Having Fun with the RestTemplate

Arjen Poutsma
SpringSource
VMware
Overview

- REST in 10 minutes
- RestTemplate
REST in 10 minutes
Identification

• Resources
• Typically nouns
• URIs

http://example.com/customers/1234
http://example.com/orders/2007/10/776654
Uniform Interface
Uniform Interface
<table>
<thead>
<tr>
<th>Action</th>
<th>HTTP Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>GET</td>
<td></td>
</tr>
<tr>
<td>Create or Update</td>
<td>PUT</td>
<td></td>
</tr>
<tr>
<td>Create</td>
<td>POST</td>
<td></td>
</tr>
<tr>
<td>Delete</td>
<td>DELETE</td>
<td></td>
</tr>
</tbody>
</table>
Uniform Interface
Uniform Interface
HATEOAS

- Hypermedia as the Engine of Application State
- No server-side state
- State transfer through links
- Google
GET /service/customers/1234 HTTP 1.1
Host: www.example.com
User-Agent: XYZ 1.1
Accept: text/html,application/xml;q=0.9,*/*;q=0.8
If-Modified-Since: Thu, 18 Dec 2008 16:47:31 GMT
If-None-Match: "600028c-59fb-474f6852c9dab"

HTTP/1.1 304 Not Modified
Date: Fri, 19 Dec 2008 19:36:25 GMT
Last-Modified: Thu, 18 Dec 2008 16:47:31 GMT
Etag: "600028c-59fb-474f6852c9dab"
Content-Length: 7160
Content-Type: application/xml

<?xml version='1.0' encoding='utf-8'?>
REST
REST
RestTemplate
RestTemplate

- Introduced in Spring 3.0
- RestTemplate as core client-side component
- Similar to other templates in Spring
  - JdbcTemplate
  - JmsTemplate
RestTemplate methods

- getForObject
  - Performs GET and converts
- put
  - Performs PUT
- postForLocation
  - Performs POST, and retrieves Location header
- delete
RestTemplate

String uri = "http://example.com/hotels/{id}"
template = new RestTemplate();
HotelList result = template.getForObject(uri, HotelList.class, "1");

Booking booking = // create booking object
uri = "http://example.com/hotels/{id}/bookings";
Map<String, String> vars = Collections.singletonMap("id", "1");
URI location = template.postForLocation(uri, booking, vars);

template.delete(location, );

template.execute(uri, HttpMethod.GET,
    myRequestCallback,
    myResponseCallback);

template.execute(uri, HttpMethod.GET,
    myRequestCallback,
    myResponseCallback);
Demo’s
Q & A