Basi di dati e Web
modulo: siti web centrati sui dati

Alberto Belussi

anno accademico 2007/2008

WEB

La tecnologia del World Wide Web (WWW) costituisce attualmente lo strumento di riferimento per la diffusione e acquisizione di informazioni sotto forma di documenti digitali.

Tale tecnologia si basa sul sistema di comunicazione planetario chiamato INTERNET.

INTERNET

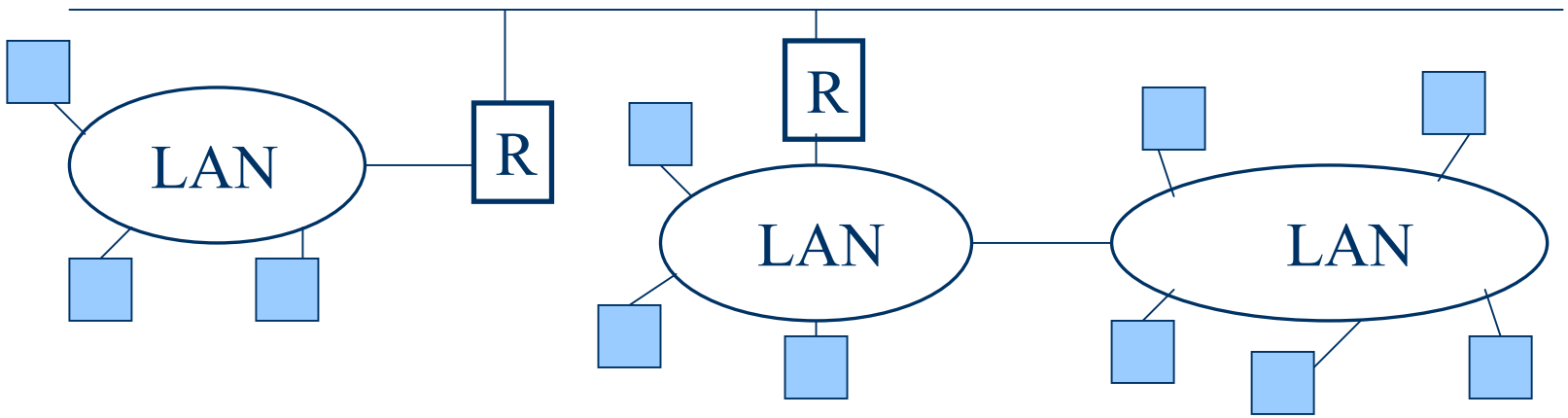
E’ una federazione di reti che comunicano attraverso la famiglia di protocolli TCP/IP.

Riferimenti storici:

- ARPANET (dipartimento della difesa USA): è stata la prima rete a commutazione di pacchetto basata sul protocollo IP
- NSFnet (national science fundation net)

Struttura attuale della rete

Internet è attualmente un insieme di reti locali (LAN) collegate tra loro e/o con altre porzioni di rete attraverso ROOTER (calcolatori dedicati alla gestione della rete) o dispositivi simili.



Struttura attuale della rete

I nodi della rete sono i calcolatori connessi ad una delle reti federate.

Ogni calcolatore ha un indirizzo univoco, detto indirizzo IP, che ha la seguente forma:

157.27.252.11

Tale indirizzo può assumere anche un formato simbolico:

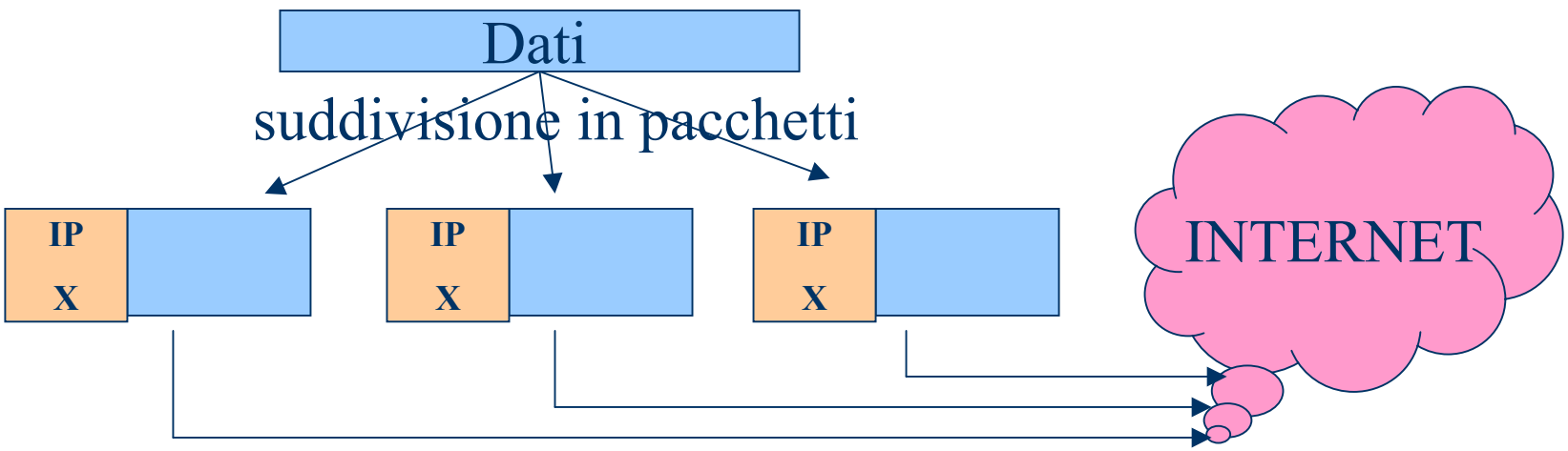
arena.sci.univr.it

Struttura attuale della rete

Come avviene la comunicazione sulla rete?

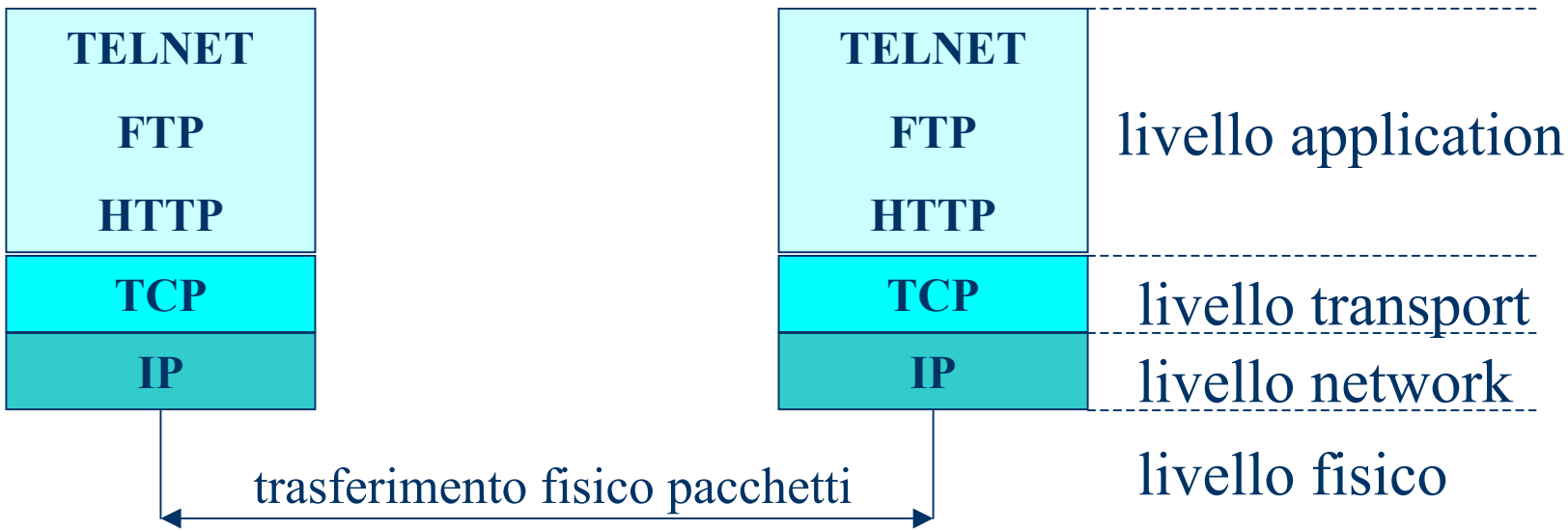
Meccanismo di base:

