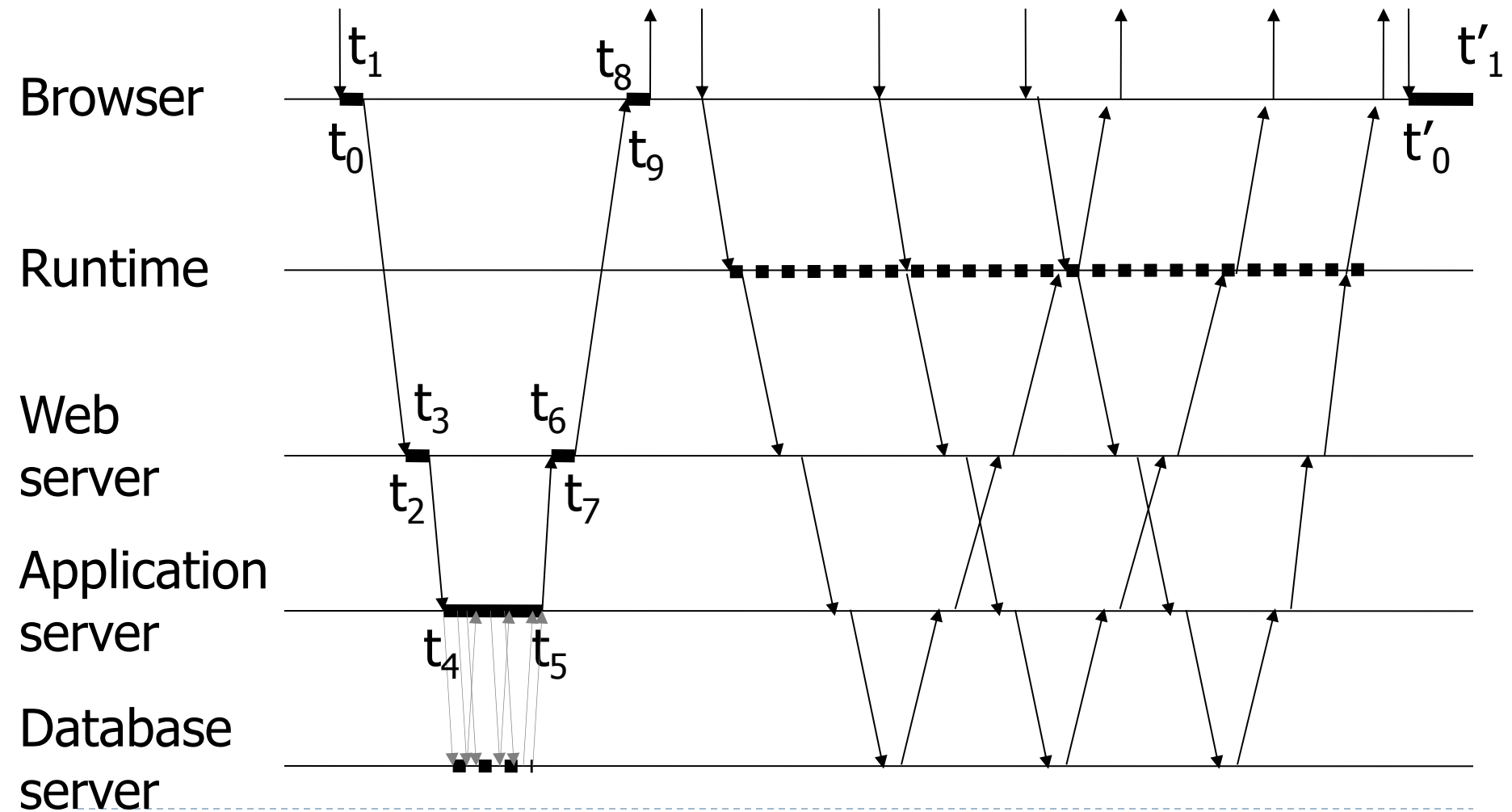
Rich-Client Asynchronous Transactions



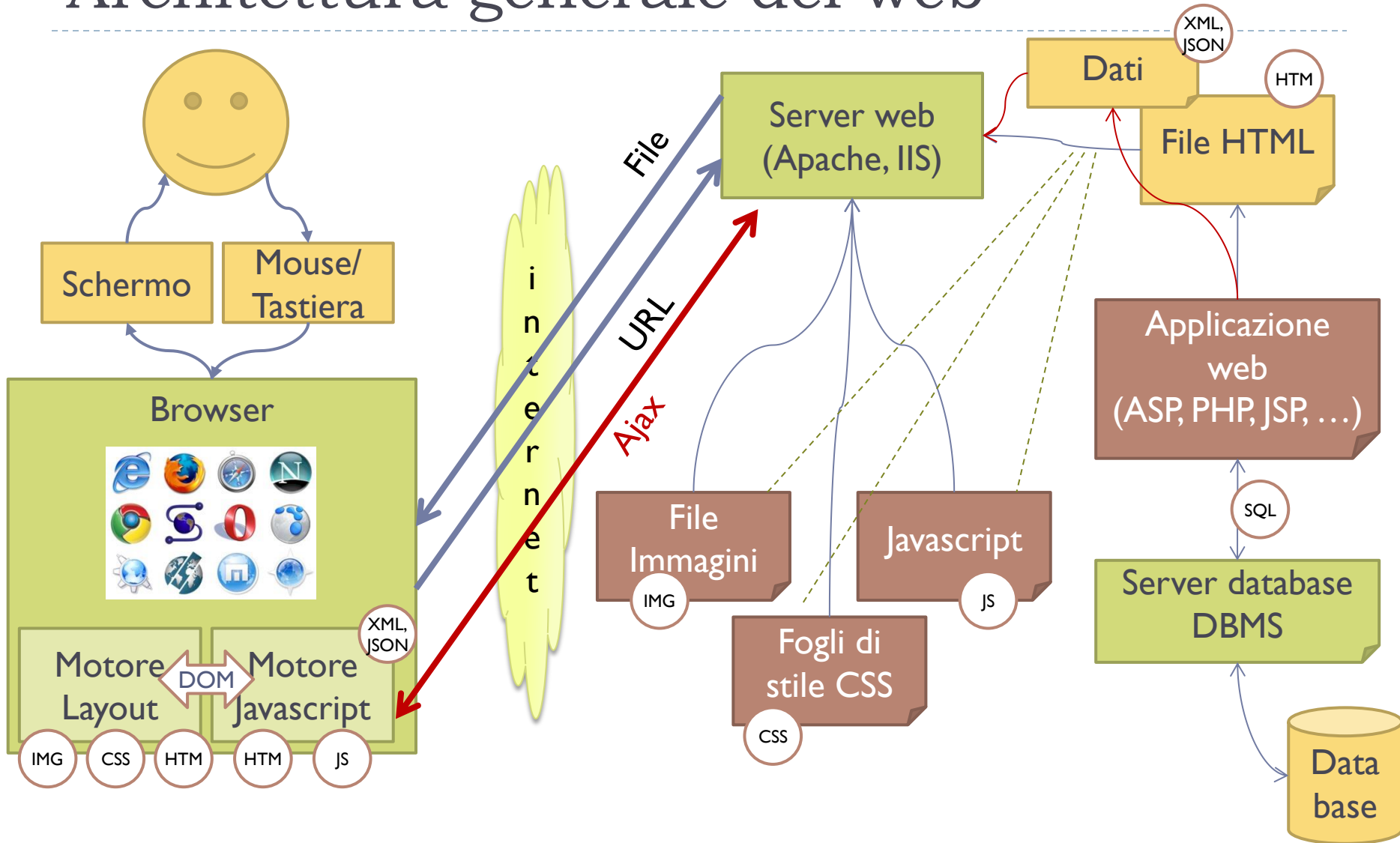
Adopted standards

- ▶ **Dynamic HTML: DOM, Javascript, CSS**
 - ▶ JavaScript, Flash to handle a runtime environment on the browser
 - ▶ DOM (XHTML Document Object Model) to allow on-the fly modification of the web page
 - ▶ CSS 2.1 to modify attribute and handle objects
- ▶ **AJAX: Asynchronous Javascript and XML**
 - ▶ XMLHttpRequest for asynchronous communication to the server
 - ▶ Data transfer formats: JSON, XML, RDF, RSS, Atom, FOAF, ...
- ▶ **Mash-up technology**

