





Sistemas Distribuídos e Concorrentes

Prof. Luiz A. de P. Lima Jr.
luiz.lima@pucpr.br

Escola Politécnica - PUCPR

RESTful Web Services



O que é REST?

- ◉ Tese de doutorado de **Roy Fielding**
- ◉ Descreve **estilo de arquitetura** de sistemas em rede.
- ◉ REST = “***REpresentational State Transfer***”



Conceitos Básicos

- ◉ **Web**: formada de recursos
- ◉ **Recursos** = qualquer item de interesse (i.e., informação com *hiperlinks*)
- ◉ Recursos não são úteis a menos que se tenha uma **representação** para eles.
- ◉ Representações: **tipos MIME**
- ◉ Recursos possuem **estado**.
 - Representações devem capturar o estado do recurso.
- ◉ **REST**: uso de **substantivos** (“recursos”) ao invés de verbos (operações)

```
<user>
  <id>1</id>
  <name>Fulano</name>
  <profession>teacher</profession>
</user>
```

<https://www.sitepoint.com/mime-types-complete-list/>



REST x SOAP

- Mais **simples** que SOAP
 - WSDL e esquemas XML não são essenciais
 - **REST**: “*Lightweight Web Services*”

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-
envelope"
soap:encodingStyle="http://www.w3.org/2001/12/
soap-encoding">
  <soap:body pb:http://www.banco.com/contas">
    <pb:ObtemSaldo>
      <pb:ID>1234-5</pb:ID>
    </pb:ObtemSaldo>
  </soap:Body>
</soap:Envelope>
```

SOAP

REST

<http://www.banco.com/contas/1234-5>



- ◉ Equivalentes em **funcionalidade**:
 - Tudo que pode ser feito com SOAP, pode também ser feito com uma arquitetura RESTful.
- ◉ SOAP = “**carta**” (c/ envelope, anexos, etc.)
- ◉ REST = “**cartão postal**”
(gasta menos “papel”, i.e., largura de banda)
- ◉ Mesmo nível de **segurança** que SOAP



- ◉ Não usa **XML** necessariamente (especialmente para requisições):
 - Formatos possíveis:
 - ✦ *CSV (Comma-Separated Values)*
 - ✦ *JSON (JavaScript Object Notation)*
- ◉ Recursos x métodos ou serviços:
 - **SOAP:** `obtemProduto()` + `obtemPreço()`
 - **REST:** `produto` = recurso com todas as informações (ou *links* para elas)



Exemplo de Recurso

- ◉ Lista de livros de SD em uma biblioteca
- ◉ Acesso ao **recurso**:
 - <http://www.bib.com/sd>
- ◉ Exemplo de **representação** do recurso:
 - [sd.html](#) (MIME: [text/html](#))
- ◉ A representação coloca o **cliente** em um **ESTADO**

(obtida no ponto
de acesso)



Exemplo de Recurso (cont.)

- ◉ Ao acessar um recurso e obter a sua representação, o **cliente muda de estado**.
- ◉ REST, segundo Roy Fielding:

“REST procura capturar a imagem de como **aplicações web bem projetadas** se comportam: uma rede de **páginas web** (uma máquina virtual de estados), onde o usuário progride em uma aplicação por meio da **seleção de links** (transições de estado) resultando na **próxima página** (representando o próximo estado da aplicação) sendo transferida ao usuário e apresentada para o seu uso.”



- ◉ Capturar as **características da web** que a fizeram bem sucedida...
 - e usar tais características para conduzir a evolução da web: **WS**