

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

REST

REST = Representational State Transfer

An architectural pattern for the web

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

Agenda

- I. What is REST?
- II. Why REST?
- III. Where is REST used?
- IV. Uniform Operations
- V. Conclusion
 - I. When to REST? And when not to..

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

What Rest is not

- REST is not a standard
 - ❑ You will not see the W3C putting out a REST specification.
 - ❑ You will not see IBM or Microsoft or Sun selling a REST developer's toolkit.
- REST is just a **design pattern**
 - ❑ You can't bottle up a pattern.
 - ❑ You can only understand it and design your Web services to it.
- REST does prescribe the **use** of standards:
 - ❑ HTTP (transfer state)
 - ❑ URL (resource identifier)
 - ❑ XML/HTML/GIF/JPEG/*etc.* (resource representations)
 - ❑ text/xml, text/html, image/gif *etc.* (resource types)

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

What Rest is – Roy Fielding

- An **architecture style** is a coordinated set of architectural constraints that restricts the roles and features of architectural elements, and the allowed relationships between those elements, within any architecture that conforms to that style.
 - ❑ A style can be applied to many architectures
 - ❑ An architecture can consist of many styles

What is REST?

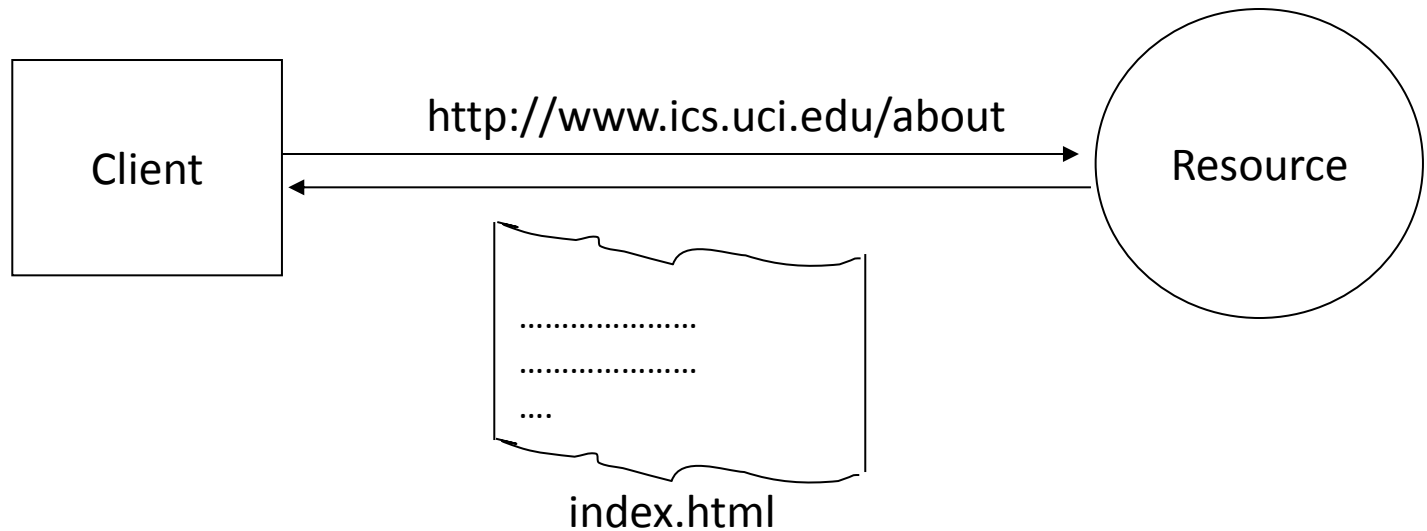
Why REST?

Where is REST used?

JSON

Conclusion

Why is it called "Representational State Transfer"?



1. The Client ask for a Web resource using a URL.
2. A **representation** of the resource is returned.
3. The representation places the client application in a **state**.
4. The client **transfers** (update) its **state** upon reception of the representation.

--> Representation State Transfer!

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

Where is REST??

- Web is an example of a REST system!
- HTTP is the most common example of REST based system.
- All of those Web services that you have been using all these many years - book ordering services, search services, online dictionary services, etc - are REST-based Web services.
- Blogosphere, Atom Publishing Protocol
- You can also create your own!

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

Characteristics of a REST-based Network

- **Client-Server:** a pull-based interaction style: consuming components pull representations.
- **Stateless:** each request from client to server must contain all the information necessary to understand the request, and cannot take advantage of any stored context on the server.
- **Cache:** to improve network efficiency responses must be capable of being labeled as cacheable or non-cacheable.
- **Uniform interface:** all resources are accessed with a generic interface (e.g., HTTP GET, POST, PUT, DELETE).
- **Named resources** - the system is comprised of *resources* which are *named* using a URI.
- **Interconnected resource representations** - the representations of the resources are interconnected using URLs, thereby enabling a client to progress from one state to another.

