

Impianti Informatici

 POLITECNICO DI MILANO



Web application



La diffusione del World Wide Web

Supporto ai ricercatori

- Organizzazione documentazione
- Condivisione informazioni

Scambio di informazioni di qualsiasi natura

- Chat
- Forum

Intranet

- Stessi standard usati in Internet



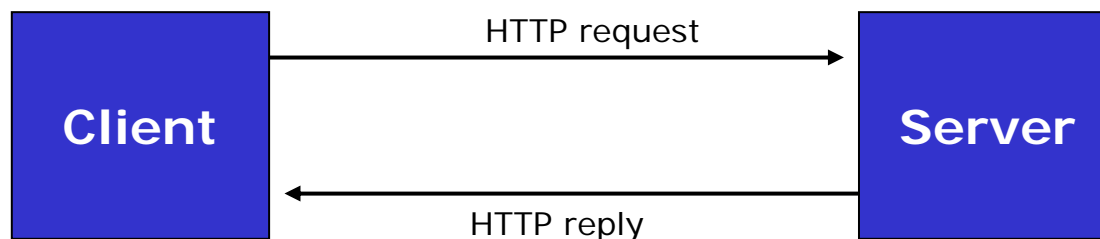


Applicazioni Web

Una qualsiasi applicazione client/server che:

- Usa HTTP come protocollo di comunicazione
 - Effettua richieste Http
 - Riceve risposte Http
 - Usa i browser standard come client (ma non necessariamente)

Solitamente tale applicazione funziona da “interfaccia” per l’accesso ad una base di dati





Applicazione standard vs Web Application

Applicazione Standard

- Protocollo proprietario
- Protocollo non fissato
- Client ad-hoc
 - Potenzialità di interazione
 - Potenzialità di visualizzazione

Web Application

- HTTP
- Stack TCP/IP
- Browser
 - Interazione limitata
 - Visualizzazione semplice

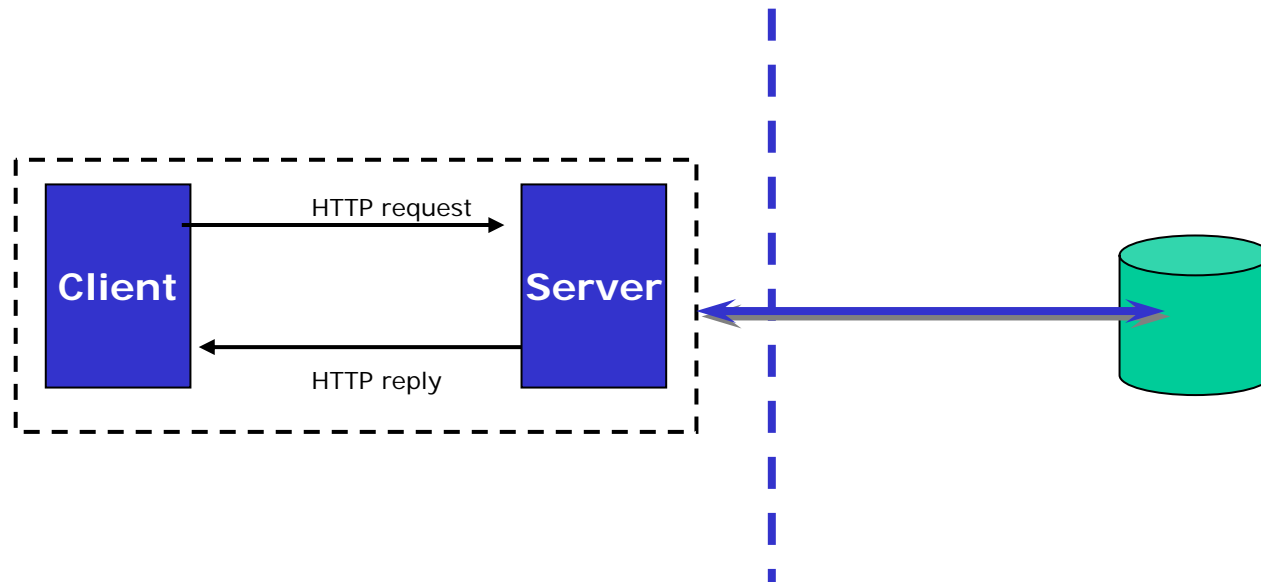


Generazione dinamica dei contenuti

Pagine HTML statiche

Web Application: pagine dinamiche

- Interfaccia verso un database
- Esegue query SQL
- Formatta i dati per visualizzarli all'utente





