《分布式计算》

RESTful 客户端编程

学 院 名 称 ：	数据科学与计算机学院
成 员 ：	陈伟宸
时 间 ：	2016 年 9 月 17 日

1. 执行语句如下：

```
Administrator@WIN-4F1ASE5GDRE MSYS ~
$ curl -v -H "Content-Type: application/xml; charset=ISO-8859-1"  -T "C:\Users\Administrator\workspace\apache-cxf-3.1.7\samples\jax_r
s\basic\src\main\resources\update_customer.xml" "http://localhost:9000/customerservice/customers"
```

返回如下：

```
* STATE: INIT => CONNECT handle 0x600057290; line 1103 (connection #-5000)
* Added connection 0. The cache now contains 1 members
*   Trying ::1...
* STATE: CONNECT => WAITCONNECT handle 0x600057290; line 1156 (connection #0)
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 9000 (#0)
* STATE: WAITCONNECT => SENDPROTOCONNECT handle 0x600057290; line 1253 (connection #0)
* STATE: SENDPROTOCONNECT => DO handle 0x600057290; line 1271 (connection #0)
> PUT /customerservice/customers HTTP/1.1
> Host: localhost:9000
> User-Agent: curl/7.46.0
> Accept: */*
> Content-Type: application/xml; charset=ISO-8859-1
> Content-Length: 84
> Expect: 100-continue
>
* STATE: DO => DO_DONE handle 0x600057290; line 1350 (connection #0)
* STATE: DO_DONE => WAITPERFORM handle 0x600057290; line 1477 (connection #0)
* STATE: WAITPERFORM => PERFORM handle 0x600057290; line 1487 (connection #0)
* HTTP 1.1 or later with persistent connection, pipelining supported
< HTTP/1.1 100 Continue
* We are completely uploaded and fine
* HTTP 1.1 or later with persistent connection, pipelining supported
< HTTP/1.1 200 OK
< Date: Tue, 04 Oct 2016 08:56:54 GMT
< Content-Length: 0
* Server Jetty(9.2.15.v20160210) is not blacklisted
< Server: Jetty(9.2.15.v20160210)
<
* STATE: PERFORM => DONE handle 0x600057290; line 1645 (connection #0)
* Curl_done
* Connection #0 to host localhost left intact
* Expire cleared
```

服务端日志记录：

```
 ----invoking updateCustomer, Customer name is: Mary
```

2. 对于 1 中的 curl 命令，

```
Administrator@WIN-4F1ASE5GDRE MSYS ~
$ curl -v -H "Content-Type: application/xml; charset=ISO-8859-1"  -T "C:\Users\Administrator\workspace\apache-cxf-3.1.7\samples\jax_r
s\basic\src\main\resources\update_customer.xml" "http://localhost:9000/customerservice/customers"
```

-H 用于在 HTTP header 中指定程序中指定的文件类型、字符集，对应代码

```
RequestEntity entity = new FileRequestEntity(input, "text/xml; charset=ISO-8859-1");
```

-T 用于指定上传的文件和 URL，对应代码

```
String inputFile = client.getClass().getResource("/update_customer.xml").getFile();
URIResolver resolver = new URIResolver(inputFile);
File input = new File(resolver.getURI());
PutMethod put = new PutMethod("http://localhost:9000/customerservice/customers");
```

程序首先将需要上传的文件加载进来，之后将该文件和对 HTTP header 的设置封装入 put 方法中，最后使用 HttpClient 发送该请求

3. 解析指定文件或资源路径，为 POST 和 PUT 方法的实体做准备

4. 缓存输出流，提供缓存，方便一次性返回全部结果

5. （1）简洁；（2）松散的耦合；（3）客户端和服务端通过一组固定的 HTTP 方法发送 HTTP 请求和 HTTP 响应；（4）可返回多种媒体 I 类型；（5）文档内容可以链接到更多的资源

6. 从可感知的 URI 模板实用类中构造 URI

7. 假设有一个旅行社的信息系统是由多个基础的服务构成的：客户服务、目的地服务、天气服务和报价服务。现在需要实现一个功能：对于一个客户，需要为他展示最近 10 个到访过的地方、为他推荐 10 个目的地并提供目的地的天气预报和报价。留意到这些请求是有一定的顺序要求的。