What is REST?


Why REST?

Where is REST used?

JSON

Conclusion

Uniform operations in REST

- In common parlance these are CRUD – Create, Read, Update and Delete 
- For HTTP these are: POST, GET, PUT, DELETE
 - ❑ Simplifies semantics
 - ❑ Simplifies client complexity
 - ❑ Simplifies application model

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

Retrieving Information: HTTP GET



- The user types in at his browser: `http://www.amazon.com`
- The browser software creates an HTTP header (no payload)
 - The HTTP header identifies:
 - The desired action: GET ("get me resource")
 - The target machine (`www.amazon.com`)

What is REST?

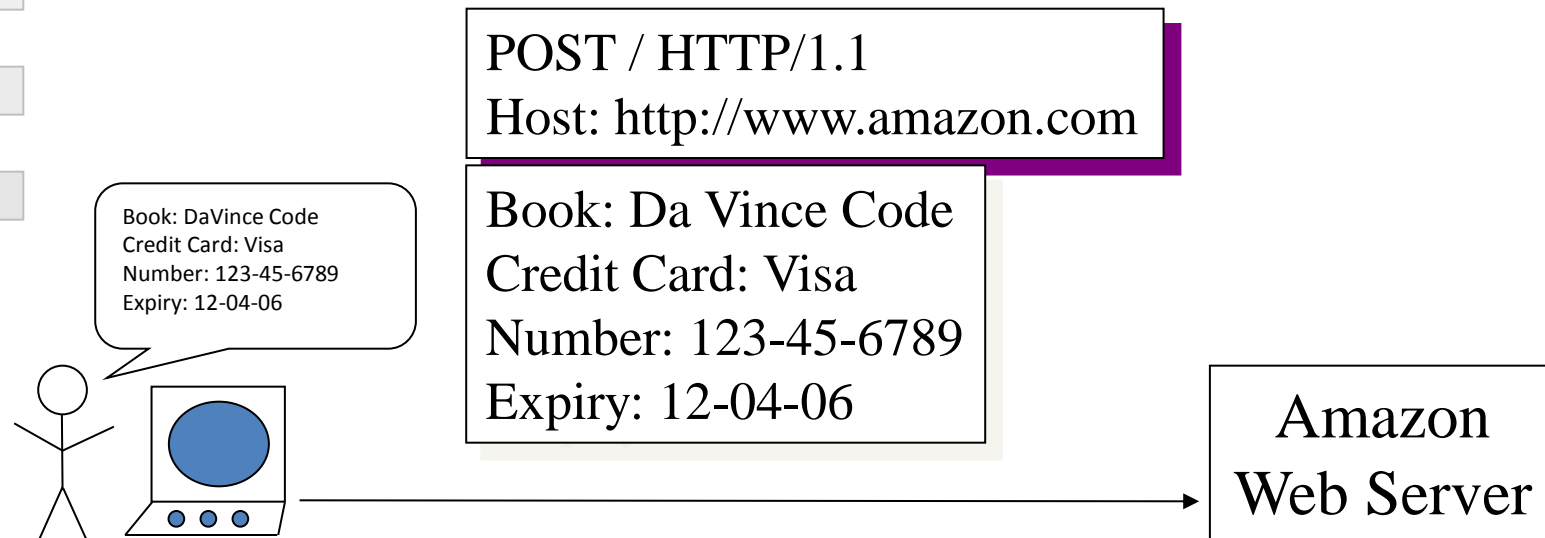
Why REST?

Where is REST used?

JSON

Conclusion

Updating information: HTTP POST



- The user fills in the Web page's form
- The browser software creates an HTTP header with a payload comprised of the form data
 - The HTTP header identifies:
 - The desired action: POST ("here's some update info")
 - The target machine (amazon.com)
 - The payload contains:
 - The data being POSTed (the form data)

What is REST?

