



TePSIT



Web Services

Giulio Angiani
I.F.O.A. - Reggio Emilia



Web Services

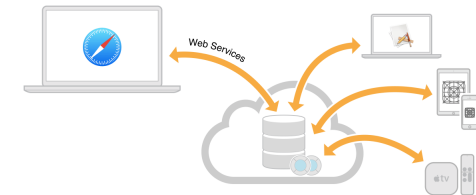


Cos'è un Web Service ?

Secondo la definizione data dal **World Wide Web Consortium (W3C)** un Web service (servizio web) è un **sistema software progettato per supportare l'interoperabilità tra diversi elaboratori su una medesima rete oppure in un contesto distribuito** [fonte Wikipedia]

Lo possiamo considerare come un componente SOFTWARE di rete che

- contattabile con vari protocolli e interfacce
- permette al client di utilizzare le funzioni che espone
- permette l'accesso controllato a risorse remote
- permette l'interoperabilità fra linguaggi diversi e piattaforme diverse



Web Services : Protocolli standard

- Molti protocolli standard per i WS
 - HTTP/HTTPS
 - XML
 - WSDL
 - SOAP
 - REST
 - JSON
 - CSV/TXT
- I servizi possono essere implementati anche con protocolli proprietari non standard



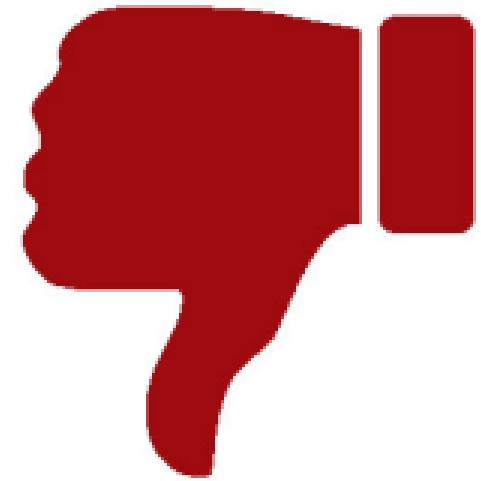
Web Services : Vantaggi dei WS

- Permettono l'**interoperabilità** tra diverse applicazioni software e su diverse piattaforme hardware/software
- Utilizzano un formato dei dati di tipo **testuale**, quindi più comprensibile e più facile da utilizzare per gli sviluppatori (esclusi ovviamente i trasferimenti di dati di tipo binario)
- Normalmente, essendo basati sul **protocollo HTTP**, non richiedono modifiche alle regole di sicurezza utilizzate come filtro dai firewall
- Sono semplici da utilizzare e possono essere **combinati** l'uno con l'altro (indipendentemente da chi li fornisce e da dove vengono resi disponibili) per formare servizi "integrati" e complessi
- Permettono di **riutilizzare applicazioni** già sviluppate.
- Fintanto che l'interfaccia rimane costante, le **modifiche** effettuate ai servizi rimangono trasparenti
- I servizi web sono in grado di **pubblicare le loro funzioni** e di scambiare dati con il resto del mondo
- Tutte le informazioni vengono scambiate attraverso **protocolli "aperti"**



Web Services : Svantaggi

- Performance **più basse** rispetto ad altre modalità di comunicazione
- Basato su **HTTP**
- **No sistemi distribuiti**



Web Services : Quando usarli?

- Per richiedere dati e/o servizi presenti su risorse remote e non accessibili direttamente
- Per fornire servizi basandosi su ws esistenti
- Per rendere omogenei dati presenti su sistemi diversi
- Per rendere disponibili a utenti e a software dati aperti (OPEN DATA) secondo formati standard



Web Services : Come usarli?