- commutazione di pacchetto
- scambio di informazione organizzata in pacchetti



Struttura attuale della rete

Livelli del Software di Rete secondo il protocollo TCP/IP.



Struttura attuale della rete

Protocollo TCP/IP:

- Il protocollo IP (Internet Protocol) garantisce la spedizione del pacchetto a destinazione
- Il protocollo TCP (Transmission Control Protocol) gestisce la suddivisione in pacchetti dell'informazione da spedire, garantisce la spedizione senza errori e gestisce il riassemblaggio dei pacchetti nell'ordine corretto.
- I protocolli del livello application gestiscono l'interazione via rete tra due calcolatori (nodi della rete). Si basano sul paradigma CLIENT-SERVER.

Protocollo HTTP

Il protocollo HTTP gestisce i servizi collegati al WEB.

L'informazione disponibile in rete attraverso il protocollo HTTP è organizzata in documenti digitali detti IPERTESTI.

HTTP significa infatti:

HYPER TEXT TRANSFER PROTOCOL.

IPERTESTI

IPERTESTO: è un documento con struttura non sequenziale, costituito da varie parti fra loro collegate. Tali legami sono detti LINK e consentono di navigare nell'ipertesto.

Le singole parti di un ipertesto si dicono PAGINE WEB.

L'insieme di pagine web gestite da un unico server HTTP della rete costituiscono un SITO WEB.

Linguaggio HTML

Il linguaggio usato per specificare IPERTESTI si chiama: HTML (HYPER TEXT MARKUP LANGUAGE).

Ogni pagina web diventa un file HTML; l'HTML consente di specificare:

- La formattazione del testo per la presentazione
- La struttura della pagina (sezioni, tabelle, liste, ecc.)
- I legami con altre pagine (LINK) o con sottoparti della pagina
- Il contenuto informativo della pagina.

Linguaggio HTML

LINK in HTML

Come è possibile identificare in modo univoco un documento (risorsa) nel World Wide Web?

Si introduce il concetto di URL (Uniform Resource Locator)

protocollo://server/risorsa

protocollo: ftp, mailto, http

server: ip o nome simbolico del nodo

risorsa: [path]nome_file[parametri]

Linguaggio HTML

LINK in HTML:

- Link esterno:

   testo

- Link interno:

   testo

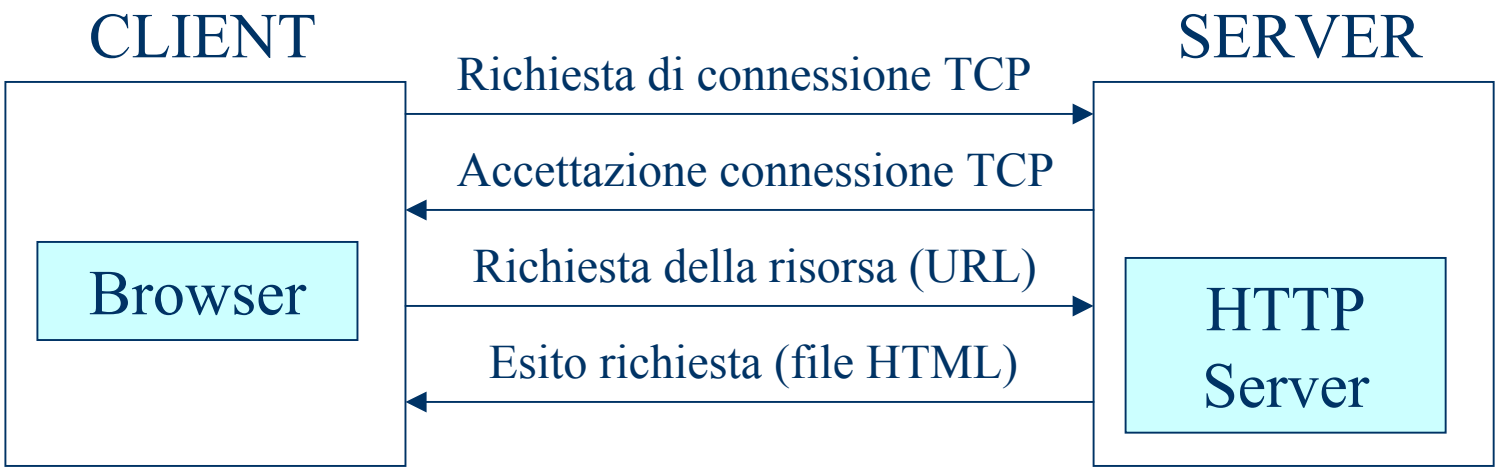
  …

  …

   testo 2

Protocollo HTTP

Consente lo scambio tra client (browser) e server (HTTP server) di pagine web (file HTML).



Protocollo HTTP

Il protocollo HTTP è “state-less” (senza stato): ogni connessione è completamente indipendente dalle altre. Il server non è in grado di mantenere informazioni sulle connessioni passate.

Richieste HTTP:

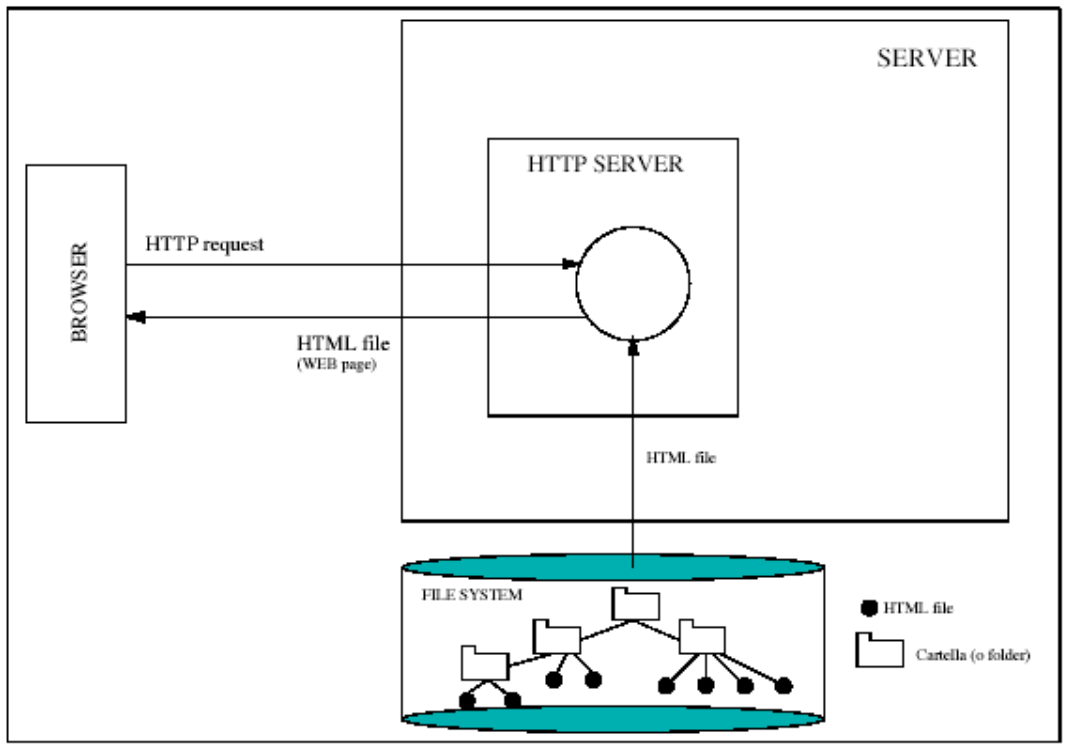
- GET: richiede una pagina (URL) inviando i parametri in modo esplicito in coda all’URL.
- POST: richiede un pagina (URL) con invio dei parametri in modo non visibile nell’URL.
- PUT, HEAD, DELETE, OPTIONS.