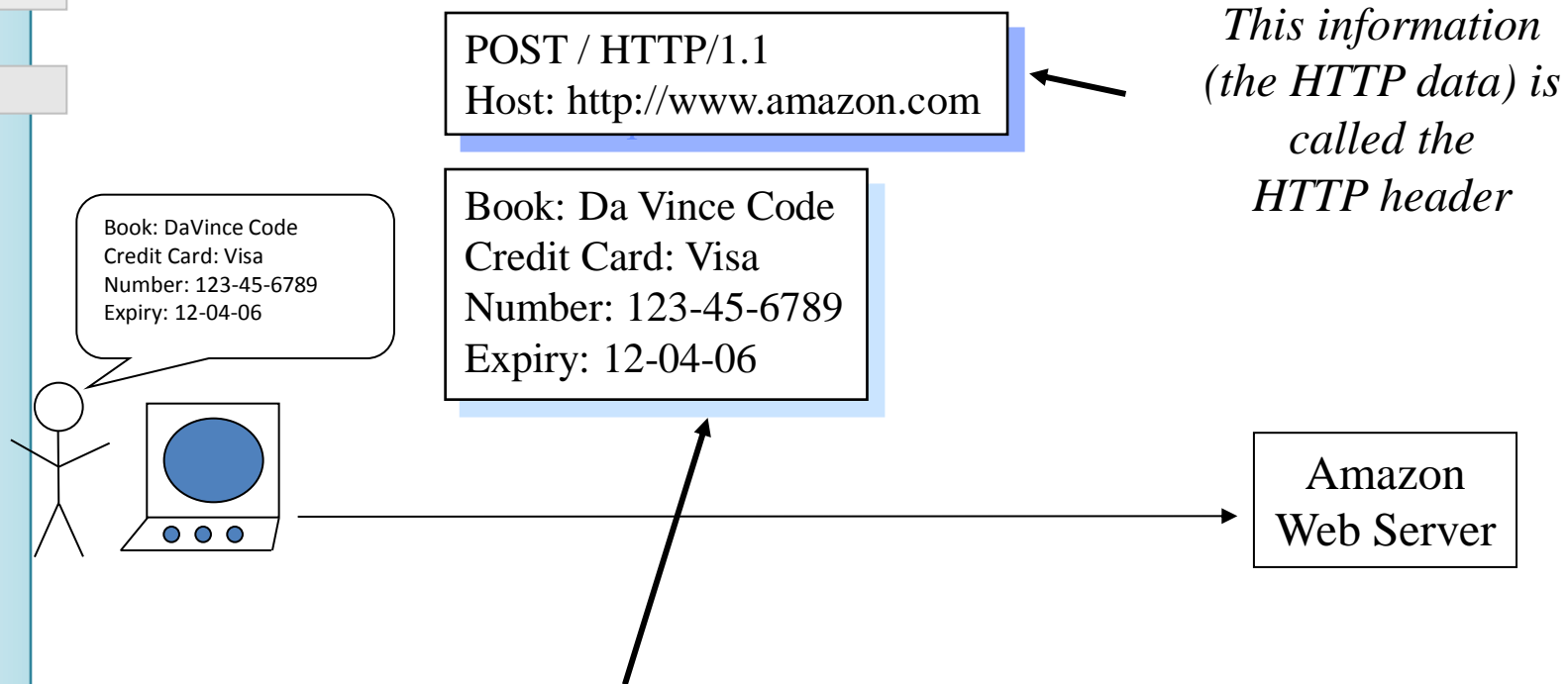
Why REST?

Where is REST used?

JSON

Conclusion

Header and payload



What is REST?

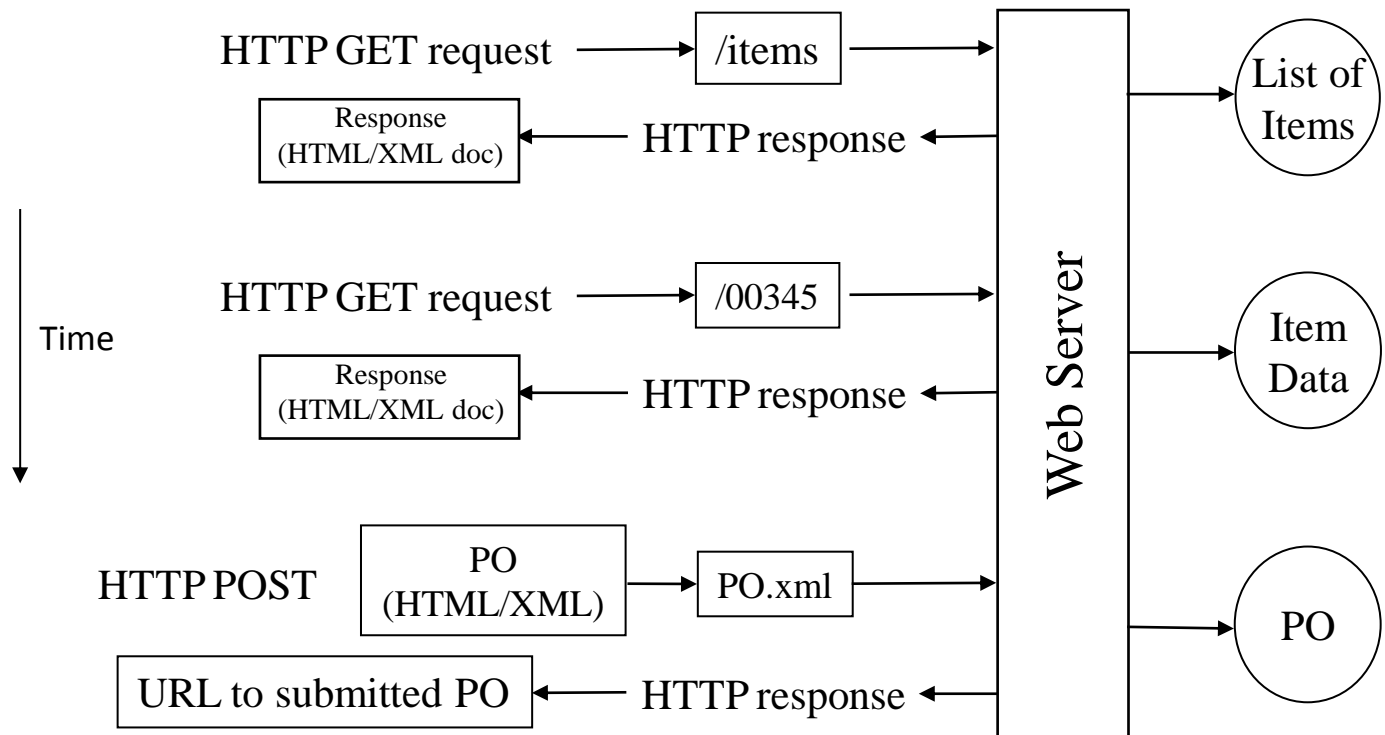
Why REST?

