

# Caracteristicile fundamentale ale unui serviciu web:

1. **opereaza conform paradigmei RPC** in care standardul de reprezentare universala a datelor are ca suport textul ASCII
2. **permite apeluri peste web** - comunicarea nu este obstructiunată de firewall-uri
3. **operează în arhitectură deschisă** – se asigură **interoperabilitate** totală, independență de platformă și de limbaj.

Este dificil de spus într-o manieră riguroasă dacă o tehnologie middleware este sau nu WS. In acest moment *serviciu* web este un termen foarte la modă și cu multă publicitate în jurul lui. După părerea unora despre servicii web se spune că este important si mare, dar nimeni nu poate spune exact ce e si de ce!?. Dupa unele spoturi publicitare, WS este noul tău prieten, soarta și salvarea pentru propriul business, noua generație Web etc. Evident, nimic din toate acestea nu spun ceea ce face de fapt un serviciu web. Si pentru a crește confuzia, mai ales în rândul celor mai puțin interesați în înțelegerea bazelor acestui concept, se folosesc o serie de acronime răsunătoare, ca UDDI, WSDL, WADL, XML, JSON, RPC, SOAP etc.

Nu încercăm să elucidăm noi subiectul. Preferăm să vedem *ce face* un astfel de serviciu, și, mai important, *cum poate fi folosit în aplicații*.

SOA: prezentare generala

<http://tutorials.jenkov.com/soa/index.html>

**Service Oriented Architecture (SOA)**

**Services vs. Applications**

**Enterprise Service Bus (ESB)**

**Service Composition**

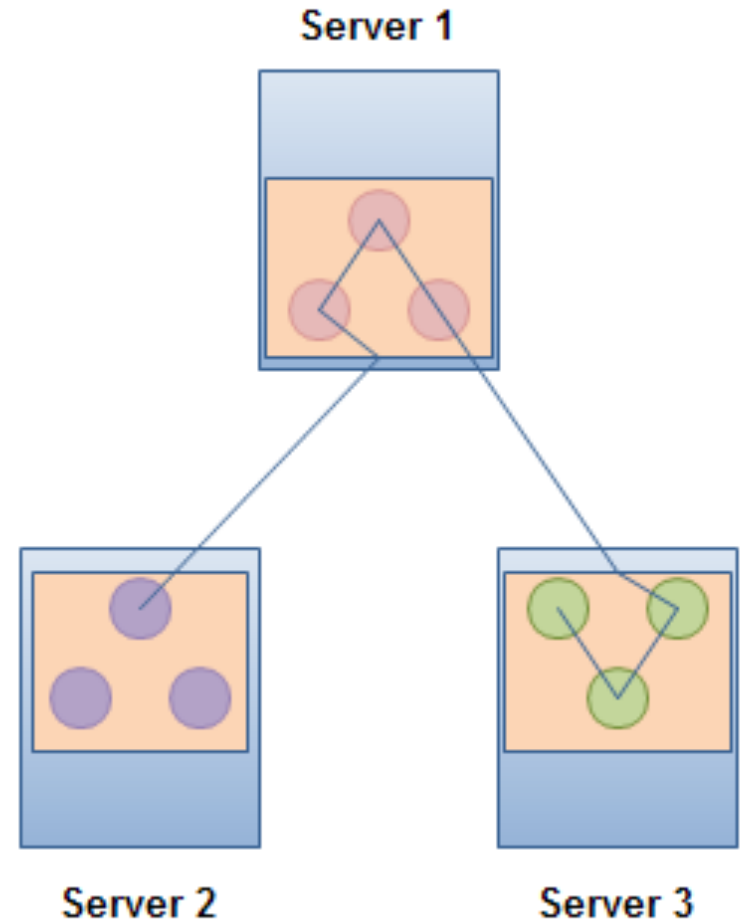
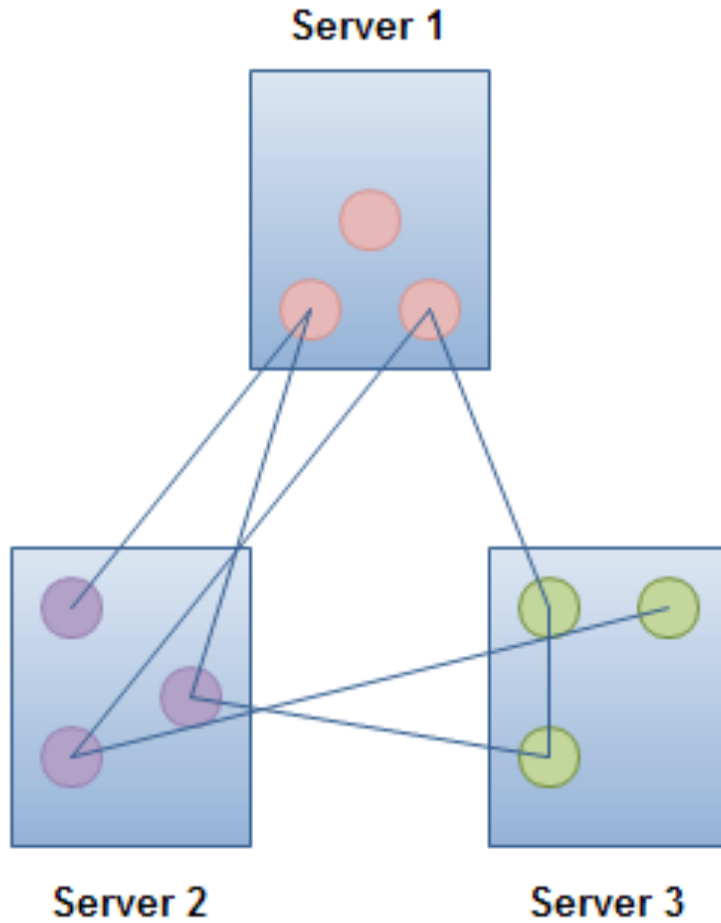
**Service Reusability**

