

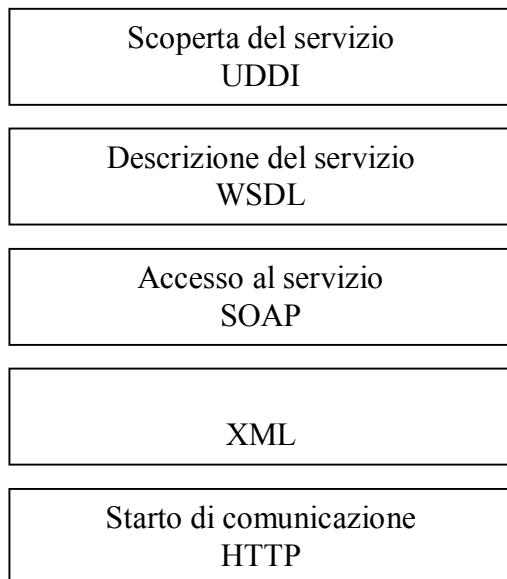
# Web Services

I web services servono a rendere interoperabili le applicazioni e favoriscono la loro integrazione. I servizi web sono applicazioni software che possono essere scoperte, descritte e usate utilizzando XML e protocolli WEB. Possono quindi essere usati sia su intranet che su Internet.

Si parla di servizi web in quanto si tratta di applicazioni che svolgono funzioni specifiche disponibili sul web.

Il linguaggio UDDI (Universal Description Discovery and Integration) viene usato per scoprire i servizi. Il linguaggio WSDL (Web Service Description Language) descrive il servizio e come dobbiamo usarlo. Il linguaggio SOAP (Simple Object Access Protocol) consente l'accesso al servizio.

Questi tre linguaggi si basano su XML. L'accesso al servizio avviene tramite il protocollo HTTP (Hypertext Transport Protocol), lo stesso protocollo che fa comunicare un browser ed un server web.

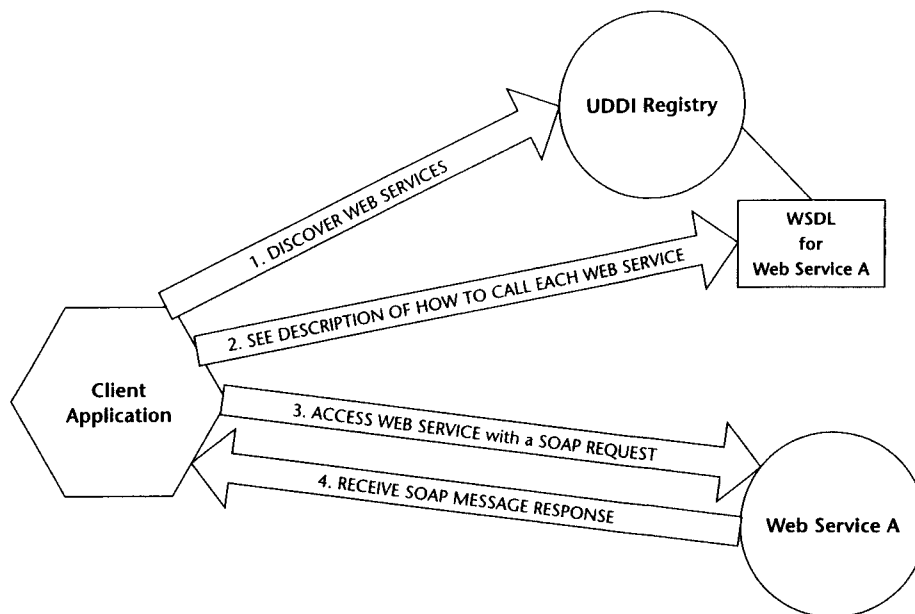


I livelli di base dei Servizi Web

## Utilizzo Dei Web Service

Uno scenario per l'utilizzo dei Web Service è costituito dalle seguenti fasi:

- 1) Il programma "client" (quello dell'utente) cerca di scoprire nel registro UDDI quale è il servizio che interessa.
- 2) Scoperto il servizio si usa la descrizione del servizio (WSDL) per vedere come usarlo (cioè come chiamarlo, che dati gli vanno passati e che dati il servizio restituisce).
- 3) Il client accede al Servizio Web passando i dati con SOAP.
- 4) Il servizio web elabora la richiesta e manda il risultato al client sempre usando SOAP



A common scenario of Web services in use.