Rich-client transaction



Architettura generale del web



Architettura generale del web

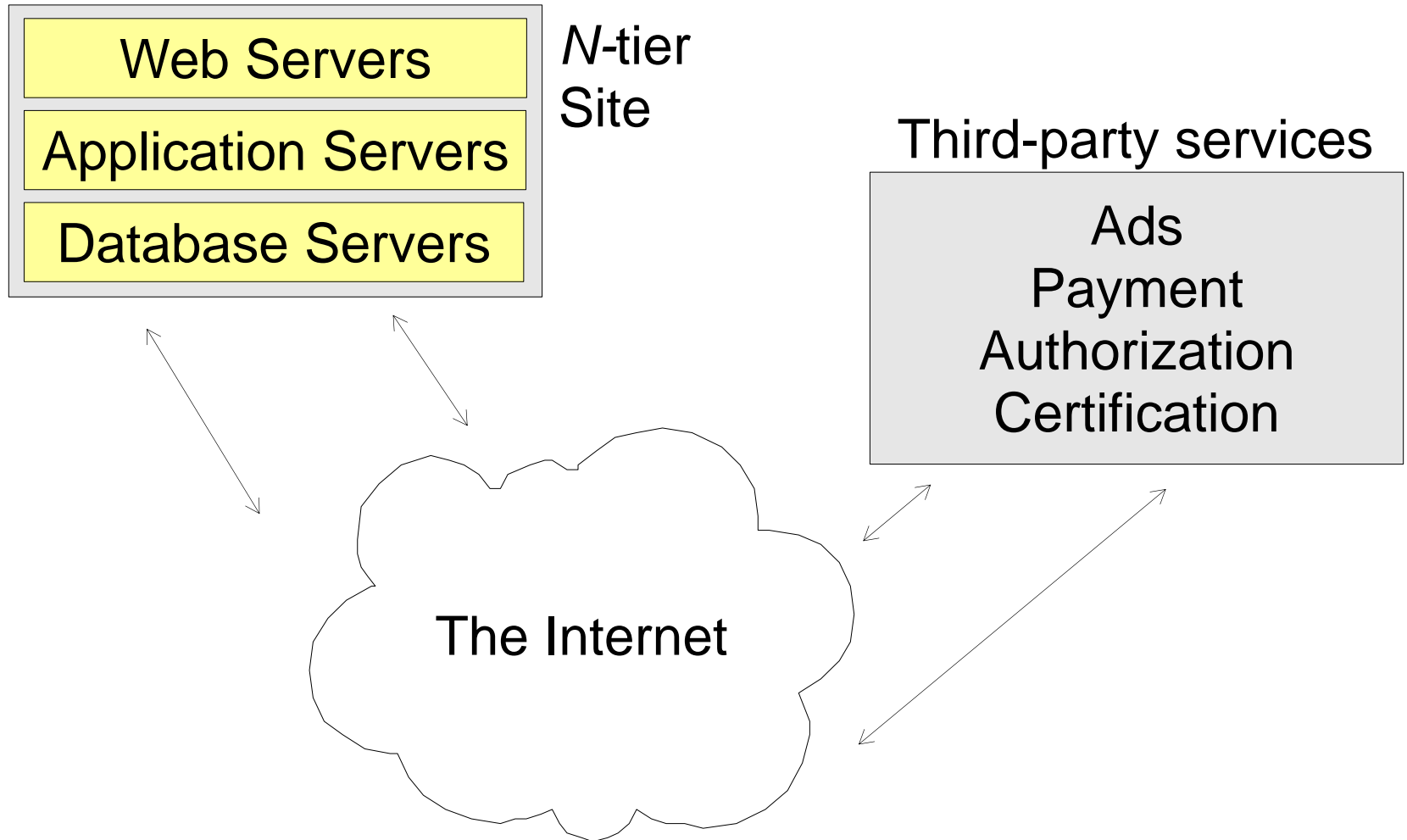


The real word is different...