**Service Transactions**

**Service Repositories**

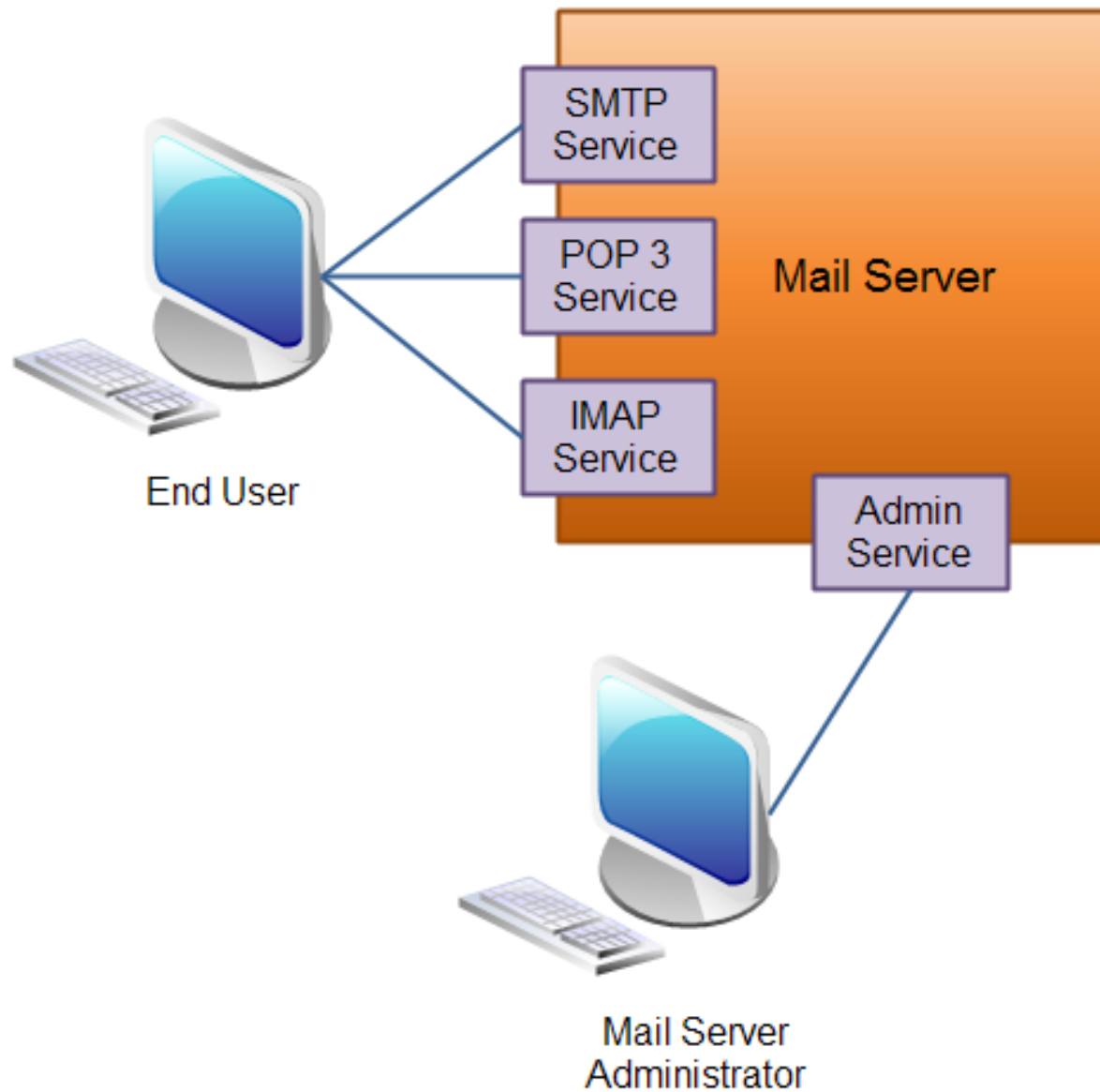
**Service Proxy**

## DOA (Distributed Object Architecture) vs. SOA

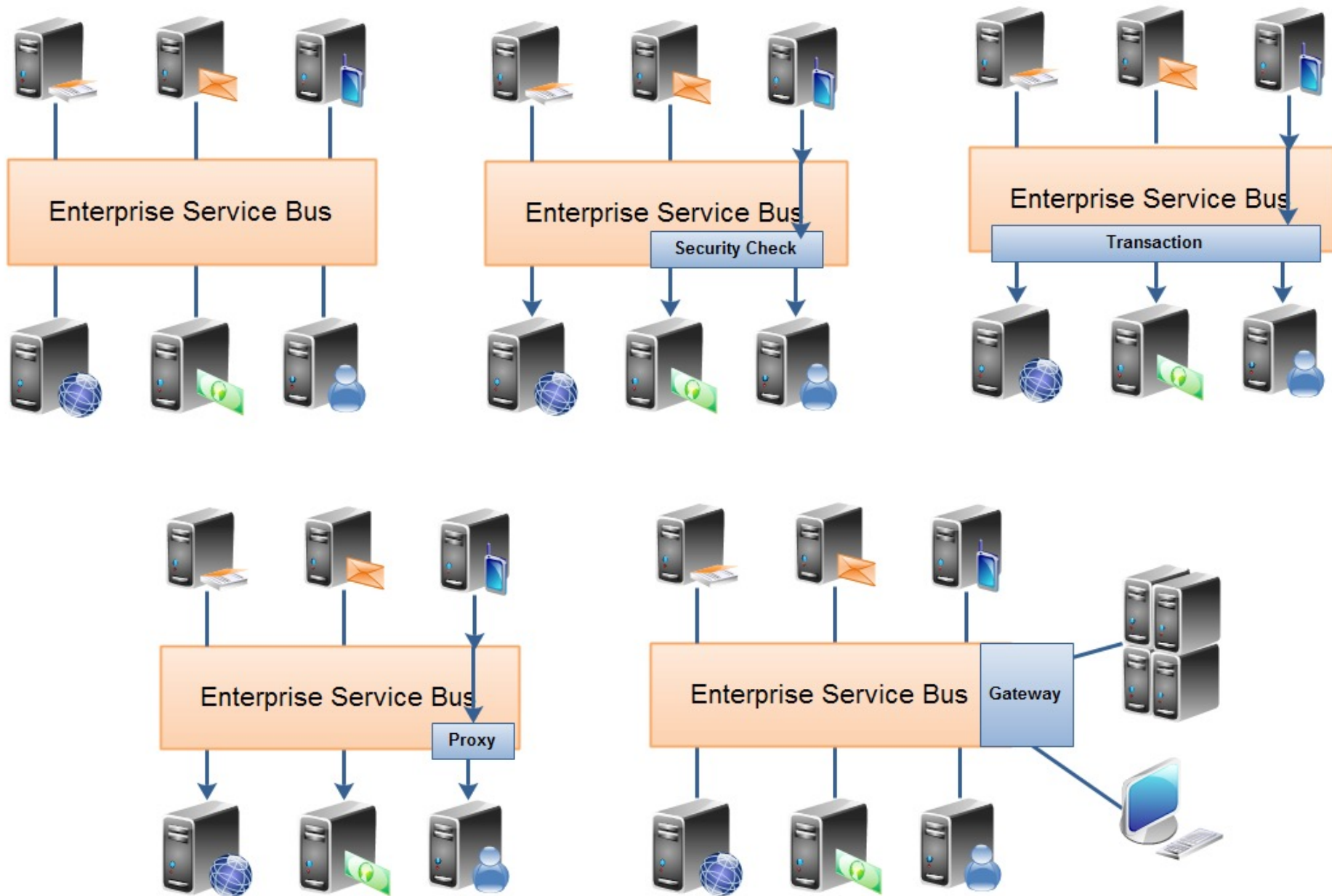


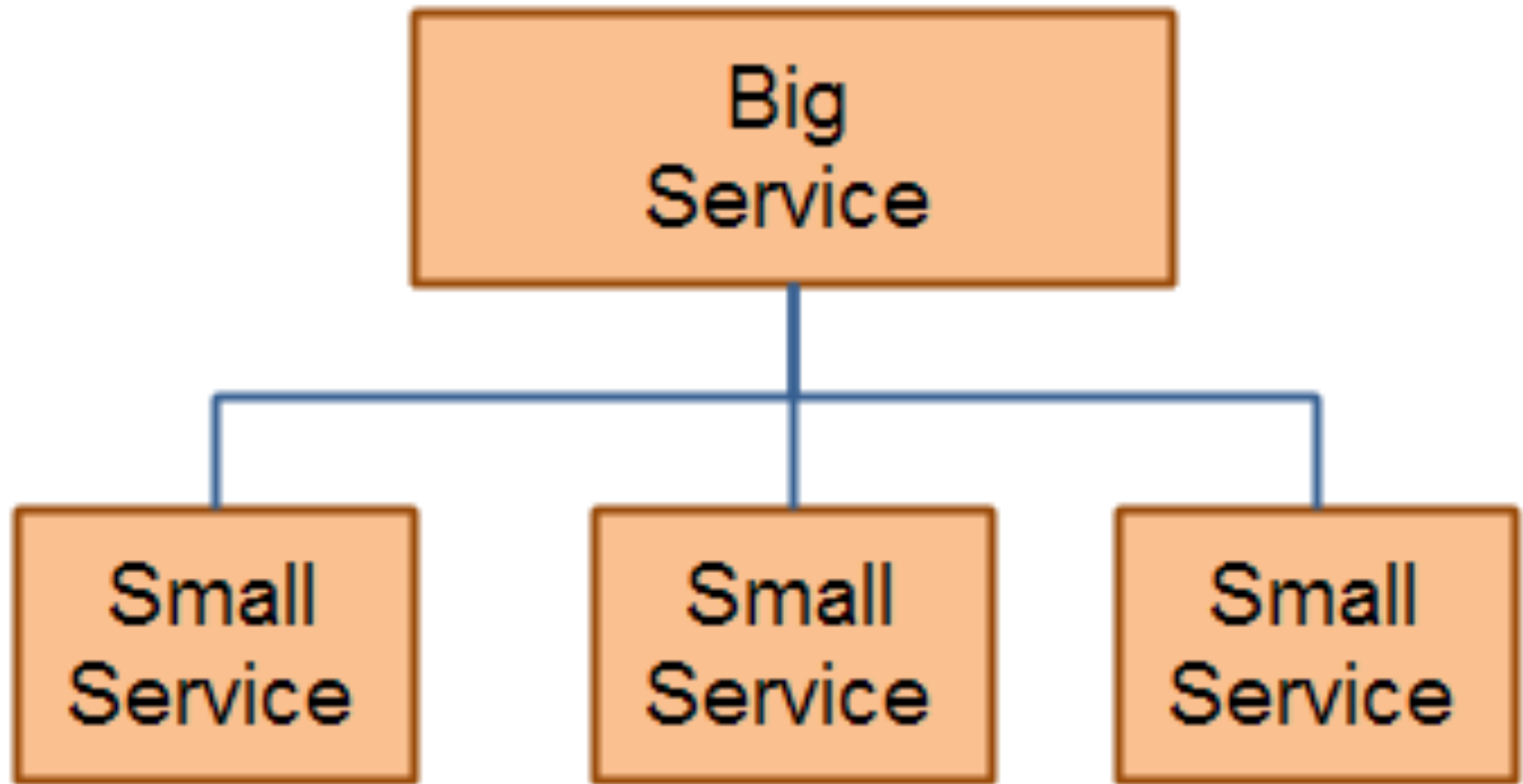
Servicii	Aplicatii
Efectueaza o singură operație sau câteva operații specializate.	Efectueaza o gamă largă de operații necesare rezolvarii problemei.
Cel mai adesea accesat de alte programe.	Adesea (dar nu întotdeauna) accesat de factorul uman.
Adesea (dar nu întotdeauna) țintește o parte din solutionarea unei probleme mai mari.	Adesea (dar nu întotdeauna) se urmărește solutionarea unei întregi probleme.

