



Giulio Angiani
Università di Parma

TePSIT





Web Services



Esercizio assegnato

- Scrivi un WS documentato riceva in ingresso una città italiana e restituisca un messaggio con il tempo metereologico corrente
- Cosa serve?
 - documentazione
 - uri
 - protocollo
 - input
 - output
 - implementazione



Esercizio assegnato

- documentazione
 - uri: **http://212.237.32.31/ws/weather/**
 - protocollo: **HTTP**
 - input:
 - parametro: **city**
 - tipoparametro: **string**
 - metodo http: **GET**
 - esempio di chiamata: **http://212.237.32.31/ws/weather/?city=Rome**
 - output:
 - formato: **JSON**
 - esempio:
 - in caso positivo : **{"city": "Rome", "weather": "cloudy", "timestamp": "2020-04-27 09:00:00"}**
 - in caso di errore : **{"error": 1, "error_msg": "valore non presente"}**



Architettura standard

- SOAP - Simple Object Access Protocol
- REST - Representational State Transfer



Architettura SOAP

- SOAP porta il proprio protocollo e si concentra su come esporre pezzi di logica applicativa (non dati) come servizi.
- SOAP espone operazioni.
- SOAP si occupa di esportare operazioni ben definite attraverso diverse interfacce
- Supporta WSDL per la documentazione dei servizi esposti
- Supporta principalmente XML
- Molto verboso



SOAP

- Esempio di un messaggio client SOAP ad un WS che espone un servizio di "Somma di due interi"
- Il servizio è esposto col nome di **Adder** ed ha come interfaccia due numero interi **a** e **b**

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Header/>
  <soap:Body>
    <Adder xmlns="http://www.mypc.com/sample.wsdl">
      <a>6</a>
      <b>2</b>
    </Adder>
  </soap:Body>
</soap:Envelope>
```



SOAP

- Esempio di un messaggio response SOAP alla chiamata precedente
- La risposta è inserita all'interno del tag **<AddResult>** figlio di **<AdderResponse>**

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <AdderResponse xmlns="http://www.mypc.com/sample.wsdl">
      <AddResult>8</AddResult>
    </AdderResponse>
  </soap:Body>
</soap:Envelope>
```

[esempio wsdl del servizio esposto](#)



SOAP : esempio in python - server

PYTHON

```
from pysimplesoap.server import SoapDispatcher, SOAPHandler
from BaseHTTPServer import HTTPServer

def sommatore(a,b):
    print("received", a, b)
    return a+b

dispatcher = SoapDispatcher('my_dispatcher', # internal http service name
    location = "http://localhost:8008/",
    action = 'http://localhost:8008/', # SOAPAction
    namespace = "http://www.mypc.com/sample.wsdl", prefix="ns0")
# register the user function : web service name CAN ABSOLUTELY BE different from real function name
dispatcher.register_function('Adder', sommatore,
    returns={'AddResult': int},
    args={'a': int, 'b': int})

# "Starting server..."
httpd = HTTPServer(("", 8008), SOAPHandler)
httpd.dispatcher = dispatcher
httpd.serve_forever()
```



SOAP : esempio in python - client

- esempio senza file wsdl

PYTHON

```
from pysimplesoap.client import SoapClient

# create a simple consumer
client = SoapClient(
    location = "http://localhost:8008/",
    action = 'http://localhost:8008/', # SOAPAction
    namespace = "http://www.mypc.com/sample.wsdl",
    soap_ns='soap',
    trace = False, ns = False) # trace = True for debugging

# call the remote method
response = client.Adder(a=6, b=2) #interface service and parameters name
# extract and convert the returned value
result = response.AddResult #interface output name
print int(result)
```



SOAP : esempio in python - client

- esempio con file wsdl

PYTHON

```
from pysimplesoap.client import SoapClient

client = SoapClient(wsdl="http://127.0.0.1:8008")
# tutto quello che c'è da sapere è pubblicato all'URI del wsdl
response = client.Adder(a=1,b=2)
result = response['AddResult']
print(result)
```



SOAP : esempio in python - client

- interrogando il file **wSDL** posso sapere tutto quello che serve dei servizi esposti

```
client = SoapClient(wsd="http://127.0.0.1:8008")  
print(client)
```

PYTHON

```
SOAP CLIENT  
ELEMENTS  
  Adder {a: int, b: int}  
  AdderResponse {AddResult: int}  
SERVICE (my_dispatcherService)  
  PORT (my_dispatcherBinding)  
    Location: http://localhost:8008/  
    Soap ver: soap11  
    Soap URI: http://schemas.xmlsoap.org/wsdl/soap/  
OPERATIONS  
  Adder(a: int, b: int)  
    > AdderResponse {AddResult: int}
```

OUTPUT



SOAP : esempio in python - client

- Se viene aggiunto un servizio al server SOAP, automaticamente sarà disponibile nel file wsdl

```
def moltiplicatore(a,b):  
    return a*b  
dispatcher.register_function('Multiplier', moltiplicatore,  
                             returns={'MulResult': int},  
                             args={'a': int, 'b': int}  
                             )
```

PYTHON (SERVER)

```
client = SoapClient(wsdl="http://127.0.0.1:8008")  
print(client)
```

PYTHON

```
[...]  
OPERATIONS  
Adder(a: int, b: int)  
> AdderResponse {AddResult: int}  
Multiplier(a: int, b: int)  
> MultiplierResponse {MulResult: int}
```

OUTPUT





Giulio Angiani
Università di Parma