Resource Oriented Architecture (ROA)

ROA

- Understand the ROA architecture

ROA

- Understand the ROA architecture

- Understand RESTful Web services

ROA

- Understand the ROA architecture

- Understand RESTful Web services

- Understand its value added to the service-based application

SOAP services : for or against?

- Standardized protocols

- Rely on a generic transport layer

- Stateful operations

- Standards to ensure a high level of security and reliability

SOAP services : for or against?

- But !!
  - Complex to set up

  - Use of a SOAP Envelope

  - Not adequate for restricted environment (bandwidth)

  - XML too verbose

INSA

SOAP services : for or against?

- But !!
  - Complex to set up

  - Use of a SOAP Envelope

  - Not adequate for restricted environment (bandwidth)

  - XML too verbose

- **RESTful Web services (Representational State Transfer)**

Resource Oriented Architecture (ROA)

RESTfull architecture

- REST is not a protocol, nor a standard, nor a format

- It's an **architectural style**
- A set of best practices for designing Web services

- Operations are **stateless**

- REST uses **HTTP specifications**

HTTP

- **H**yper**T**ext **T**ransfer **P**rotocol
- Communications between web servers and clients
- A set of rules used to formalize Web communications

HTTP verbs
- GET
- POST
- PUT
- …

HTTP code
- 200
- 201
- 400
- …

request

reply

Web Client

Server

HTTP

1. Using a browser to put the URL :
http://host:port/chemin/file

RESOURCE

3. The server uses the URL to redirect to the requested file

2. Send a request GET URL HTTP/
HOST: host:port
…

request

reply

Web Client

Server

5. Display the response

4. Send the response HTTP 200 OK
…

Resource

- The notion of resource is central in REST architectures

- Any entity (information, object, application) is seen as a resource

- Use of operations like in Databases
  - Create
  - Read
  - Update
  - Delete

Resource operations

- **CRUD** operation
  - **C**reate ————————→
  - **R**ead ————————→
  - **U**pdate ————————→
  - **D**elete ————————→

- **HTTP** verbs
  - POST
  - GET
  - PUT
  - DELETE

Resource operations

- **CRUD** operation
  - **C**reate →
  - **R**ead →
  - **U**pdate →
  - **D**elete →

- **HTTP** verbs
  - POST
  - GET
  - PUT
  - DELETE

- A resource is accessible via a **URI** (Uniform Resource Identifier)

URL, URN, and URI?

URL

- **URL** : Uniform Resource Locator

- Identifies the location of a resource.

- Example : http://homepages.laas.fr/nguermou/TDs

URN

INSA

- **URN:** Uniforme Resource Name

- Identifier of a resource

- It does not enable identifying the location

- Example : ISBN of a book

URI

- **URI:** Uniform Resource Identifier

- Generalizes URN and URL

- Identifies a resource by location, name, or both

INSA

INSTITUT NATIONAL
DES SCIENCES
APPLIQUÉES
TOULOUSE