## O aplicatie care utilizeaza un server mail expune patru servicii



## ESB – Enterprise Service Bus – tinte:





### Application A

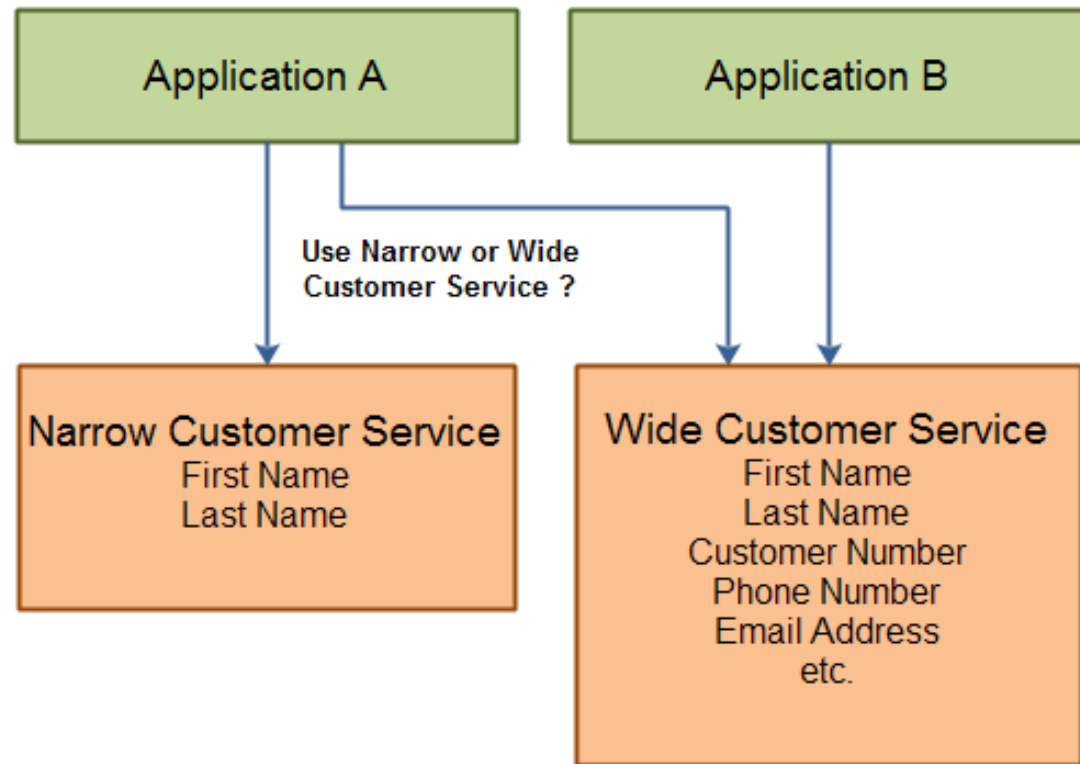
-----

- First Name
- Last Name

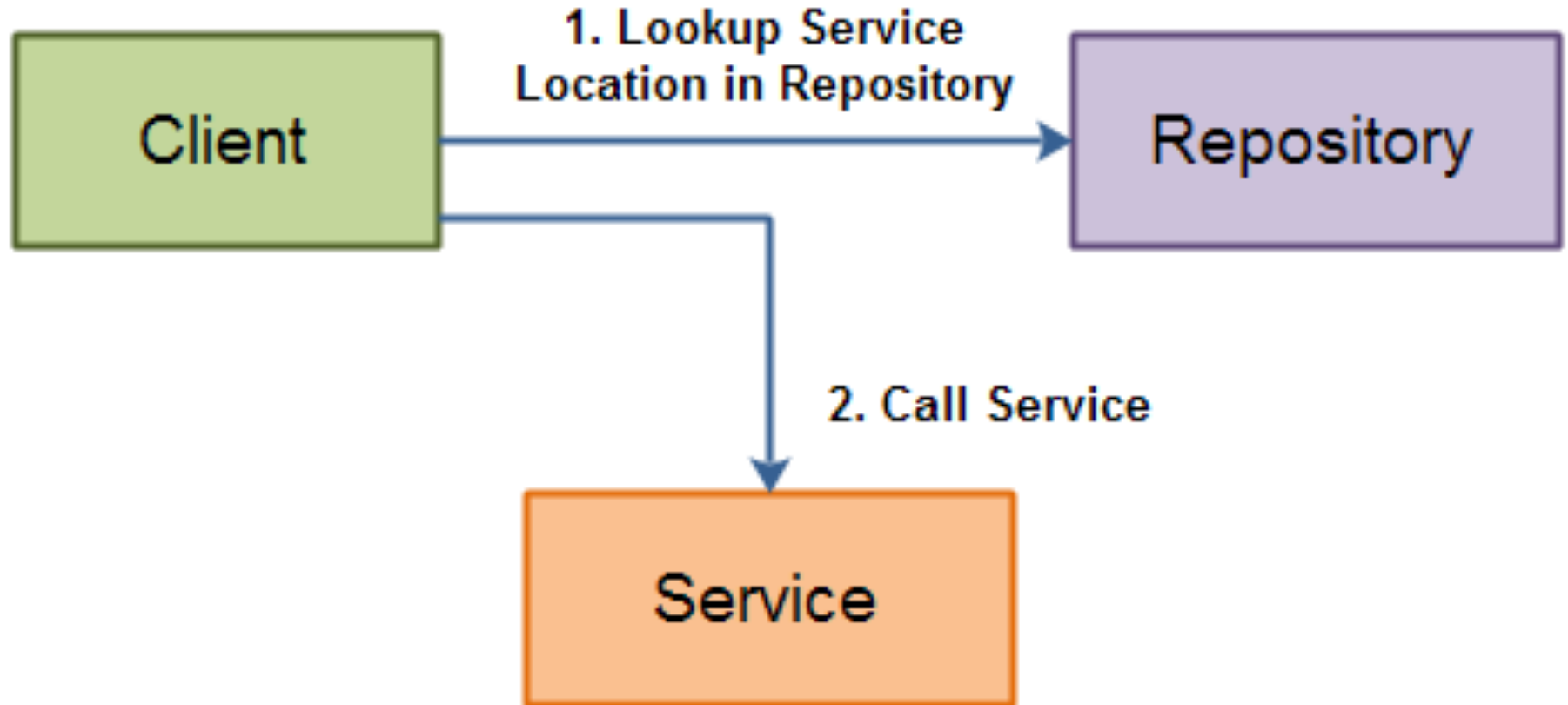
### Application B

-----

- First Name
- Last Name
- Customer Number
- Phone Number
- Email Address
- etc.