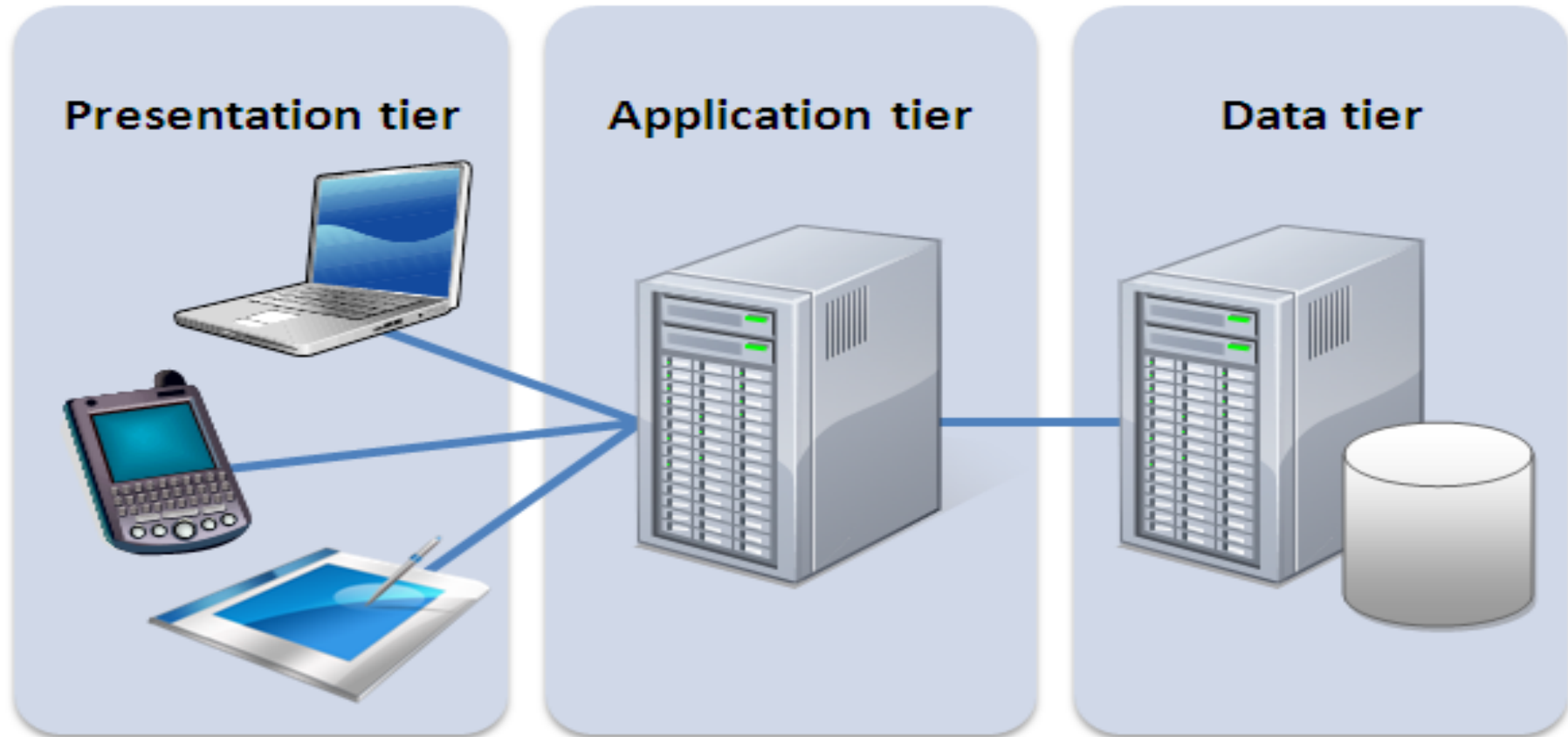
DEVO SEMPRE FORNIRE UNA DOCUMENTAZIONE DEL SERVIZIO

- Per i servizi standardizzati è standardizzato anche il modo di fornire la documentazione
 - SOAP (Autodocumentato)
 - REST (Formato standard ma da concordare con il consumatore del servizio)
- Per i servizi non standardizzati va costruita la documentazione ad hoc per l'oggetto **consumer** ovvero l'oggetto software che deve utilizzare il prodotto del WS