Gestione delle pagine WEB

- Pagine statiche
  - Sono File HTML memorizzati nel file system del server
- Pagine dinamiche:
  - Vengono generate “on the fly” attraverso l’esecuzione di programmi sul server che interagiscono con un DBMS che contiene i dati da mostrare nelle pagine.

Pagine statiche

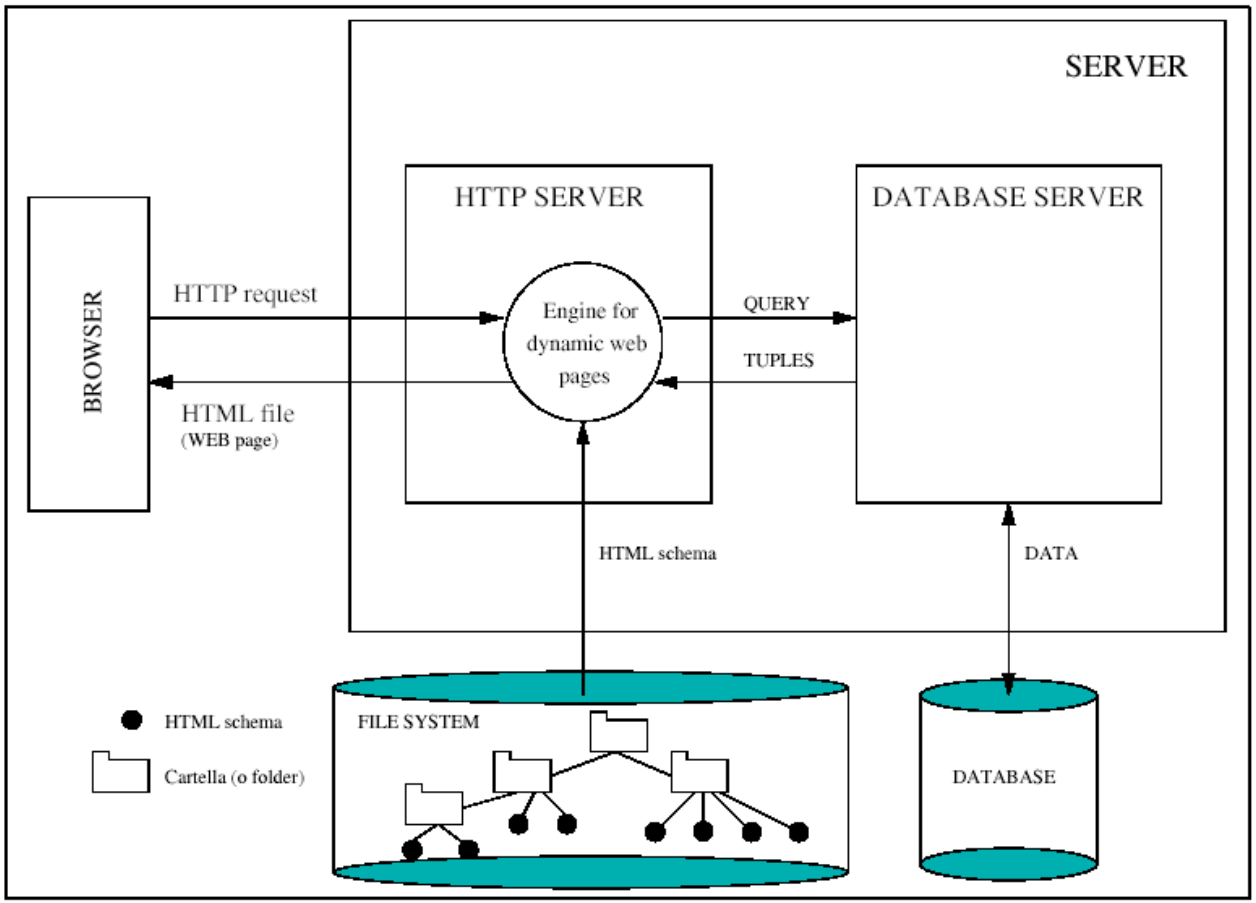
In questo caso le pagine vengono generate da un programmatore HTML e vengono rese disponibili sul server HTTP come file HTML.



