



# Web Services

## Structuring Concepts





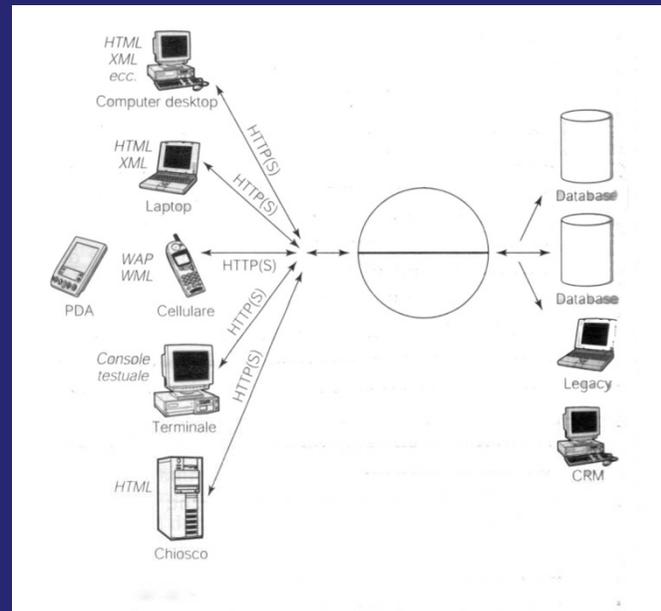
# Sommario

1. Web service e loro caratteristiche
2. Architetture BCE per un Web Service
  - I. Definizione
  - II. Architettura
  - III. Descrizione componenti
3. Esempi





# Web service. Caratteristiche



Trattasi di un sistema software progettato per supportare l'interoperabilità tra diversi elaboratori su di una medesima rete

[W3C: World Wide Web Consortium]





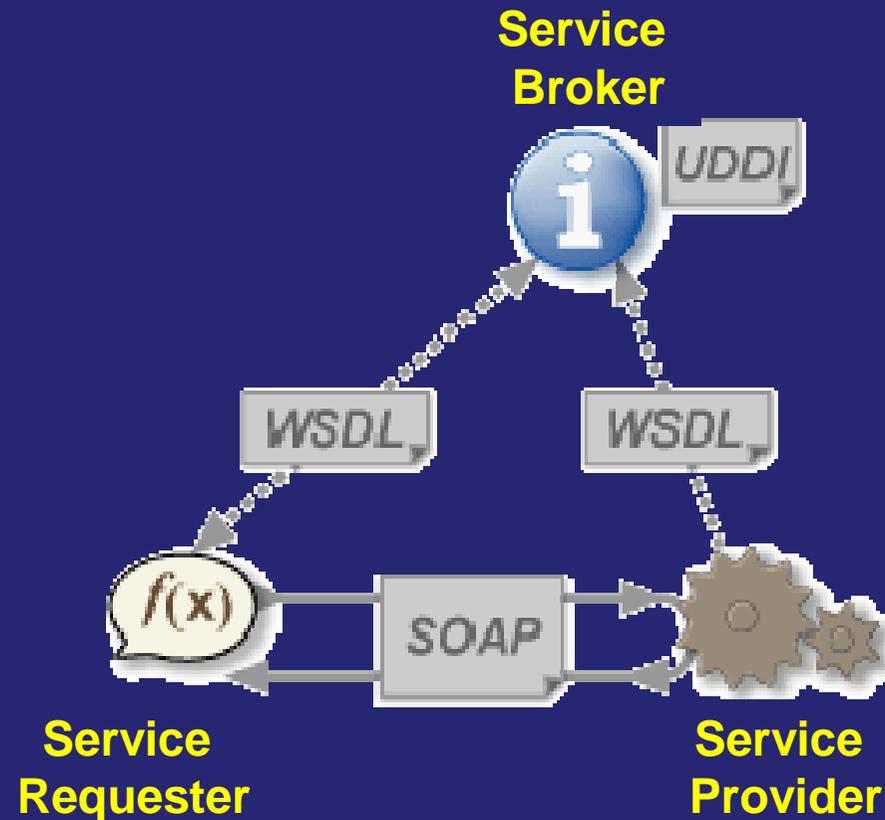
# Web service. Caratteristiche

- Permettono l'interoperabilità tra diverse applicazioni software su diverse piattaforme hardware.
- I protocolli e gli standard utilizzati sono 'open'.
- Utilizzando HTTP come livello (*layer*) di trasporto, i WS sono *firewall friendly*.
- Diversi WS possono essere facilmente integrati per implementare applicazioni complesse.