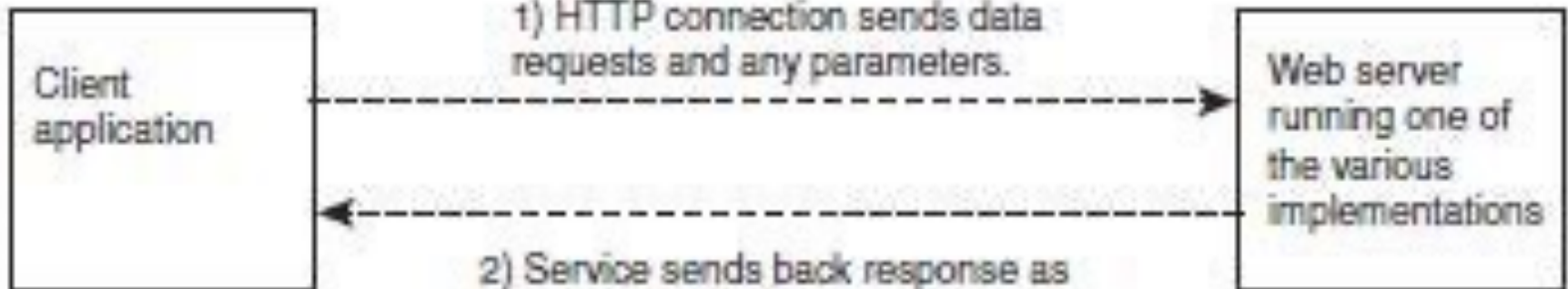
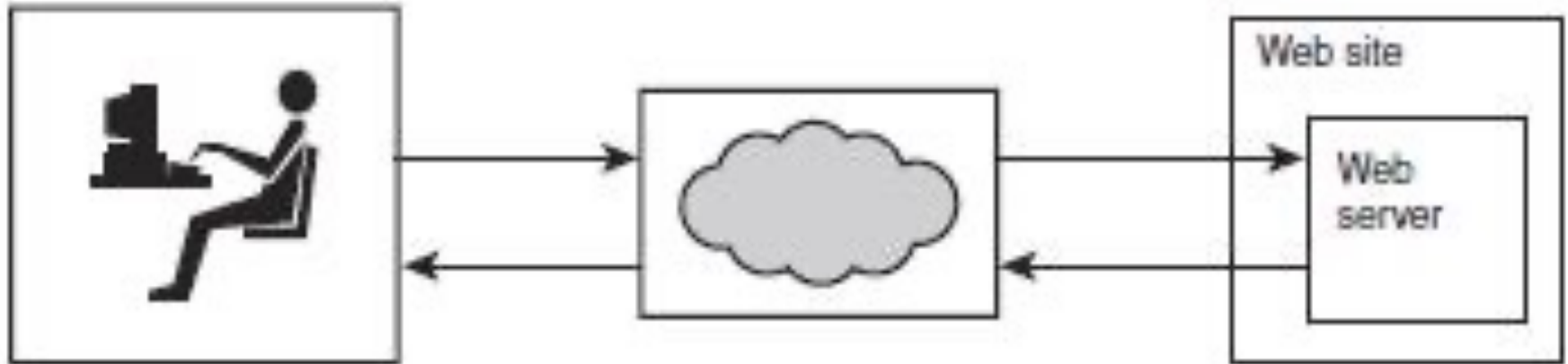