Esempio: Web Mail

È un gateway applicativo

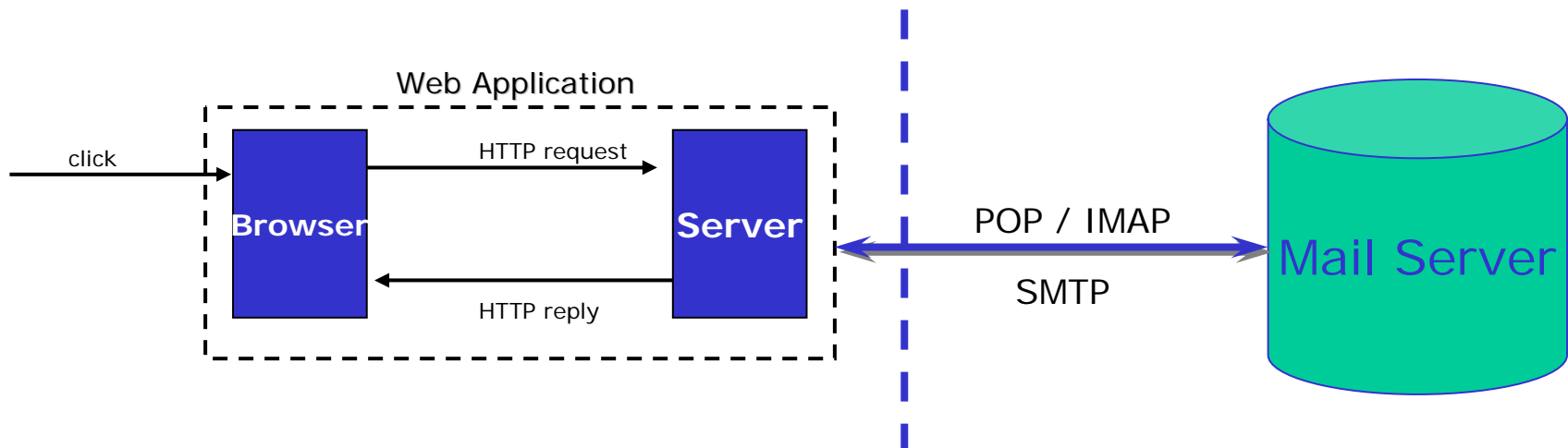
Interfaccia con il server mail

- POP / IMAP
- SMTP



L'utente interagisce solo con il browser

- La Web Application interroga il Mail Server e compone la pagina





Esempio: mappa stradale

Interazione con una base di dati

- Il DB contiene i possibili percorsi, pesati ad esempio in base a traffico, tipo di strada,...
- Calcola il percorso ottimale (il più breve, il più semplice,...)
 - Algoritmo complesso
- Genera le istruzioni necessarie per seguire l'itinerario
- Formatta una pagina contenente le indicazioni da seguire e la mappa stradale



Applicazioni Web: peculiarità

Di facile utilizzo

- si utilizza un client usato comunemente per la navigazione di pagine web

Integrazione immediata con Internet

- segue lo standard TCP/IP e utilizza l'HTTP come protocollo applicativo

Indipendente dalla piattaforma (TCP/IP)

Interoperabilità tra hw/sw di costruttori vari

Economica

- bassi costi di produzione del relativo sw applicativo

Architettura altamente scalabile

- consente di scrivere anche applicazioni distribuite

Modulare (divisa in livelli logici)



Limitazioni del protocollo HTTP

Stateless

Transazioni difficili

- Carrello on-line
 - Log-in
 - Selezione prodotti
 - Pagamento
 - Log-off

Possibili soluzioni:

- Cookie
- URL rewriting

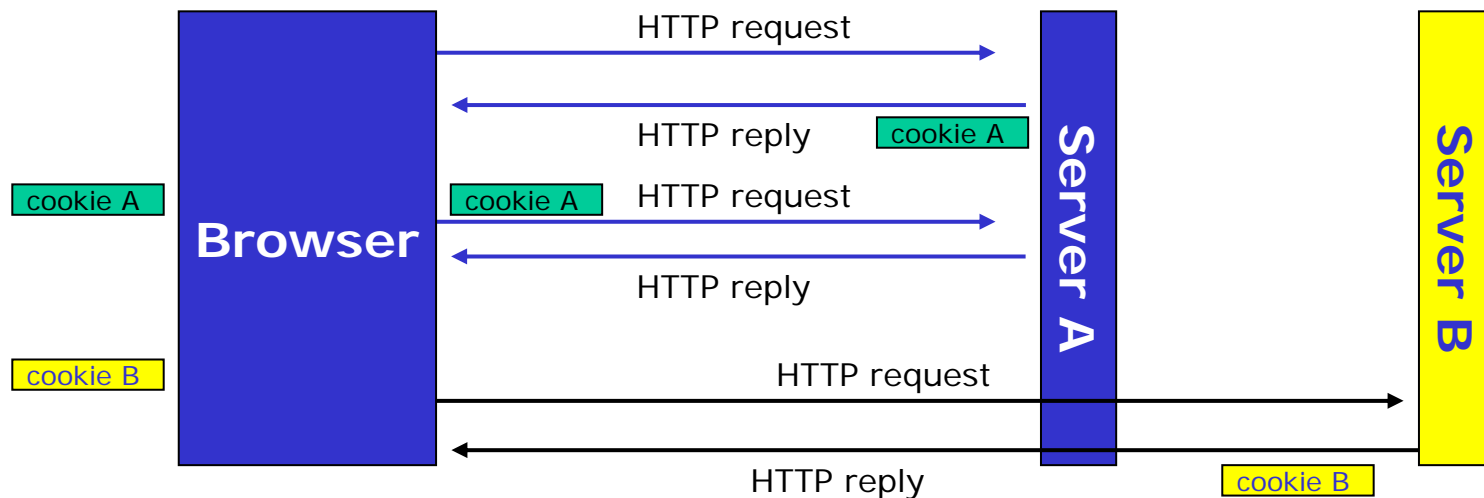




File di testo salvato sul lato client.