Pagine dinamiche

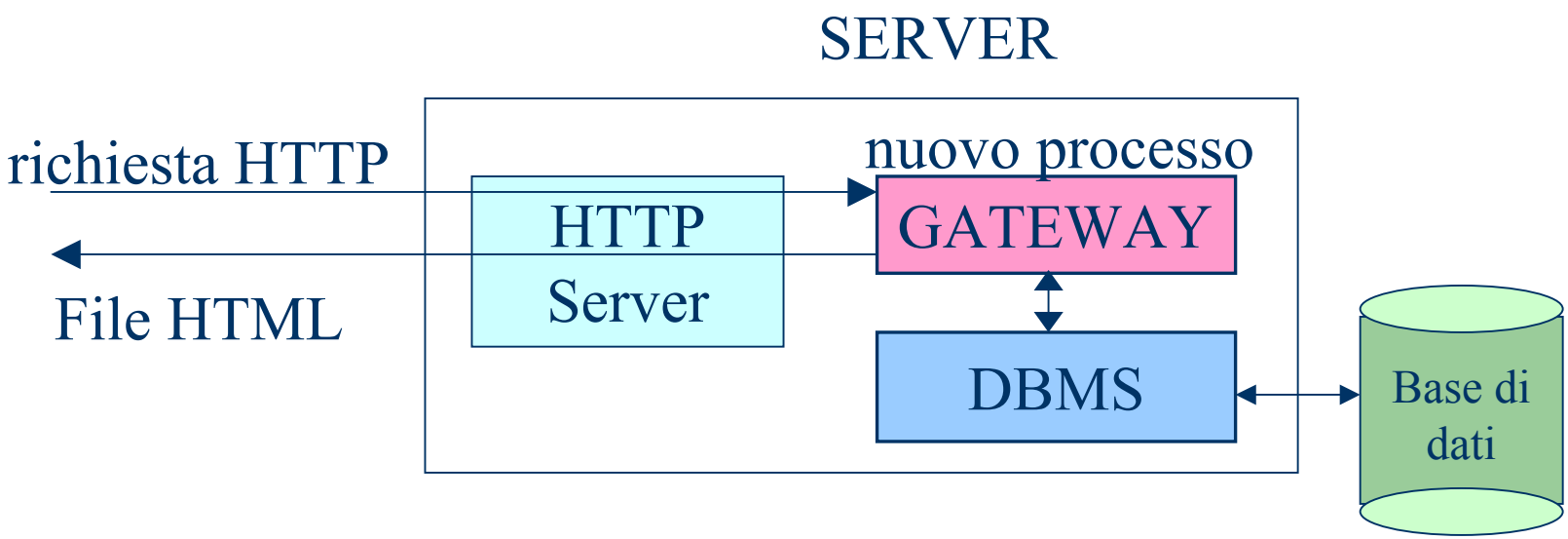
In questo caso le pagine vengono generate dal server in base ad altre applicazioni esterne o integrate nel server HTTP. La generazione dinamica delle pagine consente di estrarre da sistemi DBMS l'informazione da presentare. Tali DBMS raccolgono gli aggiornamenti dei dati prodotti dai processi "core" dell'organizzazione per cui si sta allestendo il sito web.