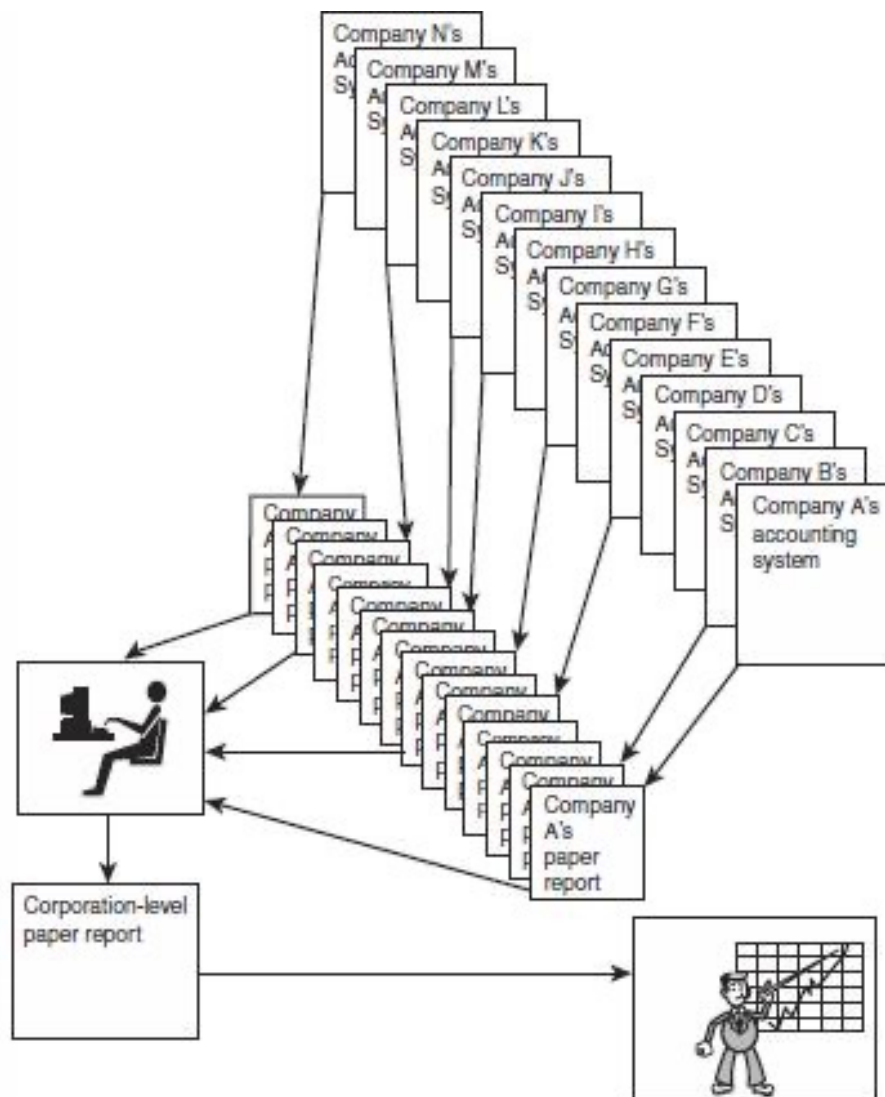


## Acces classic Web

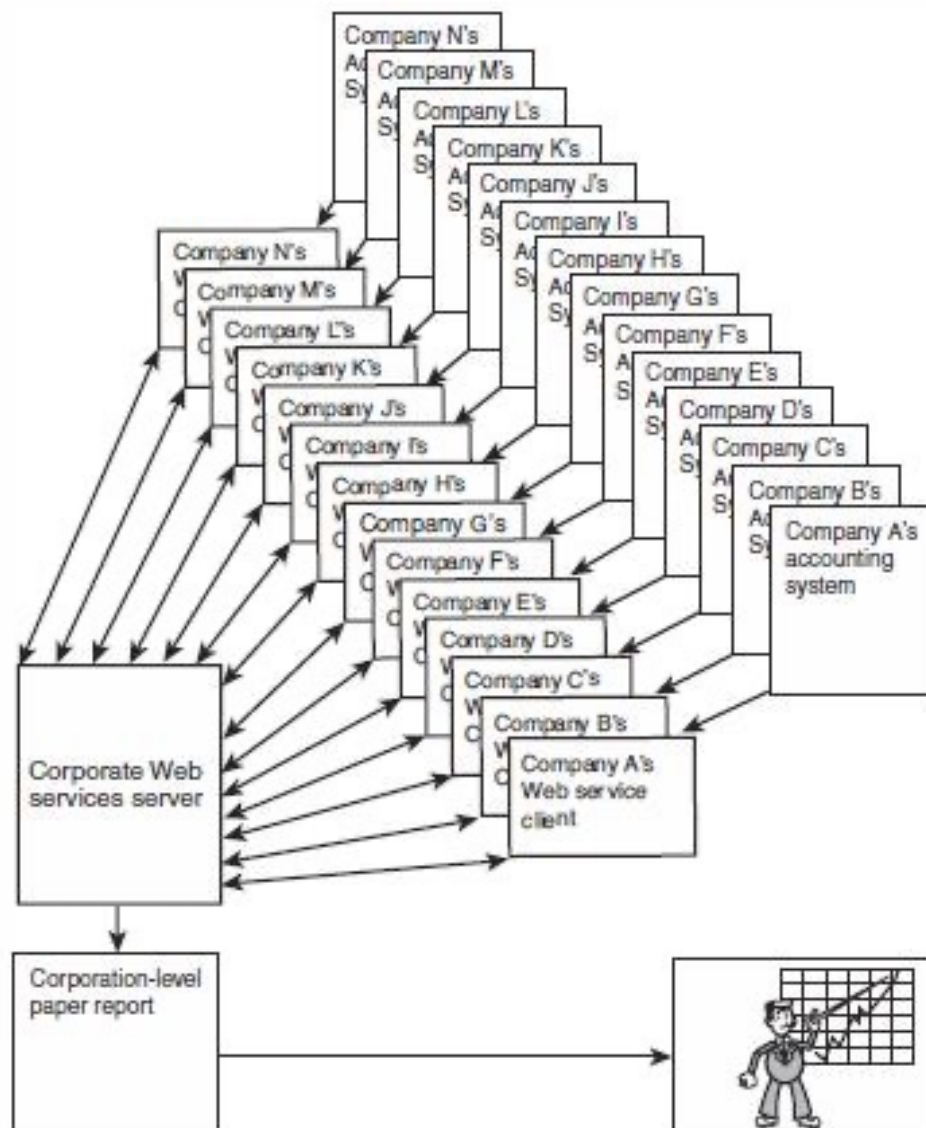


Once data is received by the client, all connections are then terminated.

# Exemplu clasic de aplicatie web (crearea unei corporatii din mai multe multinationale)



# Utilizarea WS in locul aplicatiei clasice pentru corporatii

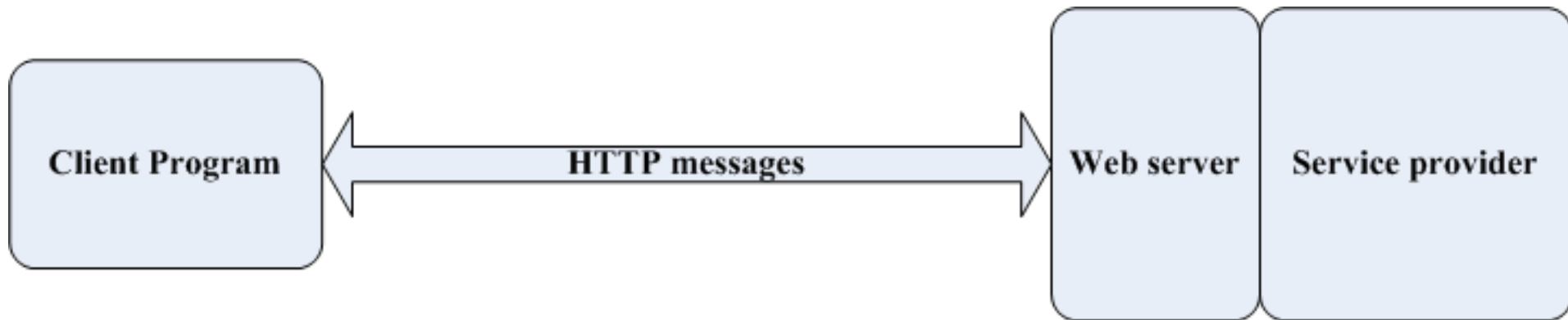


# WS:

Dispune de un API sau web API tipic, utilizat independent de platforma.

Este accesat via HTTP de un program client

Este executat la distanta, pe un sistem remote



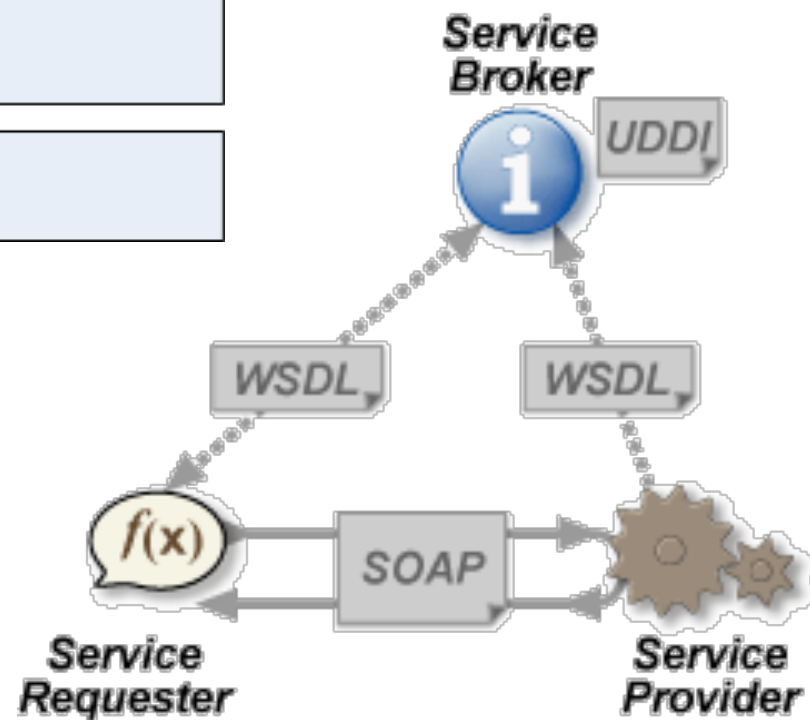
## WS technology stack

### Languages and protocols support:

Discovery	UDDI, -
Description	WSDL, WADL, XRD
Packages	SOAP, XMLRPC, XML, JSON, (X)HTML, text, -
Transport	HTTP (Web)
Network	TCP/IP (Internet)

### Predecesori WS (middleware):

- RPC
- RMI
- CORBA
- JMS
- Hessian
- Pyro
- Etc.