Viene inviato tramite HTTP

- Alla prima richiesta HTTP, il server invia un cookie al client
- Alle successive richieste allo stesso server, il client allega il cookie ricevuto
- Cookie più recenti sovrascrivono i precedenti





Cookie e sessione

Cookie

- Informazioni standard sul server
 - Data
 - Durata cookie
 - Data scadenza
- Informazioni dipendenti dalla Web Application

Sessione

- Connessione logica tra client e server web
- Lo stato è memorizzato nel server
- Il cookie viene utilizzato per identificare la sessione



URL rewriting

URL dinamici

- Con ID della sessione

Vengono riscritti gli URL all'interno delle pagine da inviare al browser

Ogni richiesta HTTP include l'ID della sessione

- Il server separa ID da URL effettivo



Thin Client

Il client svolge funzioni limitate

La logica è tutta sul lato server

Il client si occupa della visualizzazione e dell'interfaccia utente

E il Browser Web?

- Non troppo “leggero”, dato l'uso di plug-in (ActiveX, JavaScript,...)

Sovraccaricano il lato server

- Occorre connettersi al server per qualsiasi operazione



Fat (Thick) Client

Sul client vengono eseguite funzioni degli applicativi

- Il client sw è una parte dell'applicazione che va installata su ogni macchina

Più difficili da mantenere

- Molti client
- Cambiamenti nel server hanno effetti sul client

È possibile realizzare interfacce grafiche complesse

Non richiede un'interazione continua col server

- Riduce il carico sul lato server
- Aumenta il carico sul client



Browser Fat o Thin?

Browser testuali

- HTTP/HTML

Browser con grafica

- Uso improprio di HTML
- Plug-in

Ritorno ai Browser Thin

- Navigazione da dispositivi eterogenei
 - Computer
 - Palmari
 - Telefonini

Architetture a più livelli

Impianti Informatici

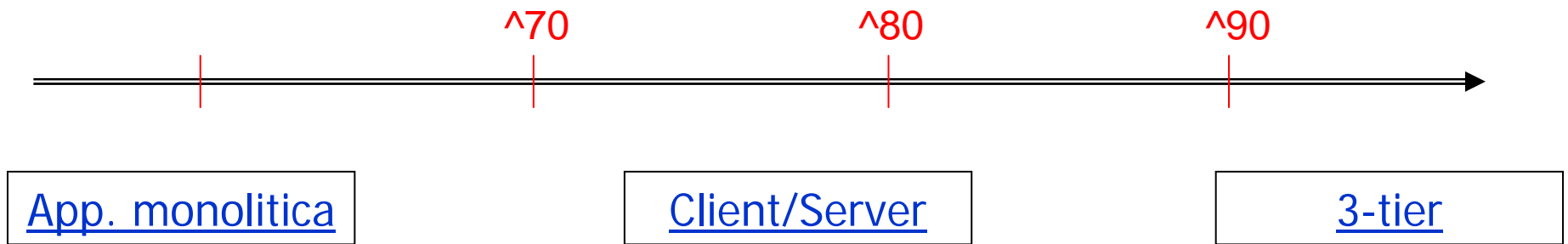
 POLITECNICO DI MILANO



Web application



Evoluzione delle applicazioni





Applicazioni Monolitiche

Popolare con i mainframe

- Adatti per processi stand-alone

Pezzi di codice indivisibile

Controlla l'intera applicazione

- Logica applicativa
- Gestione e memorizzazione dei dati
- Interfaccia utente





Applicazioni Client/Server

Passaggio da mainframe a server, pc, workstation:

- Più economici
- Più piccoli

Diffusione Database RDBMS

Architettura monolitica non flessibile

Paradigma client/server

- Server (backend):
 - Gestione di database
 - Elaborazione dei dati
- Client (frontend):
 - Interfaccia utente



Client/Server: pro e contro

Vantaggi:

- Maggiore scalabilità
- Carico computazionale distribuito sui client
- Sviluppo più veloce di applicazioni che accedono agli stessi dati

Svantaggi:

- Traffico di messaggi intenso
- Logica di business inclusa sia nel frontend che nel backend
 - Client e server dipendenti l'uno dall'altro