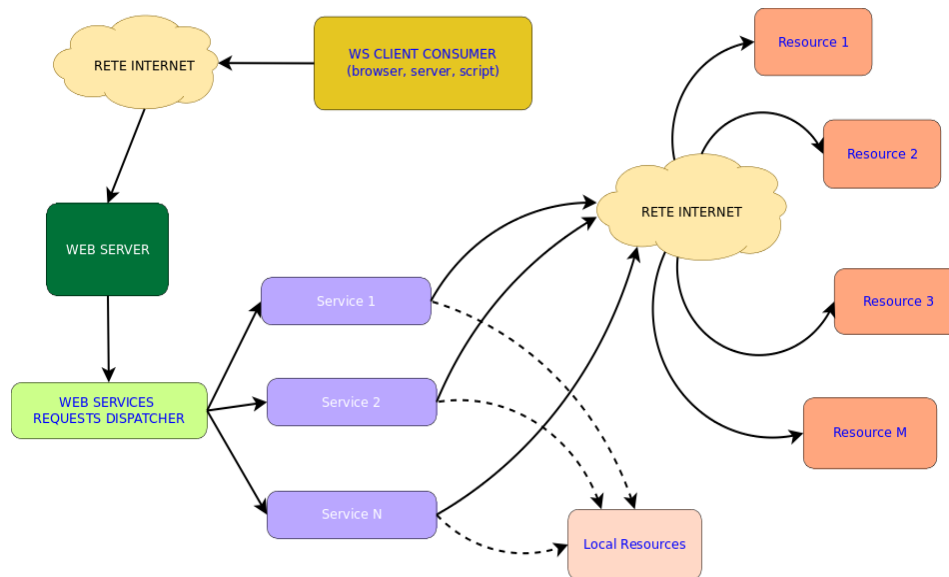
Web Services: Architettura

- Abbiamo già incontrato le architetture web three-tiered (a tre livelli)



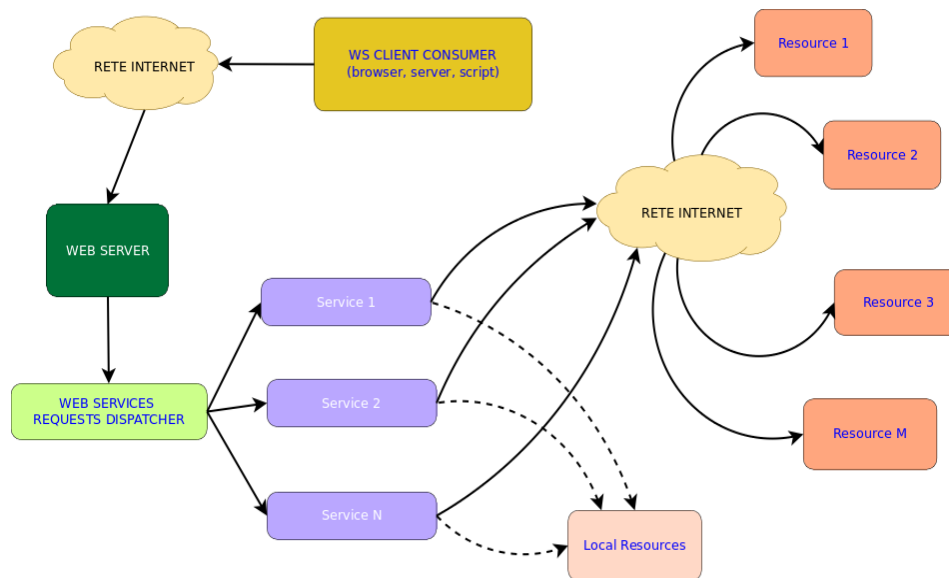
Architetture

- per i WS l'architettura può essere anche a più livelli perché la sorgente dati utilizzata dal servizio esposto non è detto (ed in generale non lo è) che sia residente sulla stessa macchina del servizio stesso.
- varie opzioni
 - dati presenti localmente (3-tiers architecture)
 - dati forniti da altri servizi (multitier architecture)



Architetture

- E' spesso presente un **dispatcher** che interpreta la richiesta e la indirizza al servizio giusto
- questo servizio può essere locale, esterno o una **composizione** di più servizi anche di tipologie diverse



Esempio: WS con architettura NON standard

- Cominciamo con un esempio semplice
- Vogliamo esporre un servizio che resituisce l'ora esatta
- Dobbiamo:
 - comunicare l'URI del servizio
 - esplicitare il formato col quale verranno esportati i dati
 - il protocollo utilizzato
 - eventuali parametri in input da fornire
- Ad esempio:
 - URI: **http://94.177.213.7/ws/oraesatta**
 - Output: **formato JSON** es: { "hour": '14', "minutes": '0', "seconds": '15' }
 - Protocollo: **HTTP**
 - Parametri: **nessuno**