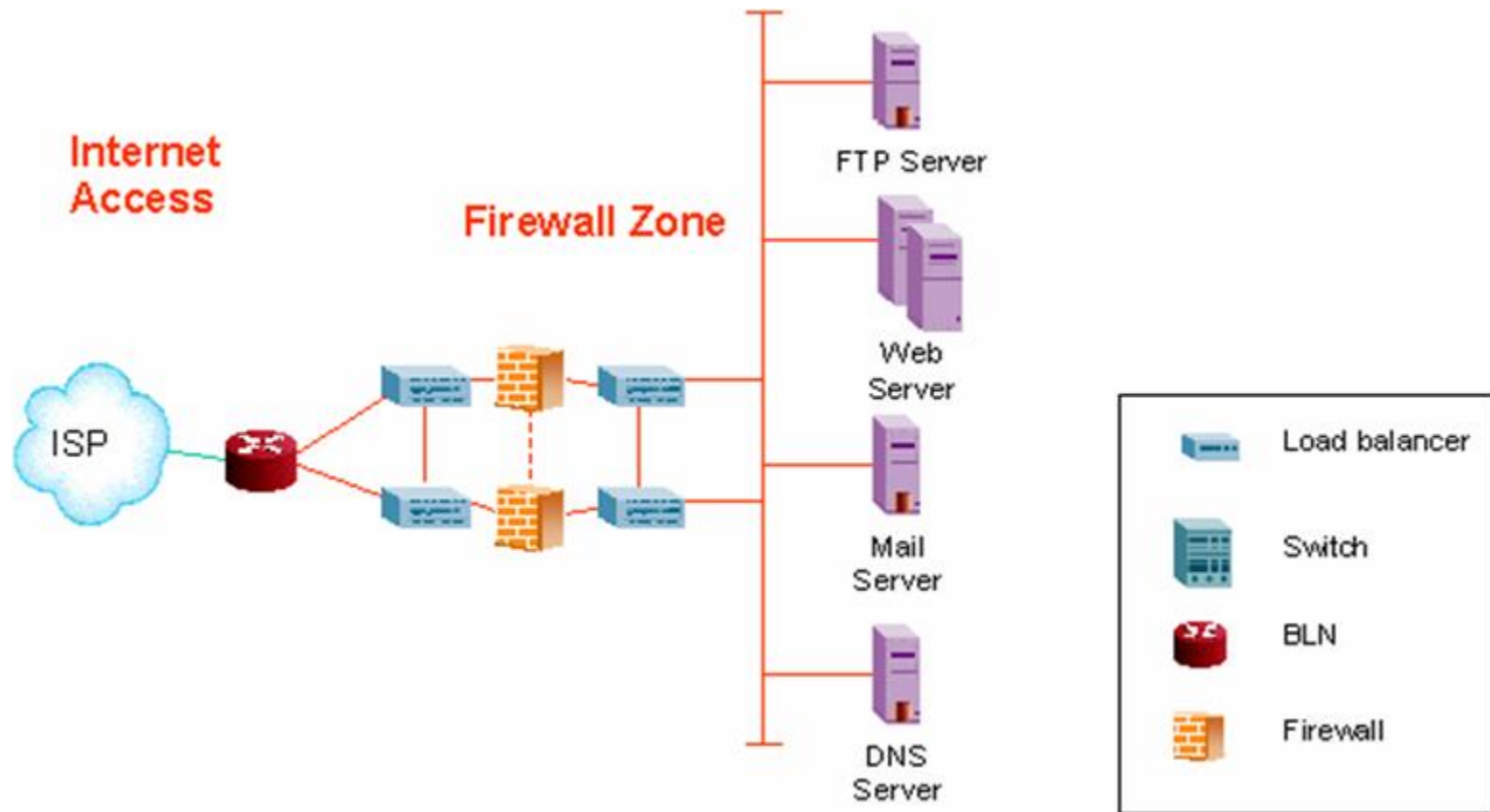
- ▶ The users
- ▶ Functionality
- ▶ Flexibility
- ▶ Portability
- ▶ Reliability
- ▶ Security
- ▶ Integrity
- ▶ Maintenance
- ▶ Performance
- ▶ Scalability
- ▶ Costs
- ▶ Maintenance
- ▶ Development times
- ▶ Interactions with existing systems
- ▶ Interactions with the “physical” world