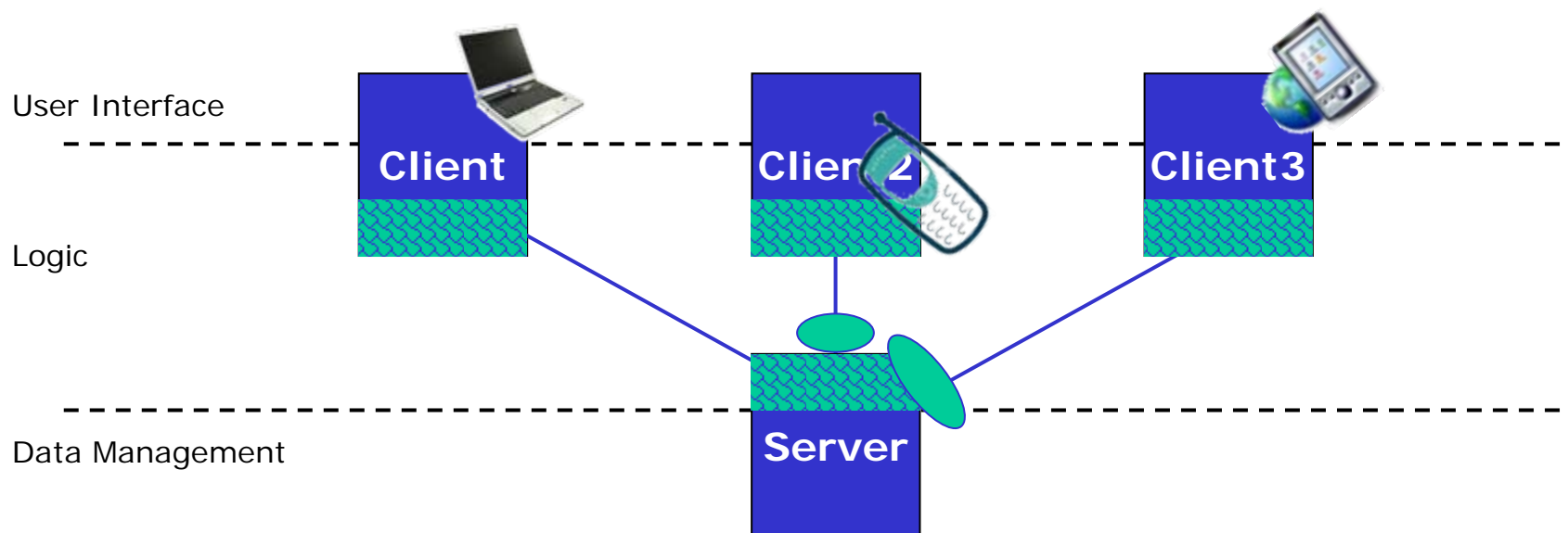
Client/Server: business logic

Problema: business logic sia nel client che nel server
Server accessibile da più device

- Adapter

Modifiche nella business logic impattano su:

- Server
- Molteplici client

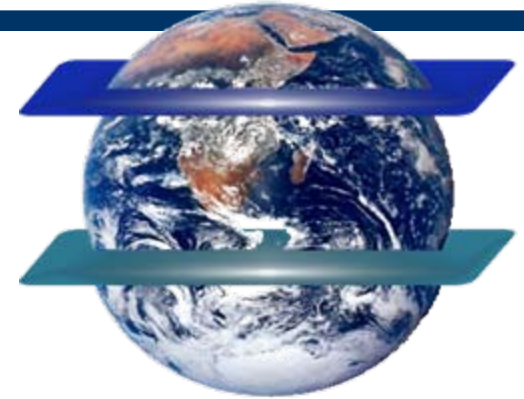




Applicazioni 3-tier

Tre livelli:

- Data Layer
 - Gestione dei dati
 - DBMS
 - File XML
- Business Layer
 - Business Logic
 - Processamento dati
- Presentation
 - Interfaccia utente
 - Presentazione Dati
 - Servizi



Indipendenza dei layer

- Ogni livello ha obiettivi e vincoli propri
- Nessuna assunzione sugli altri livelli



Web Application: Presentation Layer

Web browser

- Visualizzare gli aspetti dell'applicazione
- Interfaccia utente
- Invia richieste HTTP e interpreta le risposte



Web server. (?)

- riceve le richieste HTTP inviate dal browser
- inoltra la richiesta ai livelli successivi
- impagina la risposta e la invia al browser

Web Application: Application Layer

L'insieme dei programmi che eseguono le azioni richieste (J2EE, Enterprise Java Beans,...)

Gestisce

- Dati dell'applicazione
- Stato corrente
- Assicura integrità dati
- Gestisce errori o eccezioni



Esempi di operazioni:

- Query
- Elaborazione dei dati
- Report

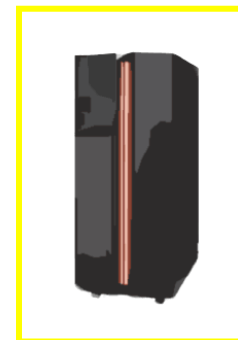
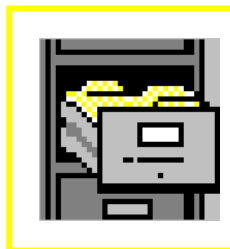
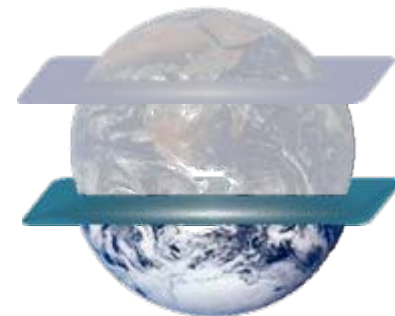