Esempio: WS con architettura NON standard

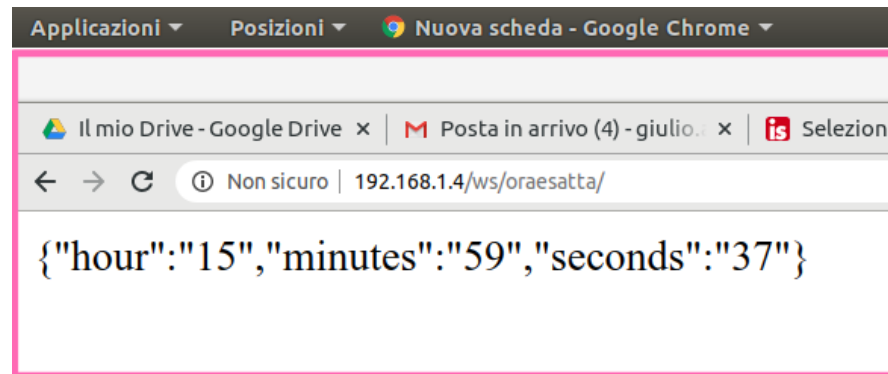
- Possiamo pensare di interrogare il nostro servizio con una chiamata via web
- oppure a riga di comando con una istruzione in un linguaggio qualsiasi

```
import urllib.request
import json
uri = "http://94.177.213.7/ws/oraesatta"
t = json.loads(urllib.request.urlopen(uri).read())
print(t)
```

PYTHON

```
{'hour': '15', 'minutes': '59', 'seconds': '37'}
```

OUTPUT



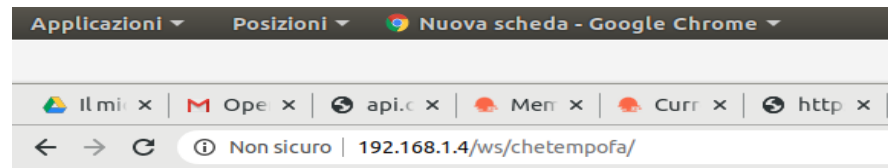
Domande

- Con quale linguaggio è stato implementato il WS ?
 - Il servizio genera il risultato o lo prende da altre fonti?
 - E' presente un database? E' locale?
-
- Risposta: NON E' POSSIBILE SAPERLO



Un altro esempio: Che tempo fa ?

- Forniamo un servizio che indica il tempo meteorologico di Roma
- Documentazione del servizio:
 - URI: **http://94.177.213.7/ws/chetempofa**
 - Output: **messaggio formattato** es:
"Hi!, sono le <ORACORRENTE>
In Rome there is the sun"
 - Protocollo: **HTTP**
 - Parametri: **nessuno**