Bisogna notare che il servizio web non si occupa dell'interfaccia utente, che sarà controllata dal client, ma implementa la "business logic" cioè la logica dell'applicazione.

Sistemi eterogenei possono comunicare perchè tutti possono capire XML e gli starti superiori (SOAP, WSDL, UDDI) basati su XML.

In più in servizio web può a sua volta utilizzare altri servizi web per completare il lavoro.

Vediamo più in dettaglio alcuni aspetti.

## SOAP

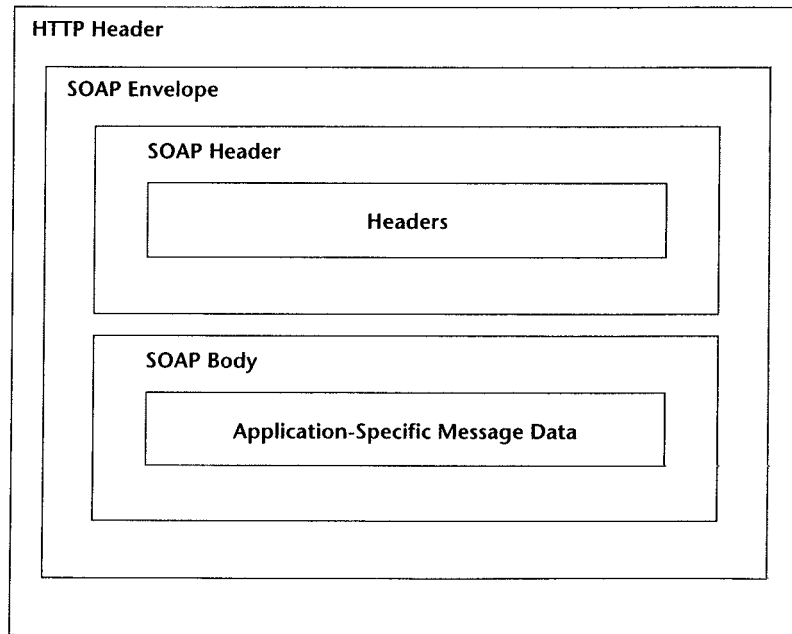
SOAP è la sintassi con cui si scambiano messaggi XML con il Servizio Web. SOAP è un contenitore che ha al suo interno messaggi XML che vengono scambiati con il protocollo HTTP. Il programma client invia una SOAP Request al servizio web ed il servizio replica con una SOAP Response.

Il messaggio SOAP ha tre parti:

Una busta (envelope) SOAP che contiene il messaggio

Una intestazione (header) che descrive come sono codificati i dati

Un corpo (body) SOAP che contiene il messaggio vero e proprio.



Structure of a Web-based SOAP message.

Un esempio di **SOAP Request** potrebbe essere quello che segue. Nel corpo si chiede il “Last Trade Price” (elemento `<GetLastTradePrice>` nel senso definito dal namespace “Some-URI”) del simbolo DIS.

```

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-
ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>DIS</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
  
```

La risposta è contenuta nel SOAP Response che segue, dove nell’elemento `<GetLastTradePriceResponse>` troviamo `<Price>34.5</Price>`

```

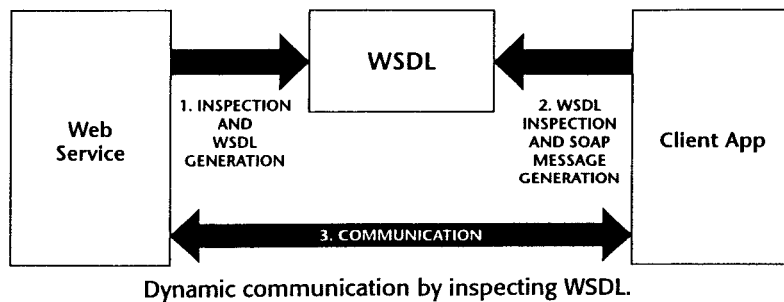
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-
ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePriceResponse xmlns:m="Some-URI">
      <Price>34.5</Price>
    </m:GetLastTradePriceResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
  