Web Application: Data Store Layer

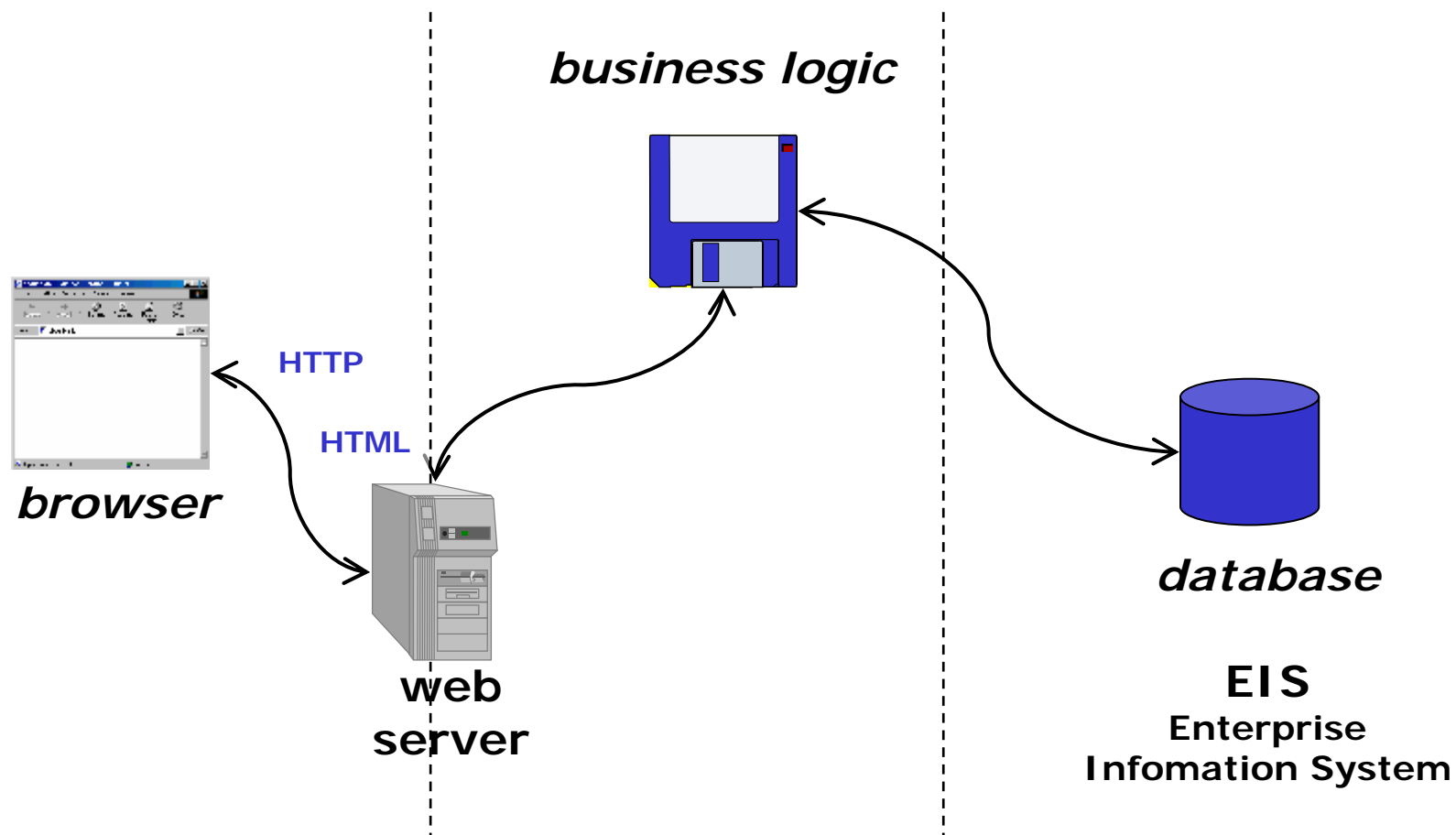
Dati dell'applicazione

- Database
- Sistemi di altro tipo
 - Mainframe
 - Sistemi legacy
 - File XML





Comunicazioni tra i livelli logici