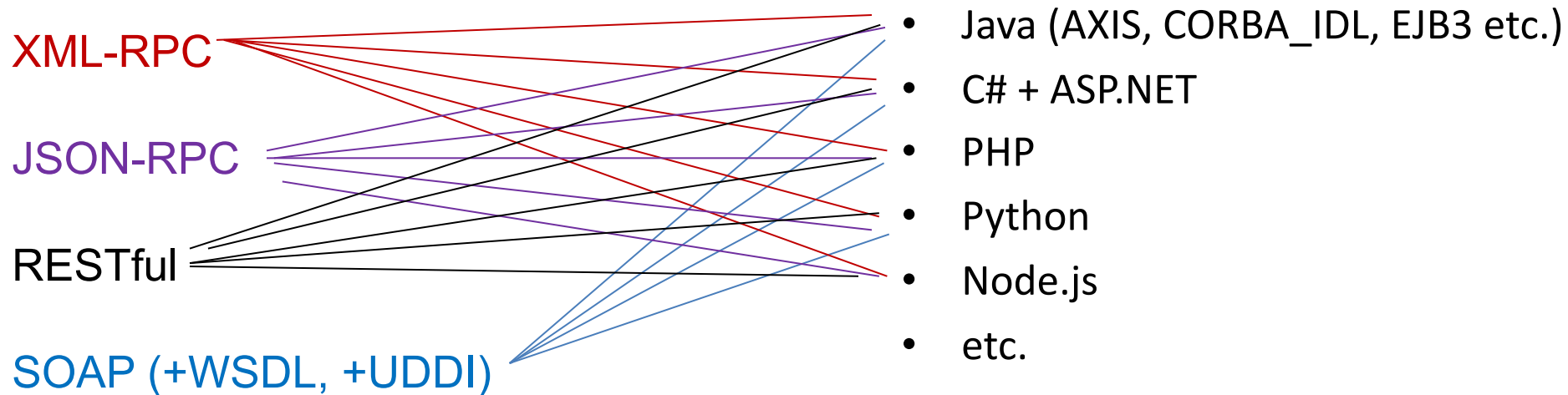
- Comunicare:
  - XML
  - SOAP (Simple Object Access Protocol)
  - JSON
  - Text
- Descriere serviciu (descrierea interfetelor cu WS):
  - WSDL (Web Service Description Language)
  - WADL (Web Application Description Language)
  - XRDl (Xml Rpc Description Language)
- Discovery Service:
  - UDDI (Universal Description Discovery and Integration)

Ce ofera SOAP (partial si celelalte standarde de comunicare)?

- Request / reply sincron ca si OOM
- Mesagerie asincrona ca si la MOM
- Suport de transport peste Internet (http, smtp etc.)
- Scheme XML de serializare obiecte si conversie de tipuri intre diverse limbaje



Dezavantaje:	Ce lipseste inca:
<p>Abstractizare low-level</p> <ul style="list-style-type: none"><li>• Este necesar un mare efort de implementare</li></ul> <p>Sunt necesare proiectarea unor noi modele de interactiune:</p> <ul style="list-style-type: none"><li>• Pe cand one-to-many, many-to-many?</li><li>• Notificare (informare data de serviciu)?</li></ul> <p>Nu se asigura transparenta fata de locatie.</p>	<p>Patterns general de interactiuni:</p> <ul style="list-style-type: none"><li>• Exista numai one-to-one si request-reply;</li><li>• Pe cand one-to-many, many-to-many?</li><li>• Notificare (informare data de serviciu)?</li><li>• Legare/dezlegare dinamica la serviciu:</li></ul> <p>Transparenta locatiei (cloud o rezolva partial)</p> <ul style="list-style-type: none"><li>• Anonimicitatea entitatilor care comunica</li></ul> <p>Suport pentru pervasive (omniprezent) computing:</p> <ul style="list-style-type: none"><li>• Valori de date de la senzori</li><li>• Software lightweight (usor)</li></ul>



**Big Web Services:**

**XML-RPC, JSON-RPC, SOAP.**

**Lightweit Web Services:**

**RESTful.**

**Servicii Web supersimple non MVC:**

SparkJava.

**Servicii Web si Middleware asincrone:**

Vert.x.

Earth

Google Search

UsersGateway

Qmath

```
import http.client
conn = http.client.HTTPConnection("new.earthtools.org")
conn.request("GET", "/timezone/46.75/23.58")
res = conn.getresponse()
print (res.status, res.reason)
print (res.read().decode())
conn.close()
```

```
200 OK
<?xml version="1.0" encoding="ISO-8859-1" ?>
<timezone xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://www.earthtools.org/timezone.xsd">
  <version>1.0</version>
  <location>
    <latitude>46.75</latitude>
    <longitude>23.58</longitude>
  </location>
  <offset>2</offset>
  <suffix>B</suffix>
  <localtime>9 Feb 2020 08:45:00</localtime>
  <isotime>2020-02-09 08:45:00 +0200</isotime>
  <utctime>2020-02-09 06:45:00</utctime>
  <dst>Unknown</dst>
</timezone>
```

```
import httpplib
conn = httpplib.HTTPConnection("new.earthtools.org")
conn.request("GET", "/sun/46.75/23.58/09/01/2/0")# zi/lu/fus/ora de vara
res = conn.getresponse()
print (res.status, res.reason)
print (res.read().decode())
conn.close()
```

200 OK

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<sun xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://www.earthtools.org/sun.xsd">
  <version>1.0</version>
  <location>
    <latitude>46.75</latitude>
    <longitude>23.58</longitude>
  </location>
  <date>
    <day>09</day>
    <month>01</month>
    <timezone>2</timezone>
    <dst>0</dst>
  </date>
  <morning>
    <sunrise>08:17:49</sunrise>
    <twilight>
      <civil>07:42:12</civil>
      <nautical>07:03:19</nautical>
      <astronomical>06:26:12</astronomical>
    </twilight>
  </morning>
  <evening>
    <sunset>16:44:45</sunset>
    <twilight>
      <civil>17:20:22</civil>
      <nautical>17:59:14</nautical>
      <astronomical>18:36:21</astronomical>
    </twilight>
  </evening>
</sun>
```

## WS Google search – Java