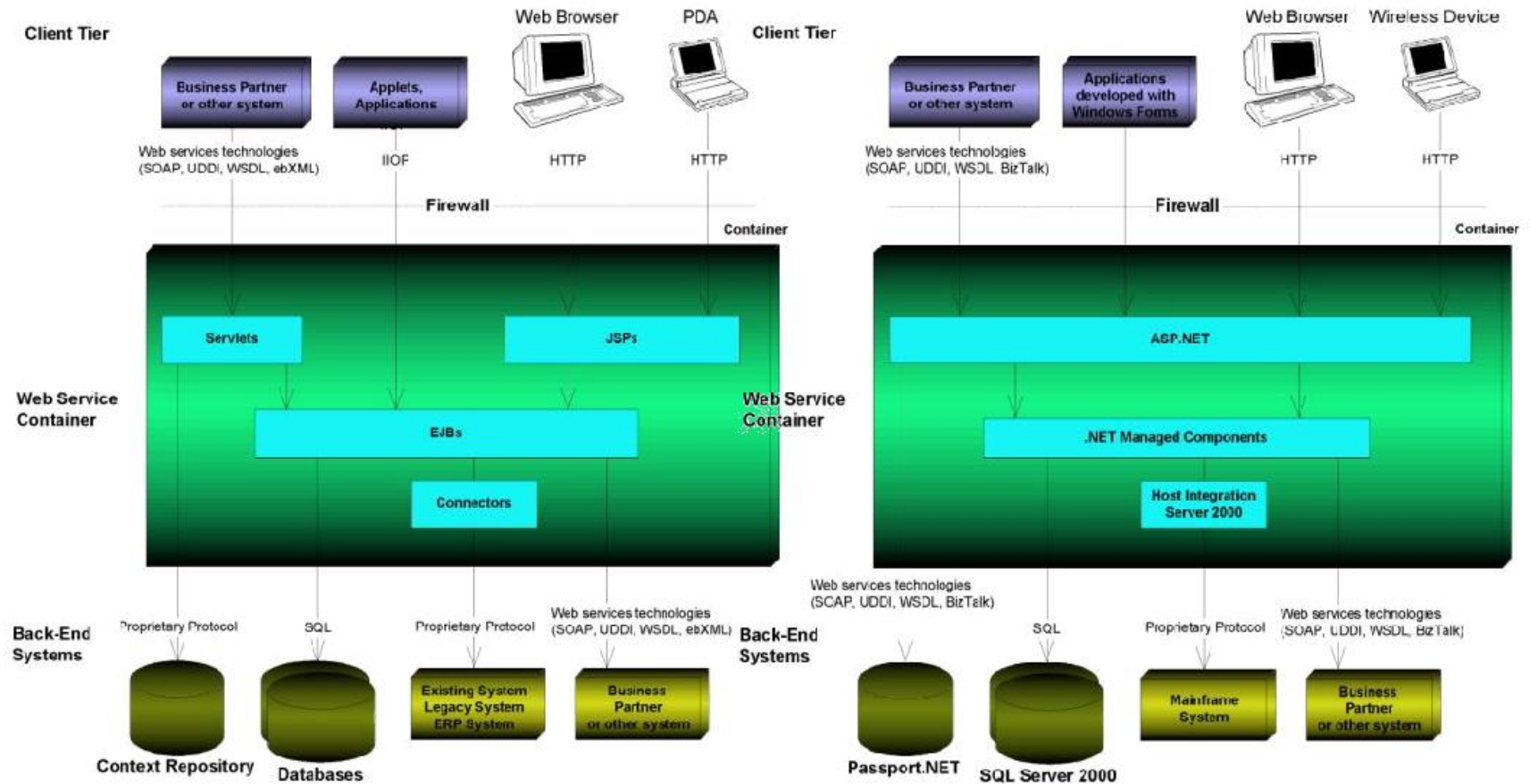
E-business architectures



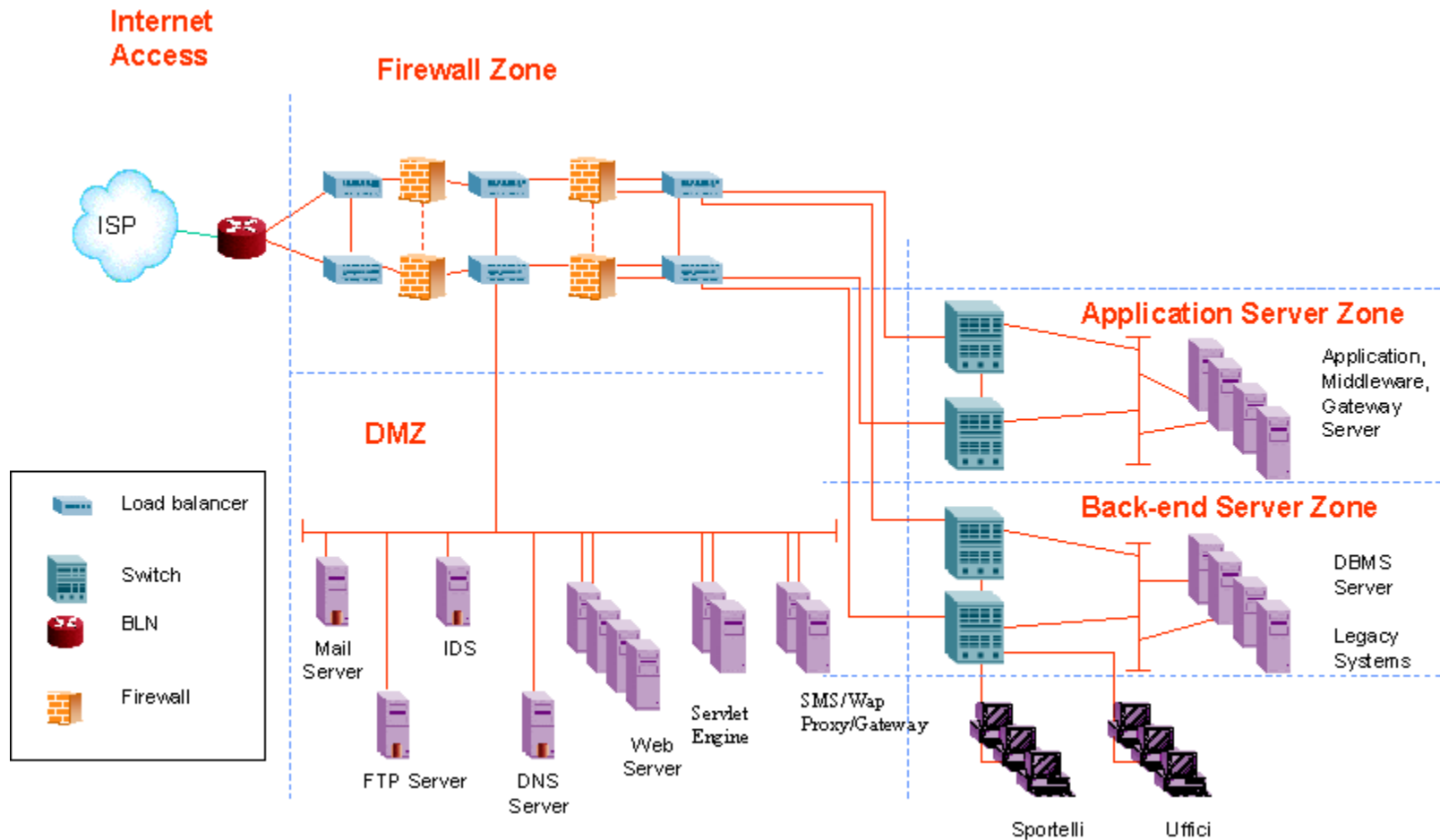
Sito informativo – completo



Ricordiamo i sistemi legacy...



Sito dispositivo – schema tipico



Interagire con altri fornitori...

- ▶ L'Application Server deve richiamare funzionalità presenti su un sito esterno
 - ▶ Servizi dispositivi (es. pagamenti)
 - ▶ Servizi informativi (es. stock quotes)
 - ▶ Servizi di sicurezza (es. autenticazione)
- ▶ La pagina web contiene delle sezioni provenienti da siti diversi
 - ▶ Approccio “a portale”, le sezioni sono indipendenti (es. iGoogle)
 - ▶ Approccio “applicativo”, le sezioni interagiscono e condividono dati (*mashup*)

Licenza d'uso



- ▶ Queste diapositive sono distribuite con licenza Creative Commons “Attribuzione - Non commerciale - Condividi allo stesso modo 2.5 Italia (CC BY-NC-SA 2.5)”

- ▶ Sei libero:

- ▶ di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare quest'opera
- ▶ di modificare quest'opera



- ▶ Alle seguenti condizioni:

- ▶ **Attribuzione** — Devi attribuire la paternità dell'opera agli autori originali e in modo tale da non suggerire che essi avallino te o il modo in cui tu usi l'opera.
- ▶ **Non commerciale** — Non puoi usare quest'opera per fini commerciali.
- ▶ **Condividi allo stesso modo** — Se alteri o trasformi quest'opera, o se la usi per crearne un'altra, puoi distribuire l'opera risultante solo con una licenza identica o equivalente a questa.



- ▶ <http://creativecommons.org/licenses/by-nc-sa/2.5/it/>