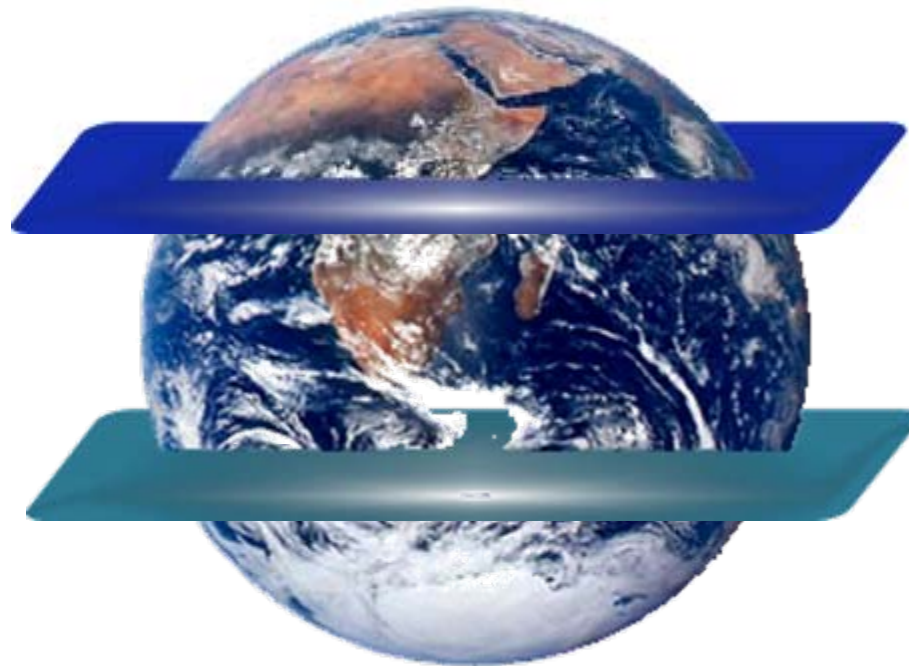


Applicazione come collezione di componenti

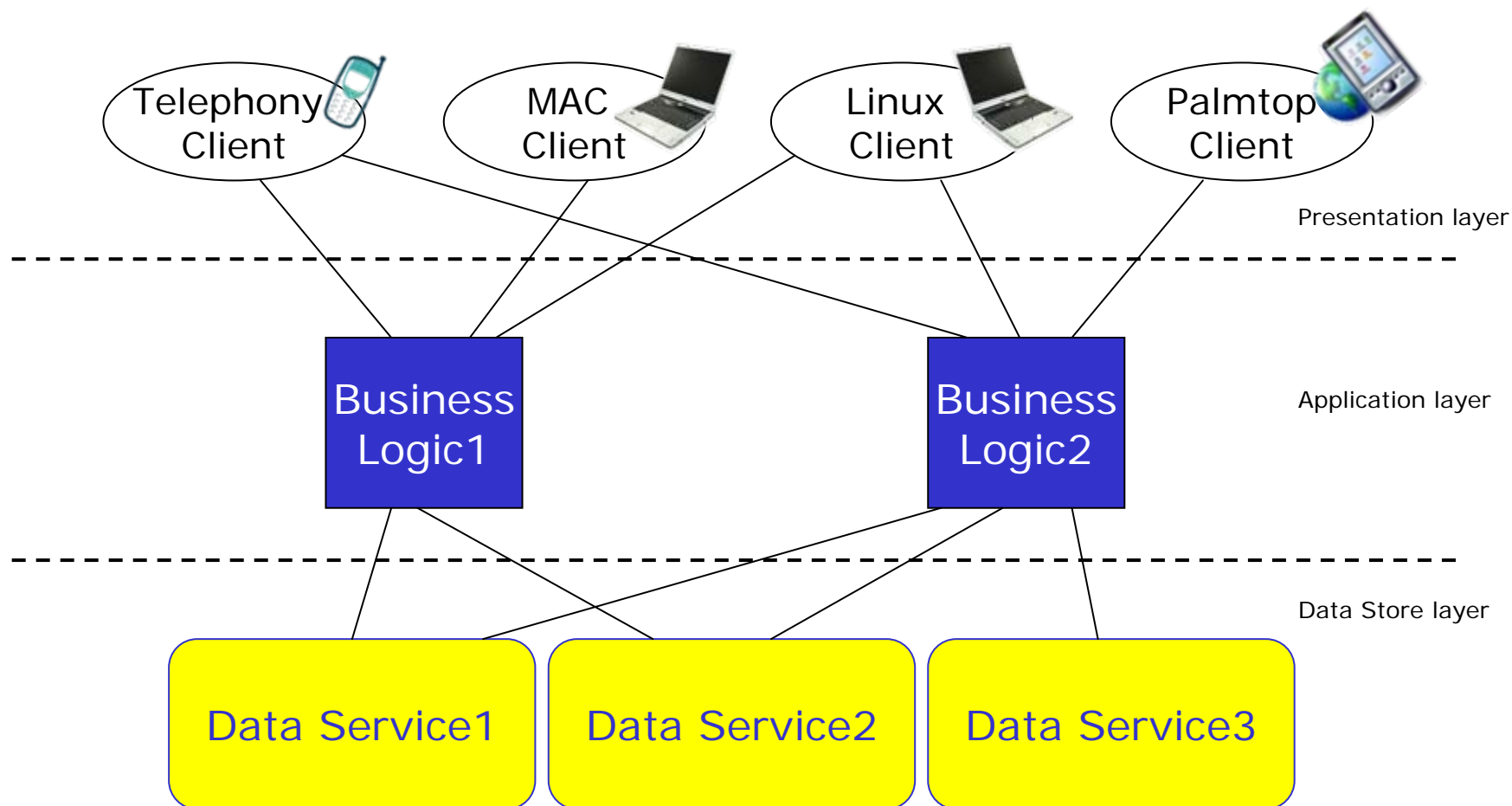
Non c'è comunicazione diretta tra *presentation* e *data* layer

Applicazioni viste come collezione di componenti cooperanti.