在同步的实现中，对各个基础服务的请求是按顺序的，实现的代码可读性强，但是运行时间较长；

在异步的实现中，对各个基础服务的请求是异步进行的，实现的代码可读性差，但是效率高；

而 Reactive 客户端的实现囊括了同步和异步实现的优点，实现的代码运行时间与异步实现相当，而可读性甚至比同步实现更强

8. 运行结果如下：

```
{
  "visited" : [ {
    "destination" : "Bulgaria"
  }, {
    "destination" : "Bahrain"
  }, {
    "destination" : "Germany"
  }, {
    "destination" : "Nicaragua"
  }, {
    "destination" : "Thailand"
  } ],
  "recommended" : [ {
    "destination" : "Latvia",
    "forecast" : "Flurries",
    "price" : 6426
  }, {
    "destination" : "Kenya",
    "forecast" : "Mostly Cloudy",
    "price" : 966
  }, {
    "destination" : "Bosnia & Herzegovina",
    "forecast" : "Dust",
    "price" : 6182
  }, {
    "destination" : "Cambodia",
    "forecast" : "Rain and Snow",
    "price" : 7353
  }, {
    "destination" : "United States of America",
    "forecast" : "Rain and Snow",
    "price" : 4323
  } ],
  "processingTime" : 5134
}
```

```
{
  "visited" : [ {
    "destination" : "Italy"
  }, {
    "destination" : "South Sudan"
  }, {
    "destination" : "Maldives"
  }, {
    "destination" : "Liberia"
  }, {
    "destination" : "Tuvalu"
  } ],
  "recommended" : [ {
    "destination" : "Nepal",
    "forecast" : "Sunny",
    "price" : 3100
  }, {
    "destination" : "Japan",
    "forecast" : "Cloudy",
    "price" : 6591
  }, {
    "destination" : "Armenia",
    "forecast" : "Haze",
    "price" : 7547
  }, {
    "destination" : "Antigua & Barbuda",
    "forecast" : "Scattered Thunderstorms",
    "price" : 7574
  }, {
    "destination" : "Bhutan",
    "forecast" : "Freezing Drizzle",
    "price" : 1445
  } ],
  "processingTime" : 944
}
```

```
{
  "visited" : [ {
    "destination" : "The Netherlands"
  }, {
    "destination" : "Cambodia"
  }, {
    "destination" : "Yemen"
  }, {
    "destination" : "Kenya"
  }, {
    "destination" : "Bhutan"
  } ],
  "recommended" : [ {
    "destination" : "Brunei",
    "forecast" : "Sleet",
    "price" : 7098
  }, {
    "destination" : "Andorra",
    "forecast" : "Thunderstorm",
    "price" : 2235
  }, {
    "destination" : "Colombia",
    "forecast" : "Dust",
    "price" : 7353
  }, {
    "destination" : "Somalia",
    "forecast" : "Chance of Snow",
    "price" : 1022
  }, {
    "destination" : "Bangladesh",
    "forecast" : "Smoke",
    "price" : 9894
  } ],
  "processingTime" : 919
}
```
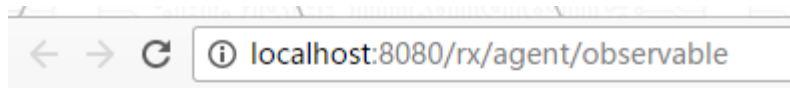
```
{
  "visited" : [ {
    "destination" : "Cambodia"
  }, {
    "destination" : "Tanzania"
  }, {
    "destination" : "Indonesia"
  }, {
    "destination" : "Solomon Islands"
  }, {
    "destination" : "Kiribati"
  } ],
  "recommended" : [ {
    "destination" : "Azerbaijan",
    "forecast" : "Storm",
    "price" : 6731
  }, {
    "destination" : "Croatia",
    "forecast" : "Haze",
    "price" : 4269
  }, {
    "destination" : "Iraq",
    "forecast" : "Smoke",
    "price" : 4115
  }, {
    "destination" : "Bolivia",
    "forecast" : "Chance of Storm",
    "price" : 5643
  }, {
    "destination" : "Japan",
    "forecast" : "Snow",
    "price" : 1536
  } ],
  "processingTime" : 1007
}
```

9. 通常 web 程序的接口与实现是分开的，实现一般来说不是一个人可以完成的，而且实现会环境的不同实用不同的方式，也可能随着时间的流逝 web 程序会需要更多的功能，先声明接口可以很大程度减少代码间的耦合度，使代码更容易重用。

10. JAXRSClientFactory 提供了大量的方法用于构建 JAX-RS 客户端对象，如：

```
BookStore store = JAXRSClientFactory.create("http://bookstore.com", BookStore.class);
```

11. 透明代理指的是对于客户端来说，代理可以是完全不可见的，客户端根本不需要知道代理服务器的存在，代理服务器在客户端不知情的情况下帮助客户端完成对服务器的访问。

小结

1. 学习了如何使用 curl 命令发送指定头部和 HTTP 方法的 HTTP 请求

2. 学习了 jersey 客户端的编写

3. 学习了 Reactive 客户端的编写

4. 学习了动态代理的原理

5. 对正向代理、反向代理和透明代理进行了区分