# Web service. Come funzionano?



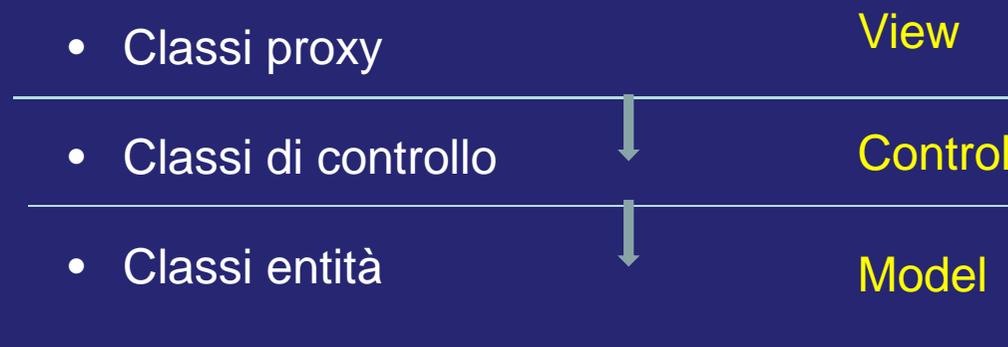


# Web service – MVC

## Definizione

Trattasi di sistemi software progettati per fornire funzionalità accessibili via web. Espongono un'interfaccia software tramite la quale altri sistemi possono usufruire dei servizi messi a disposizione.

Un web service basato sul pattern MVC è tipicamente composta da quattro tipologie di componenti:

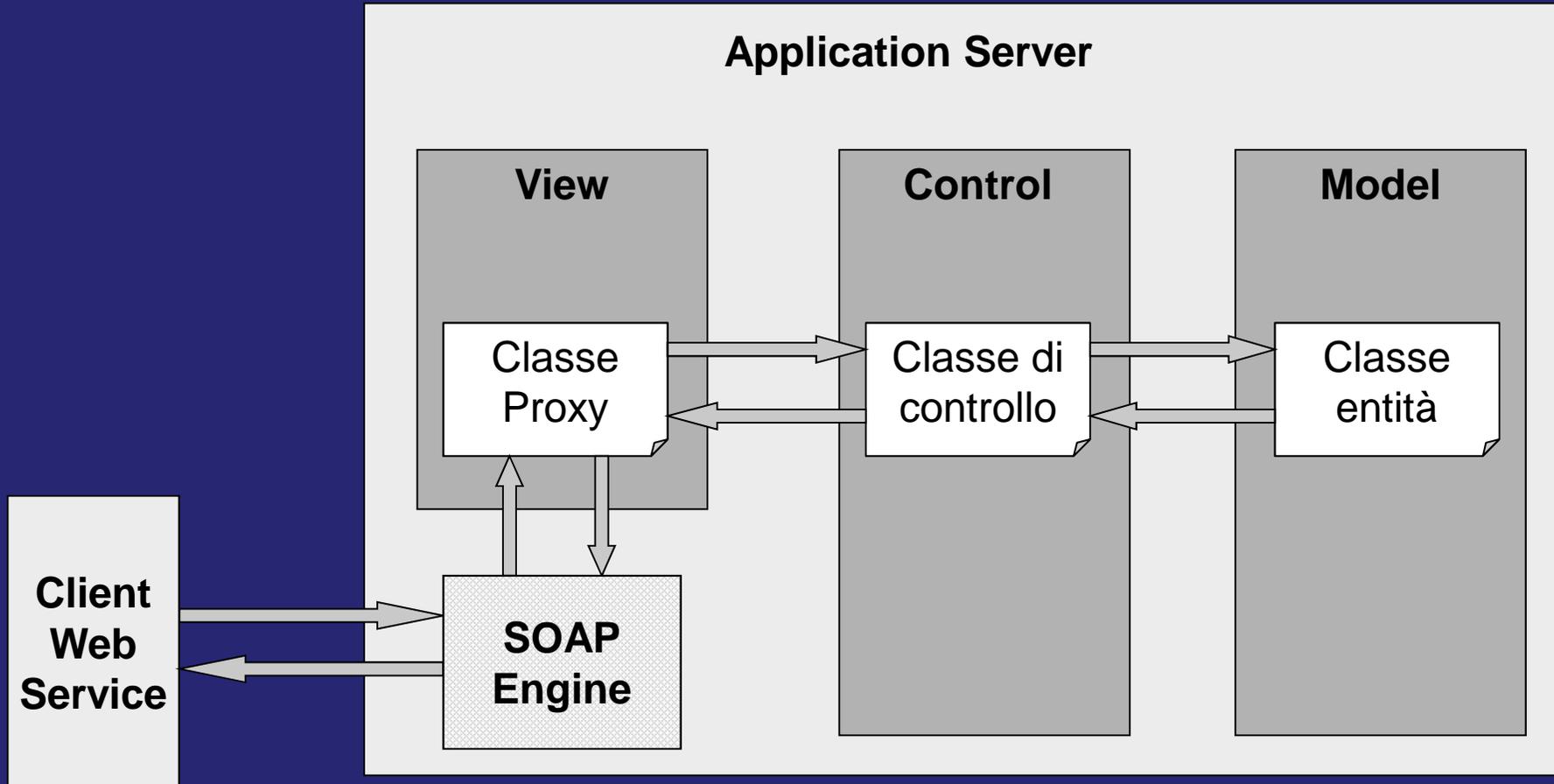


Usualmente l'interfaccia software è descritta in WSDL e il modello di comunicazione è quello di RPC.