Ogni componente può essere contemporaneamente parte di applicazioni diverse



Componenti appartenenti a più applicazioni



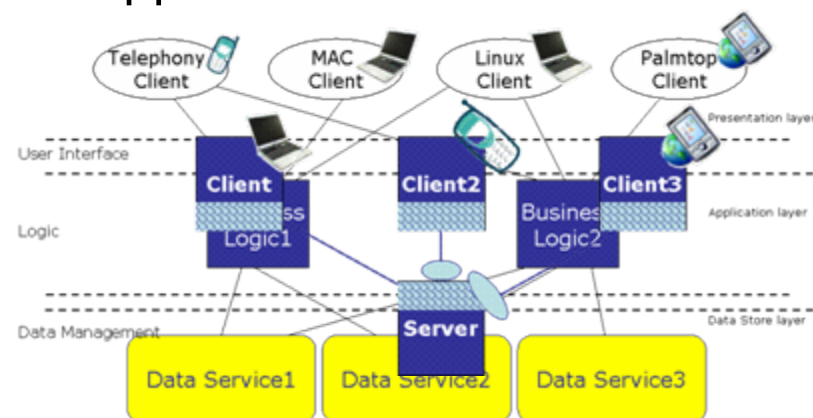
3-tier: vantaggi

Flessibilità e modificabilità

- Componenti usabili in sistemi diversi
- Modifiche indipendenti
- Estensioni dell'applicazione facilitate
- Ricerca di bug agevolata

Interconnettività

- Non sono necessari gli *adapter*
- Accesso a dati comuni da parte di applicazioni diverse facilitato





Web Application 3-tier: vantaggi

Gestione di sistemi distribuiti

- Business logic aggiornabile senza interventi sui client

Preleva direttamente i dati aggiornati (*hot data*) operando on-line sul database

- Consente di realizzare pagine HTML dinamiche
- Elimina la necessità di aggiornare le pagine HTML (statiche)

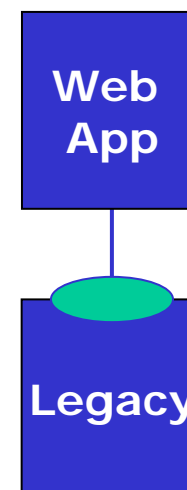
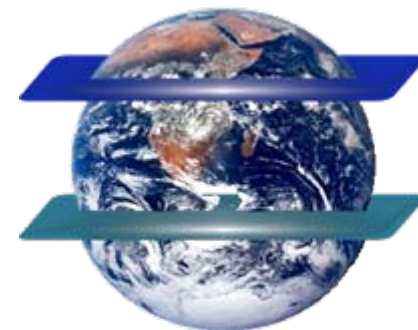
3-tier: svantaggi

Dimensione ed efficienza

- Comunicazione tra componenti onerosa
- Uso di apposite librerie software

Software Legacy

- Difficile applicare il modello 3-tier
- Uso di *adapter*



Web Application 3-tier: svantaggi

Web server e *Db server* sono spesso i colli di bottiglia del sistema
Elevati problemi di *sicurezza*