Pagine dinamiche: schema



Tecniche per realizzare pagine web dinamiche

Esistono attualmente diverse tecniche, ma tutte derivano dalla prima tecnica applicata storicamente: CGI (Common Gateway Interface)



Tecniche per realizzare pagine web dinamiche

LIMITI DELL'APPROCCIO CGI

- Richiede la creazione di un **nuovo processo per ogni richiesta**: tale creazione richiede tempo non trascurabile e necessita inoltre di uno specifico spazio di indirizzamento in memoria virtuale
- Il processo CGI attivato richiede necessaria-mente una nuova connessione al DBMS e la sua corrispondente chiusura a fine programma.

Tecniche per realizzare pagine web dinamiche

Tecniche alternative a CGI

- Soluzioni basate sull'estensione del server HTTP che diventa un application server: questo ad esempio è il caso della tecnologia **Servlet di Sun**.

- Soluzioni basate sull'immersione di codice nei file HTML: **Java Server Pages (jsp)**, Active Server Pages (asp), Hypertext Preprocessor (php).

Tecniche per realizzare pagine web dinamiche

SERVLET (Sun): si basa sul linguaggio JAVA.

Vantaggi rispetto a CGI:

- Ogni richiesta genera un thread e non un processo;
- È possibile mantenere connessioni aperte con il DBMS tra una richiesta e l'altra
- Portabilità JAVA

Tecniche per realizzare pagine web dinamiche

Template Systems: si basano su versioni di HTML estese con tag per l'inserimento di codice.

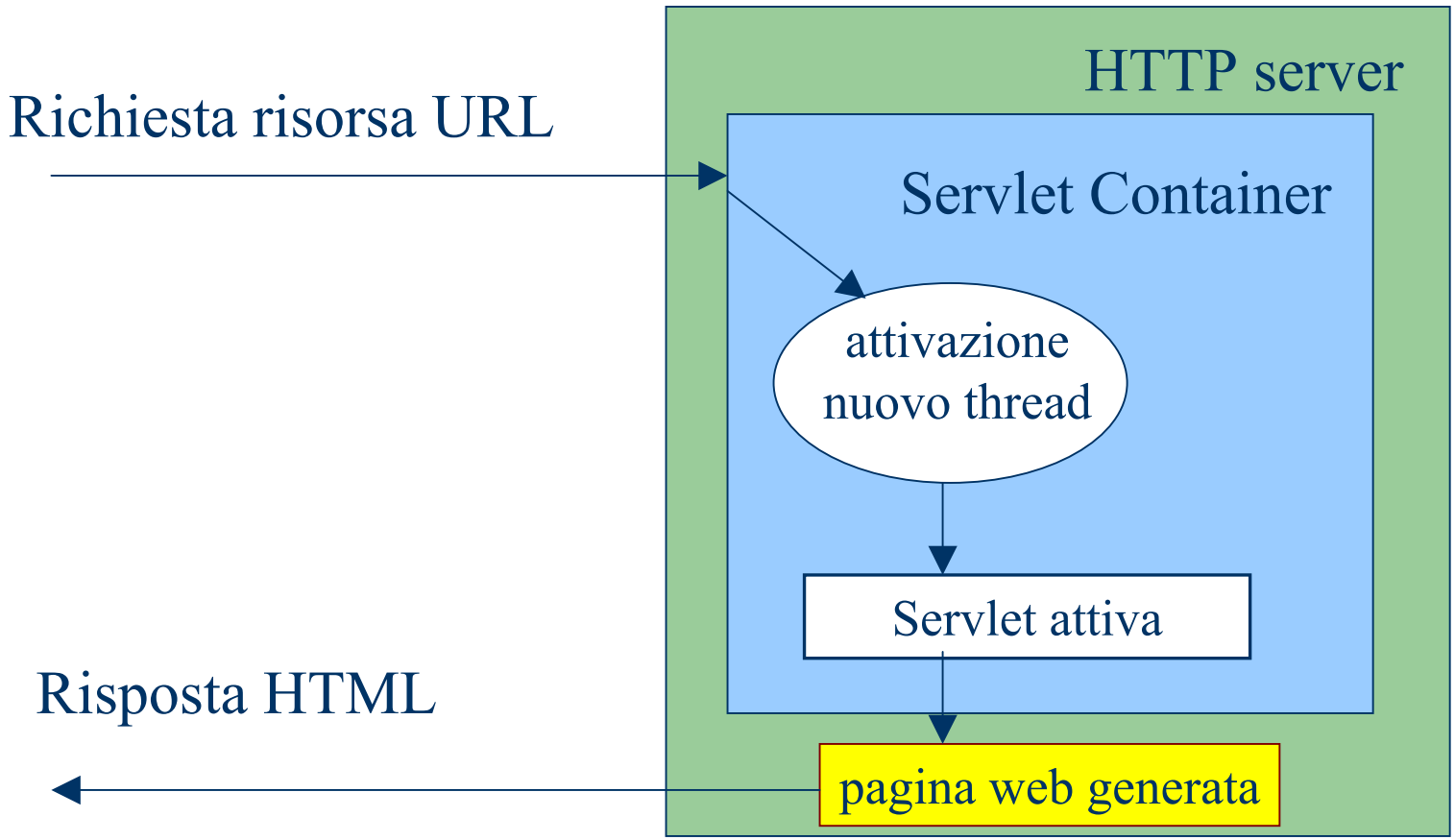
Vantaggi rispetto a CGI:

- Tutti i vantaggi delle servlet;
- Il codice per la generazione delle parti dinamiche viene inserito con opportuni tag nel file HTML che rappresenta invece la parte statica della pagina WEB. Evita al programmatore di generare dinamicamente anche le parti fisse.

Servlet

Richiede l’installazione di un SERVLET ENGINE da integrare nel server HTTP.
Ogni servlet engine predispone un albero di direttori dove vanno inseriti i file contenenti le servlet, le classi java di supporto, i file html statici, ecc..
Servlet engine usato in laboratorio: TOMCAT

Servlet engine: struttura



Servlet container

E’ un processo sempre attivo che implementa una

JAVA VIRTUAL MACHINE

Servlet

Ogni servlet è una classe JAVA ottenuta estendendo la classe `HttpServlet`.
Esempio di servlet semplice:

```
import java.oi.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class HelloWorld extends HttpServlet {
      public void doGet (
            HttpServletRequest request,
            HttpServletResponse response)
      throws ServletException, IOException  {
```

Servlet

```
  {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();
      String docType = "<!DOCTYPE HTML …";
      out.println(docType +
            "<html>\n"+
            "<head><title>Hello World</title>"+
            "</head>\n"+
            "<body>\n"+
            "<h1>Hello World</h1>\n"+
            "</body></html>");
  } // end doGet
} // end servlet
```

Servlet

Note sui parametri di `doGet` (`doPost`)

`request`: consente di accedere a tutte le informazioni relative alla richesta HTTP che ha invocato la servlet.

Alcuni metodi di `request`

- `request.getParameter(NOME_PARAMETRO)`
  restituisce: una stringa (String) che rappresenta il valore della prima occorrenza del parametro `NOME_PARAMETRO`; la stringa vuota se il parametro esiste ma non ha valore; NULL se il parametro non esiste.
- `request.getParameterValue(NOME_PARAMETRO)`
  restituisce un array di stringhe contenenti i valori di tutte le occorrenze del parametro `NOME_PARAMETRO`; un array con una stringa vuota se il parametro esiste ma non ha valore; NULL se il parametro non esiste.
- `request.getParameterNames()`
  restituisce un array di stringhe contenenti i nomi dei parametri della richiesta HTTP.

Servlet

`response`: gestisce l'invio dell'output (HTML) al server HTTP.

Alcuni metodi di `response`

- `response.getWriter()`
  restituisce un oggetto della classe `PrintWriter` al quale è possibile inviare stringhe di caratteri con il metodo `println`.
- `response.getBufferSize()`
  restituisce la dimensione del buffer degli oggetti `PrintWriter`.
- `request.setBufferSize()`
  consente di ridefinire la dimensione del buffer degli oggetti `PrintWriter`.