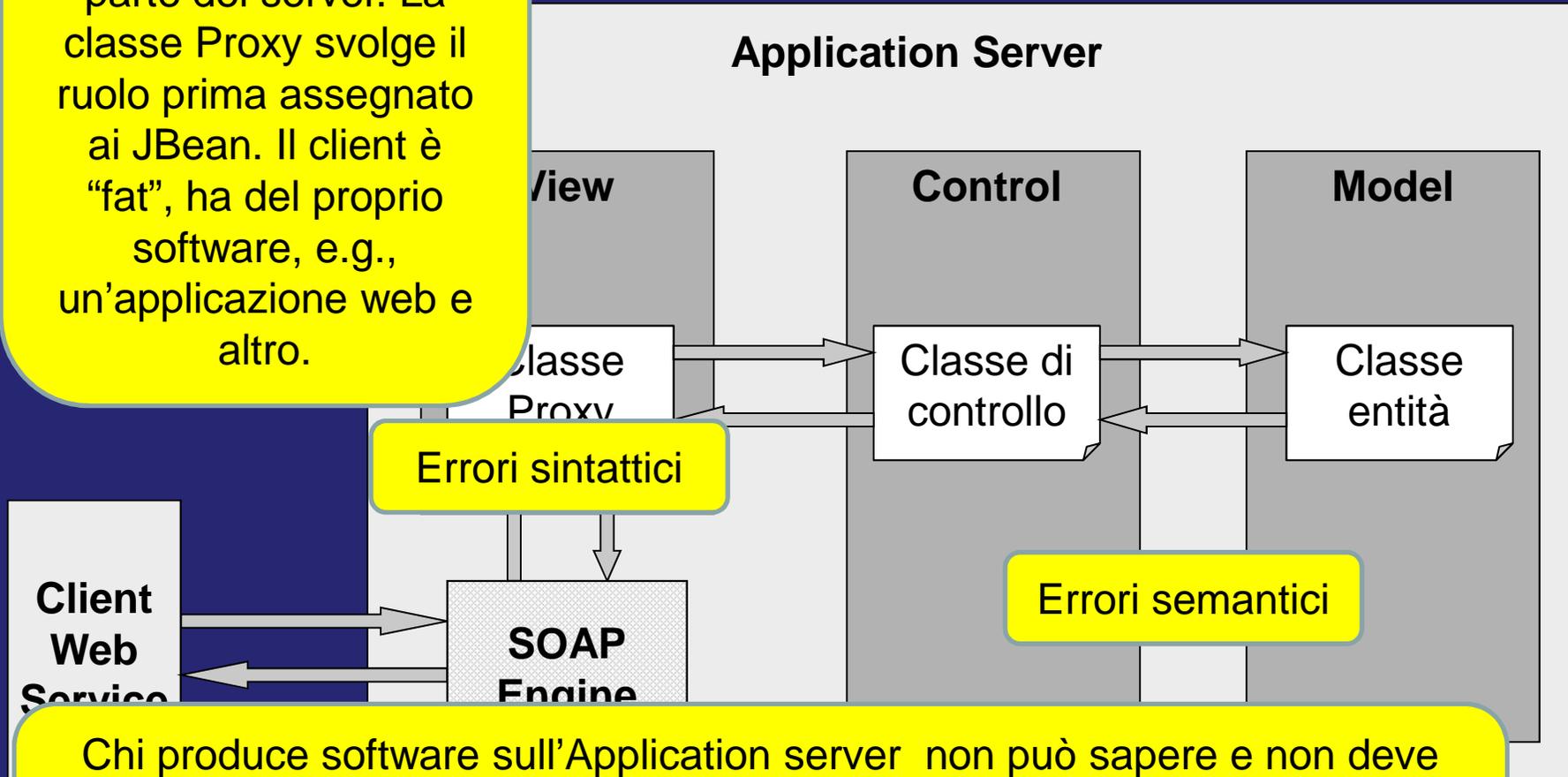
# Web service – MVC. **Architettura**





## Service – MVC. Architettura

Non c'è più JSP. C'è un nuovo componente (SOAPE) il quale è parte del server. La classe Proxy svolge il ruolo prima assegnato ai JBean. Il client è "fat", ha del proprio software, e.g., un'applicazione web e altro.



Chi produce software sull'Application server non può sapere e non deve fidarsi della parte client: peraltro, chi produce un'applicazione client potrebbe non validare i propri dati; pertanto, il servizio tipicamente li **valida in proprio** sia sintatticamente che semanticamente.



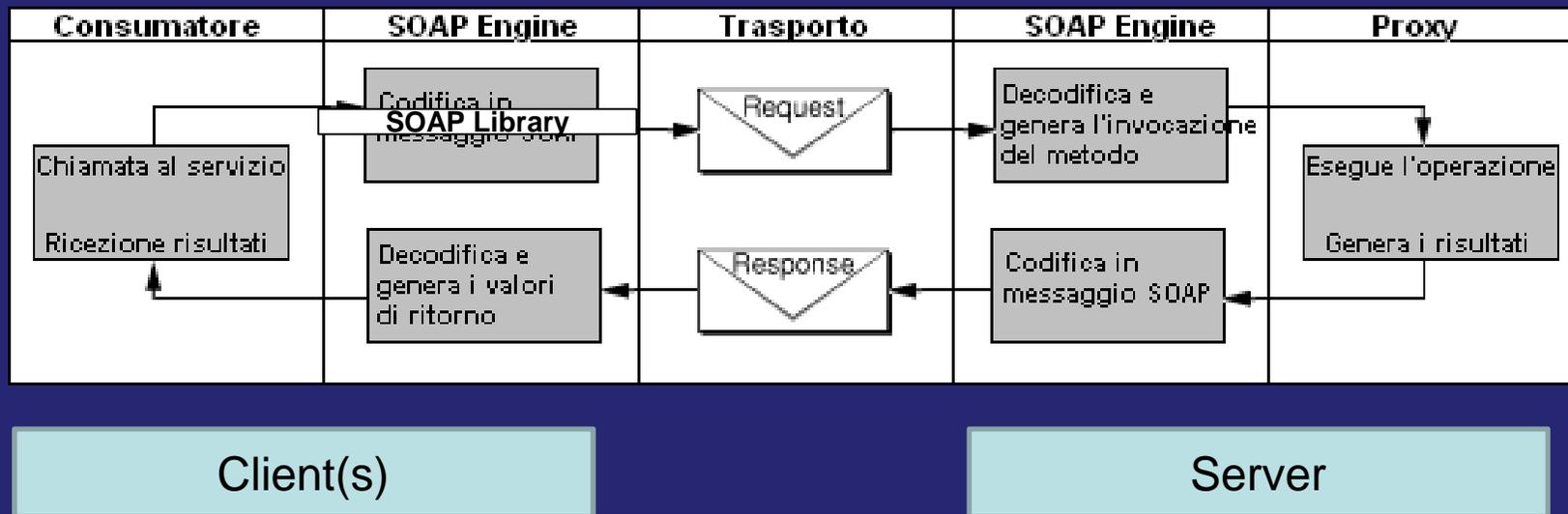
# Web service – MVC. **Descrizione dei Componenti**