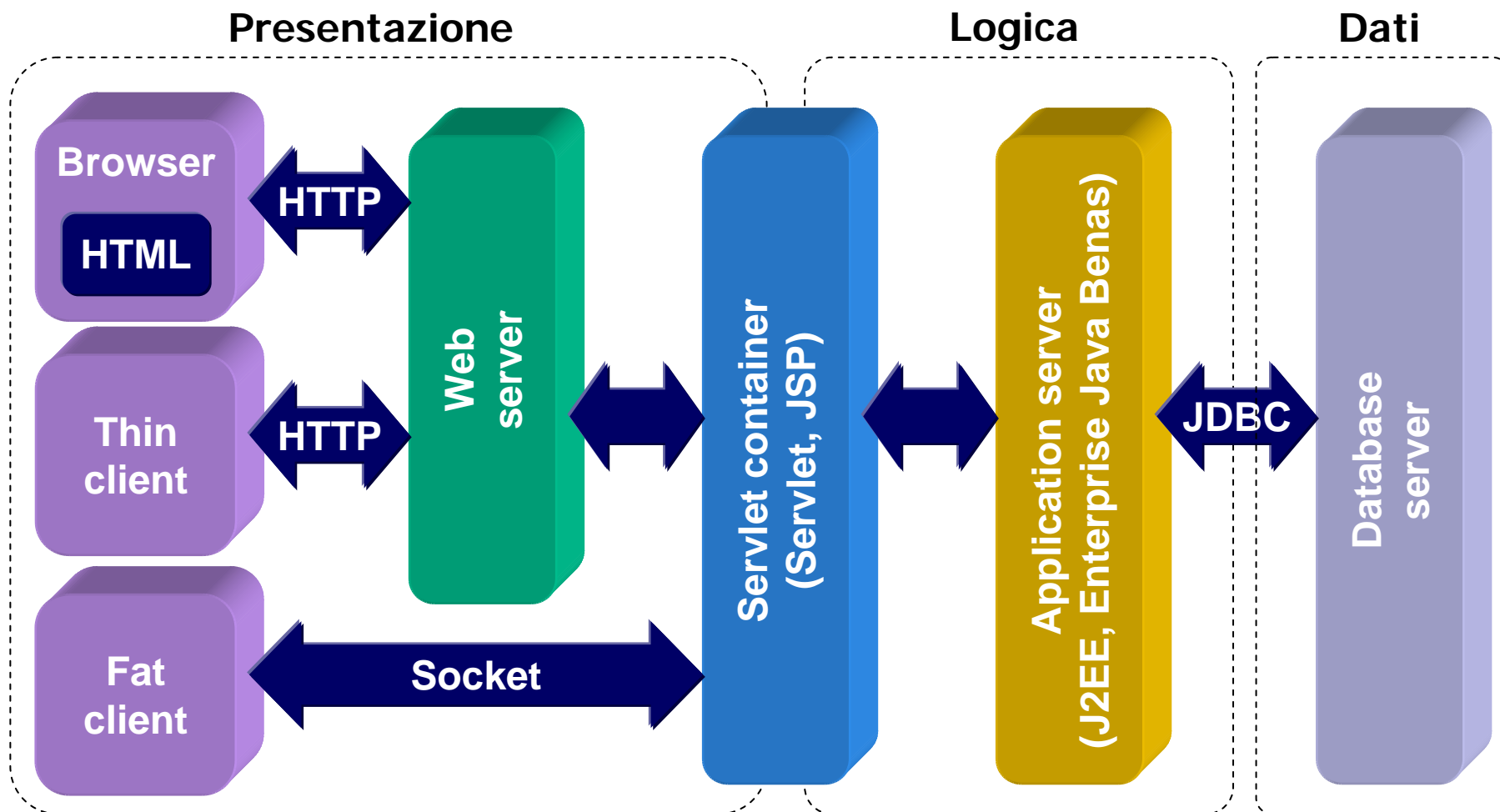
- Virus
- Worm
- Trojan horses

(Rapida evoluzione delle tecnologie)





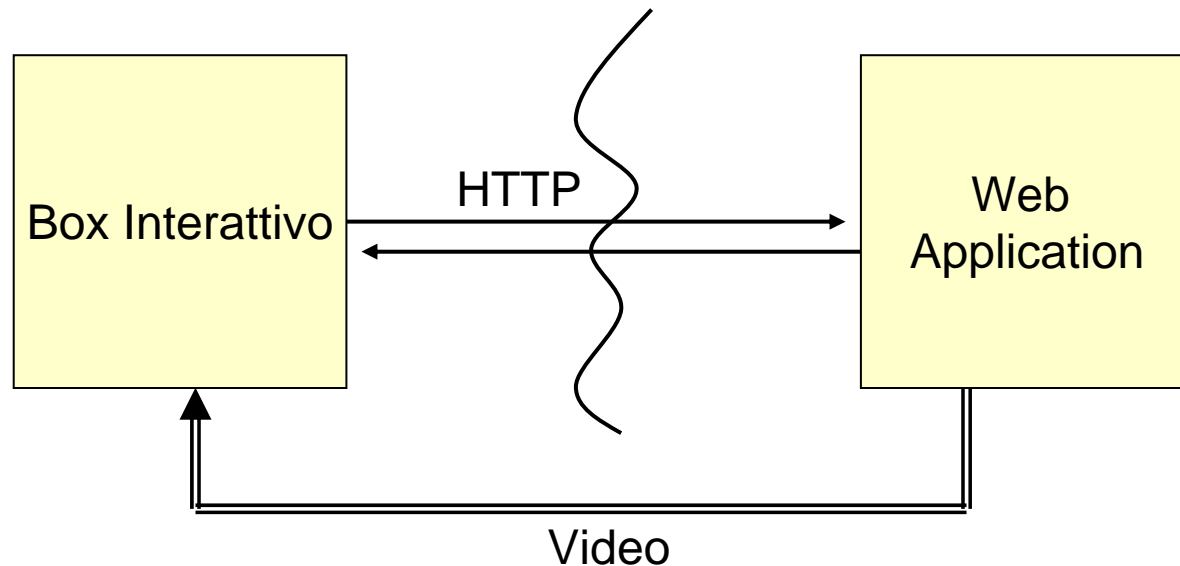
Livelli applicativi di una Web Application





Web Application per il Video On Demand (VOD)

Il Box è un PC ed opera come un browser (pur non avendo la classica interfaccia dei browser), scambiando messaggi HTTP con una WebApp. Lo streaming del video richiesto avviene poi con un altro protocollo





Modello Logico