```

## Descrivere i Servizi Web - WSDL

WSDL descrive dove si trova il servizio, cosa fa, come va chiamato. Le descrizioni WSDL vengono create con dei toolkit da chi sviluppa i servizi. Vengono utilizzati dal client per generare (sempre in maniera automatica) i messaggi SOAP.

Ci sono 3 passi.

- 1) lo sviluppatore del servizio produce con l'ispezione del servizio stesso la descrizione WSDL
- 2) il client usa questa descrizione per generare (automaticamente) i messaggi SOAP
- 3) la comunicazione fra client e web service può avere luogo.



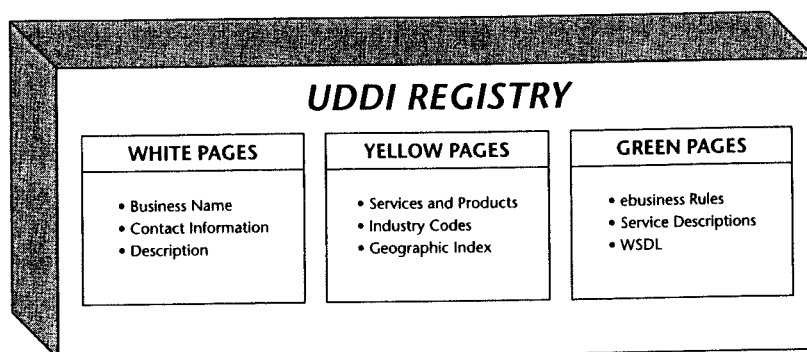
## Scoprire i Servizi Web – UDDI

L'UDDI (Universal Description Discovery and Integration) può essere vista come una guida del telefono per i Web Services. Le organizzazioni possono rendere pubblici, in questi registri, i servizi che esse rendono disponibili. Le applicazioni possono automaticamente esaminare questi elenchi e trovare quei servizi che servono.

Possiamo avere tre tipi di informazioni.

- Pagine bianche con informazioni su come contattare una organizzazione.
- Pagine gialle dove le organizzazioni sono elencate per categorie usando una tassonomia.
- Pagine verdi che elencano i Web Services offerti dalle varie organizzazioni con i loro WSDL.

I registri UDDI possono essere pubblici ed essere accessibili da Internet oppure possono servire una singola organizzazione ed essere accessibili da una intranet.



A UDDI Registry as a conceptual phone book.

## Integrare i Servizi

Per realizzare applicazioni complesse si possono integrare servizi web diversi.

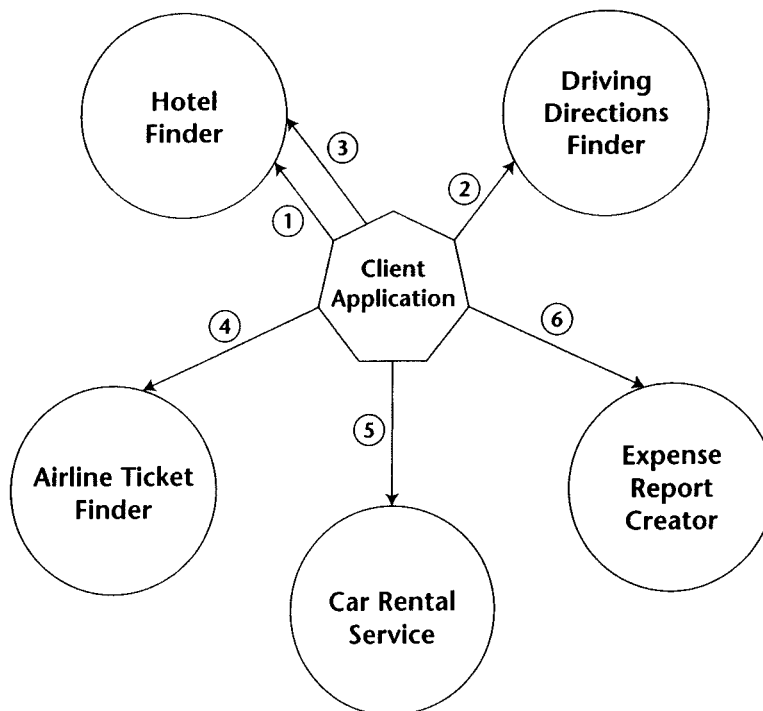
Possiamo pensare di organizzare un viaggio. Potremmo disporre dei seguenti servizi:

- Hotel Finder
- Driver Direction Finder
- Airline Ticket Finder
- Car Rental Service
- Expense Report Creator

Ogni servizio può avere una sua logica interna. Prenotare l'albergo più conveniente, in una certa zona etc., affittare l'auto richiesta al prezzo minore etc. In alcuni casi sarà l'utente a dover prendere delle decisioni.

Potrebbe succedere qualcosa del genere:

- 1) L' Hotel Finder seleziona un certo numero di hotel nella zona e con le caratteristiche volute.
- 2) Si ricorre al Driving Direction Finder per vedere quale è l'hotel raggiungibile più facilmente
- 3) Scelto l'hotel più comodo da raggiungere si ricorre ancora all' Hotel Finder per fare la prenotazione.
- 4) Si prenota l'aereo usando l' Airline Ticket Finder
- 5) Si prenota una auto a noleggio.
- 6) Infine l'Expense Report Creator può essere usato per rendicontare le spese.



An orchestration example.

## Applicazioni all'E-Learning

Nel campo dell'e-learning potremmo pensare a Servizi Web che assolvano a diverse funzioni.

- 1) Autenticazione Lo studente ricorre a questo servizio per autenticarsi quando entra nel sistema
- 2) Gestione Carriera Studenti. Si identificano i dati dello studente, i corsi a cui è iscritto etc.
- 3) Iscrizione ai Corsi
- 4) Iscrizione agli Esami
- 5) LMS per l'erogazione di materiali didattici e per il tracciamento dell'attività dello studente.
- 6) Sistemi di valutazione e autovalutazione.
- 7) Streaming di materiali multimediali.
- 8) Sistemi di comunicazione asincrona (forum, chat, etc.).
- 9) Sistemi di comunicazione sincrona (Videoconferenza, strumenti di collaborazione).

## GRID Computing

Un GRID è una collezione di sistemi di calcolo distribuiti che possono lavorare insieme. Il GRID può essere organizzato per offrire Servizi Web con caratteristiche aggiuntive come bilanciamento del calcolo, fault-tolerance e sono adatti a fornire servizi su larga scala.

Un progetto che permette risorse di GRID computing in forma di Web Services è GLOBUS (<http://www.globus.org>).

## AGENTI

Gli agenti software servono a realizzare sistemi distribuiti. Gli agenti possono cooperare per eseguire dei compiti. Per cooperare gli agenti hanno bisogno di comunicare. A questo fine gli agenti si ispirano agli "speech acts" (atti linguistici) di Searle. Le ontologie vengono utilizzate dagli agenti per dare una semantica agli atti comunicativi.

## Rapporto fra Semantic Web e Web Services

Per automatizzare la scoperta e l'uso di servizi web è necessario aggiungere delle informazioni semantiche. Si stanno sviluppando ontologie che descrivono le proprietà e le funzionalità dei servizi web. Ad esempio uno dei linguaggi per rappresentare ontologie OWL (Ontology Web Language) proposto dal w3c ha una specializzazione chiamata OWLS dedicata alla descrizione dei servizi web.