## Strato View – classi proxy

- Sono le classi i cui metodi vengono esposti come servizi.
- Componenti atti a collegare il SOAP Engine con lo strato di Control.

## SOAP Engine

- Strumento che permette di generare ed utilizzare Remote Procedure Call appoggiandosi al protocollo SOAP.





## Web Application & Web Service in MVC/VPM

- Come è possibile notare, l'architettura di una Web Application è molto simile a quella di un Web Service. Difatti, le due strutture risultano essere identiche con la sola *differenza dello strato View* (controllori ed entità non cambiano).
- Il pattern gioca, in questo caso, un *ruolo fondamentale* nel disaccoppiare gli strati, in particolare, rende gli strati Control e Model completamente indipendenti da quello View (VPM).
- Di conseguenza, un'applicazione realizzata seguendo il pattern BCE risulta essere ben disposta a subire una *modifica di tipo strutturale* (e.g. migrazione da un'architettura software a un'altra).





# Web service. Come funzionano?

- L'accesso ad un Web service viene effettuato tramite *Remote Procedure Call* (RPC)
- Una RPC si codifica riportando in *formato XML* la firma di una funzionalità del WS: il **nome** della funzionalità, i **parametri** d'ingresso, l'eventuale **tipo restituito**. L'esecuzione della RPC restituirà il **valore** del parametro di uscita in formato XML.
- Se i parametri di ingresso o quello restituito sono tipi di dati complessi, devono essere serializzati anch'essi (riportati in forma testuale secondo uno standard XML).
- Sia il WS che il client che lo utilizza necessitano di procedure di *serializzazione* e *deserializzazione*.





## Web service. Pubblicazione di un servizio: WSDL

La pubblicazione di un servizio sul web può descriversi nel linguaggio **WSDL** (**Web Service Description Language**).

- Per permettere ad un qualunque sviluppatore di realizzare un client ad hoc, il quale sia in grado di interfacciarsi con un servizio, insieme al WS viene anche pubblicata sul web una *descrizione del WS* medesimo.
- Tale descrizione viene riportata in un linguaggio standard detto WSDL.
- Nella descrizione sono elencati il **nome del servizio**, i **tipi dei parametri**, il **tipo eventualmente restituito** e *come serializzare i parametri di tipo complesso*.





# Web service. UDDI

*Localizzazione di un servizio sul web: protocollo **UDDI (Universal Description Discovery and Integration)**.*

- Trattasi di una particolare **base di dati** (ordinata ed indicizzata) detta **registry**, basata su **XML** e **indipendente dalla piattaforma** hardware.
- Permette alle aziende la **pubblicazione** dei propri dati e dei servizi offerti su internet, permette, quindi, la **scoperta** e l'**interrogazione** dei servizi offerti sul web, delle aziende che li offrono e della maniera per usufruirne.
- La **centralizzazione** della descrizione e della localizzazione dei WS in un registro comune permette la ricerca ed il **reperimento veloce** dei WS disponibili in rete.





# Esempi

1. [Esempio\\_01](#)
2. [Esempio\\_02](#)