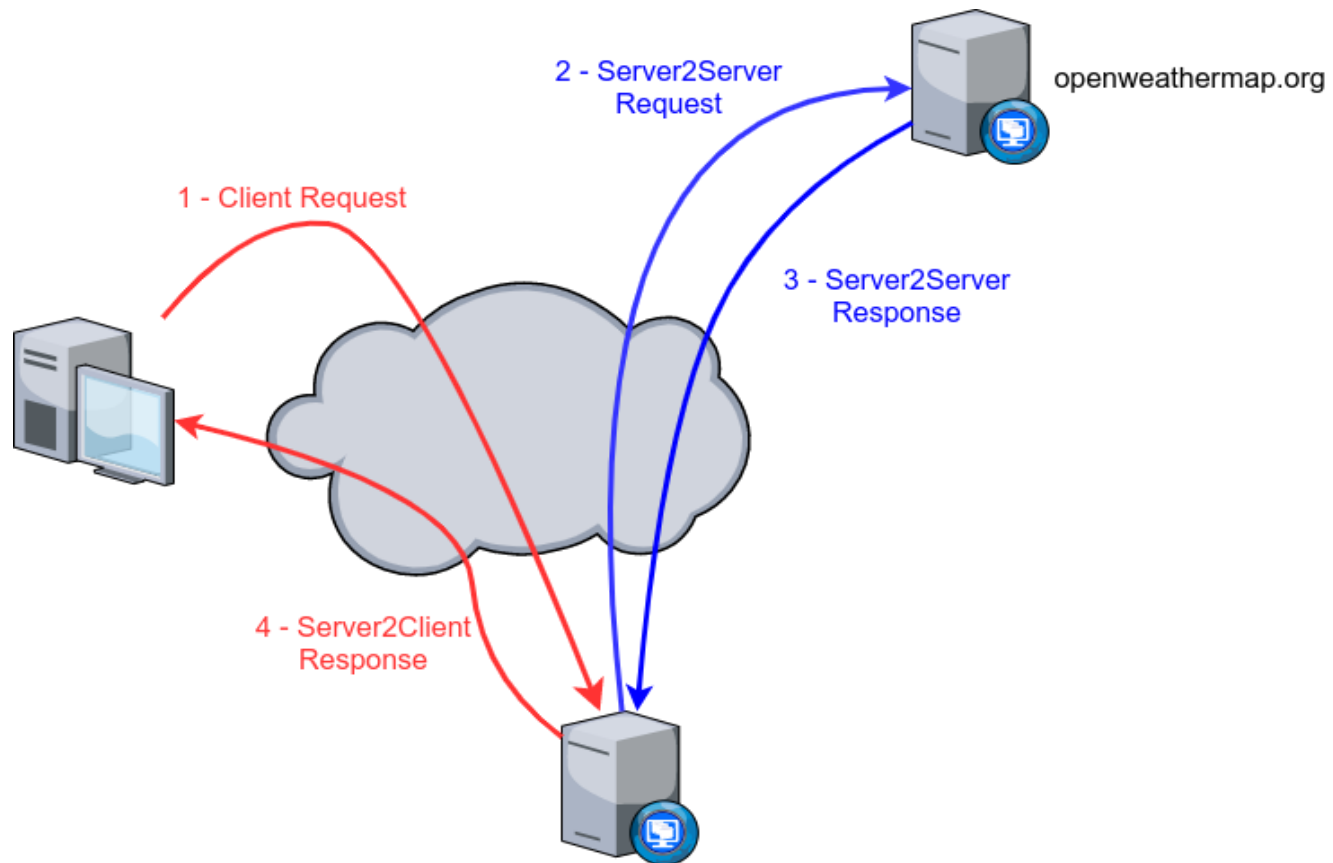
Hi! It's 16:48:51

Now in Rome, there are scattered clouds



Che architettura?

- i dati relativi alla situazione del tempo NON risiedono sul server contattato dal client



Realizzazione

PHP

```
$uri = "http://api.openweathermap.org/data/2.5/weather?q=Roma,it&appid=1234567890";  
$content = file_get_contents($uri);  
$info = json_decode($content);  
$t = strftime("%H:%M:%S");  
$result = "Hi! It's {$t}<br><br>Now in {$info->name}, there are {$info->weather[0]->description}";  
exit($result);
```

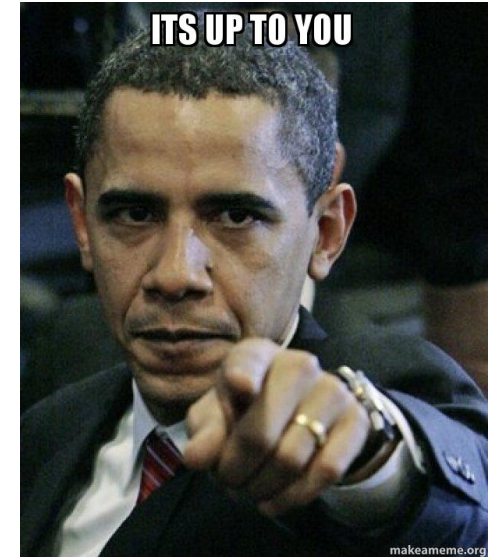
- **openweathermap.org** restituisce un oggetto tipo JSON così fatto

```
{ "coord": { "lon": 12.48, "lat": 41.89 }, "weather": [ { "id": 802, "main": "Clouds", "description": "scattered  
clouds", "icon": "03d" } ], "base": "stations", "main": { "temp": 292.63, "feels_like": 291.48, "temp_min": 291.15,  
"temp_max": 294.82, "pressure": 1014, "humidity": 72 }, "visibility": 10000, "wind":  
{ "speed": 3.6, "deg": 240 }, "clouds": { "all": 40 }, "dt": 1587308701, "sys":  
{ "type": 1, "id": 6796, "country": "IT", "sunrise": 1587270175, "sunset": 1587318896 },  
"timezone": 7200, "id": 3169070, "name": "Rome", "cod": 200 }
```



Esercizio

- Scrivi un WS documentato riceva in ingresso una città italiana e restituisca un messaggio con il tempo metereologico corrente





Giulio Angiani
I.F.O.A. - Reggio Emilia