```
String wsdl = "http://api.google.com/GoogleSearch.wsdl";
URL url = new URL(wsdl);
QName serviceName = new QName("urn:GoogleSearch", "GoogleSearchService");
QName portName = new QName("urn:GoogleSearch", "GoogleSearchPort");
Service service = Service.create(url, serviceName);
Dispatch dispatch=service.createDispatch(portName, SOAPMessage.class, Service.Mode.MESSAGE);
MessageFactory mf = MessageFactory.newInstance();
SOAPMessage req = mf.createMessage(null, new FileInputStream(args[0]));
SOAPMessage res = (SOAPMessage)dispatch.invoke(req);
```

```
<?xml version='1.0' encoding='UTF-8'?>
<soap11:Envelope search.xml
  xmlns="urn:GoogleSearch"
  xmlns:soap11="http://schemas.xmlsoap.org/soap/envelope/">
  <soap11:Body>
    <doGoogleSearch>
      <key>4B0KufpQFHJxhAxzua0tR11EILNrHRJ6</key>
      <q>rares boian</q> <start>0</start>
      <maxResults>10</maxResults>
      <filter>true</filter><restrict/>
      <safeSearch>>false</safeSearch><lr/>
      <ie>latin1</ie> <oe>latin1</oe>
    </doGoogleSearch>
  </soap11:Body>
</soap11:Envelope>
```

```
<hostName xsi:type="xsd:string"></hostName>
<relatedInformationPresent
  xsi:type="xsd:boolean">true</relatedInformation
Present>
<snippet
  xsi:type="xsd:string">&lt;b>Rares&/b>
Florin &lt;b>Boian&/b> CAIP Center,
Rutgers University 96 Frelinghuysen Road,
Room &lt;br> 703 Piscataway, NJ 08854
Work: 1-732-445-0561 Home: 1-732-373-0026
&lt;b>...&/b></snippet>
<summary xsi:type="xsd:string"></summary>
<title xsi:type="xsd:string">Homepage -
&lt;b>Rares Boian&/b></title>
</item>
<item xsi:type="ns1:ResultElement">
```



[Web](#) [Images](#) [Groups](#) [News](#) [Scholar](#) [more »](#)

rares boian

Search

[Advanced Search](#)  
[Preferences](#)

Web

Results 1 - 10 of about 14,800 for rares boian. (0.11 seconds)

[Homepage - Rares Boian](#)

Rares Florin Boian CAIP Center, Rutgers University 96 Frelinghuysen Road, Room 703

Servicii Web\_22/26\_

[authUser](#)(String user, String password)

Autentifica in retea LAN (folosind LDAP sau FTP) *user* cu parola *password*.

[getAllAni](#)(String user, String password)

Intoarce anii in format CSV (Comma Separator Values)

[getAllCatedre](#)(String user, String password)

Intoarce toate catedrele in format CSV (Comma Separator Values)

[getAllFtpuri](#)(String user, String password)

Intoarce toate ftpurile in format CSV (Comma Separator Values)

[getAllGrupe](#)(String user, String password)

Intoarce toate grupele in format CSV (Comma Separator Values)

[getAllHosturi](#)(String user, String password)

Intoarce toate hosturile in format CSV (Comma Separator Values)

[getAllMailuri](#)(String user, String password)

Intoarce toate hosturile de mail in format CSV (Comma Separator Values)

[getAllSpecializari](#)(String user, String password)

Intoarce toate specializarile in format CSV (Comma Separator Values)

[getUsers](#)(String user, String password, java.lang.String filtru)

Lista utilizatorilor care satisfac *filtru* solicitata de *user*, cu parola *password*.

## WS - UsersGateway – exemplu Python

```
import getpass          \ import xmlrpclib
USER = "florin";         \ PAROLA = getpass.getpass()
URL = "http://www.scs.ubbcluj.ro:7777/";
proxy = xmlrpclib.Server(URL)
raspuns = proxy.UsersGateway.getUsers( USER, PAROLA, \
    "{nume=Po.*|Mi.*,host=scs.ubbcluj.ro,grupa=gr244|gr254}" )
print (raspuns)
```

d:\UserGatewayClients>python ugclient.py

Password:

```
[
{"grupa":"gr244","mail":"masd1469@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"4","nume":"Mihai D Alexandru Ionut","user":"masd1469","catedra":"-"},
{"grupa":"gr254","mail":"pesd1298@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"5","nume":"Pop B. Emilia Maria","user":"pesd1298","catedra":"-"},
{"grupa":"gr254","mail":"pasd1299@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"5","nume":"Pop D. Alexandra Flavia","user":"pasd1299","catedra":"-"},
{"grupa":"gr254","mail":"pcsd1300@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"5","nume":"Pop I. Cristian Nicolae","user":"pcsd1300","catedra":"-"},
{"grupa":"gr254","mail":"ptsd1301@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"5","nume":"Pop V. Tudor Marian","user":"ptsd1301","catedra":"-"},
{"grupa":"gr244","mail":"pmsd1480@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"4","nume":"Popa G Marina Teodora","user":"pmsd1480","catedra":"-"},
{"grupa":"gr254","mail":"pmsd1302@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"5","nume":"Popa C. Mihai","user":"pmsd1302","catedra":"-"},
{"grupa":"gr254","mail":"pbsd1303@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"5","nume":"Popescu D. Bianca","user":"pbsd1303","catedra":"-"},
{"grupa":"gr254","mail":"pfsd1304@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"5","nume":"Porutiu A.T. Flaviu","user":"pfsd1304","catedra":"-"},
{"grupa":"gr244","mail":"pmsd1481@scs.ubbcluj.ro","host":"scs.ubbcluj.ro","specializare":"sd","an":
:"4","nume":"Potor D Marius Daniel","user":"pmsd1481","catedra":"-"}
]
```



## Serviciul ofera:

- Aritmetica peste Q (suma, diferenta, produs, cat, putere, divizori, cmmdc etc.).
- Aritmetica de polinoame peste Q (suma, diferenta, produs, cat, cmmdc, compunere, radacini rationale etc.).
- Aritmetica matriceala peste Q (suma, produs, putere, rang, determinant, inversa, simplex, rezolvari de sisteme etc.)

## Implementare in java ca servicii JsonRpc.

## Exemplu de client:

```
import http.client, sys, json
class Clie:
    def urlCall(self, urlServ, metoda, paramS):
        - - -
    def __init__(self, urlServ):
        - - -
        self.urlCall(urlServ, "leastquaressystem", "a=[[1, -1],[1 ,0],[1, 1]]&b=[1,
1 2]")
        - - -
```

```
leastquaressystem ({ "a": "[[1, -1],[1 ,0],[1, 1]]", "b": "[1, 1 ,2]" }) ==>
{"x1": " (4/3) ", "x2": " (1/2) "}
```