Where is REST used?

JSON

Conclusion

User Session using REST



What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

When to REST? And when not to..

- Easier integration
- Easier for humans to understand
- Hypermedia systems like Web
- ✗ Efficiency
- ✗ No Hypermedia content
- ✗ Integrating non RESTful stuff
- ✗ Challenges –Security, getting the URI's right etc.

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

Conclusion

- REST is simple and easy to use
- Designed alongside the HTTP specifications to support the Web
- Has issues related to security but is still preferred over other architectures

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

JSON

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

JavaScript Object Notation

- **What is JSON?**

- ❑ JSON is lightweight text-data interchange format
- ❑ JSON is language independent*
- ❑ JSON is "self-describing" and easy to understand
- ❑ JSON uses JavaScript syntax for describing data objects, but JSON is still language and platform independent. JSON parsers and JSON libraries exists for many different programming languages.

- **JSON - Evaluates to JavaScript Objects**

- ❑ The JSON text format is syntactically identical to the code for creating JavaScript objects.
- ❑ Because of this similarity, instead of using a parser, a JavaScript program can use the built-in eval() function and execute JSON data to produce native JavaScript objects.

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

When to use JSON?

- SOAP is a protocol specification for exchanging structured information in the implementation of Web Services.
- SOAP internally uses XML to send data back and forth.
- REST is a design concept.
- You are not limited to picking XML to represent data, you could pick anything really (JSON included).

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

JSON example

```
• {  
•   "firstName": "John",  
•   "lastName": "Smith",  
•   "age": 25,  
•   "address": {  
•       "streetAddress": "21 2nd Street",  
•       "city": "New York",  
•       "state": "NY",  
•       "postalCode": 10021  
•   },  
•   "phoneNumbers": [  
•       {  
•           "type": "home",  
•           "number": "212 555-1234"  
•       },  
•       {  
•           "type": "fax",  
•           "number": "646 555-4567"  
•       }  
•   ]  
• }
```

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

JSON to XML

```
• <?xml version="1.0" encoding="UTF-8"?>
• <persons>
•   <person>
•     <firstName>John</firstName>
•     <lastName>Smith</lastName>
•     <age>25</age>
•     <address>
•       <streetAddress>21 2nd Street</streetAddress>
•       <city>New York</city>
•       <state>NY</state>
•       <postalCode>10021</postalCode>
•     </address>
•     <phoneNumbers>
•       <phoneNumber>
•         <number>212 555-1234</number>
•         <type>home</type>
•       </phoneNumber>
•       <phoneNumber>
•         <number>646 555-4567</number>
•         <type>fax</type>
•       </phoneNumber>
•     </phoneNumbers>
•   </person>
• </persons>
```

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

JSON vs XML size

- XML: 549 characters, 549 bytes
- JSON: 326 characters, 326 bytes
- XML ~68,4 % larger than JSON!
- But large data set is going to be large regardless of the data format you use.
- Most servers gzip or otherwise compress content before sending it out, the difference between gzipped JSON and gzipped XML isn't nearly as drastic as the difference between standard JSON and XML.

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

JSON Values

- JSON values can be:
 - A number (integer or floating point)
 - A string (in double quotes)
 - A boolean (true or false)
 - An array (in square brackets)
 - An object (in curly brackets)
 - null

What is REST?

Why REST?

Where is REST used?

JSON

Conclusion

Questions?

- ❑ A little REST and Relaxation : presentation by Roy Fielding
- ❑ REST and the Real World
 - <http://www.xml.com/pub/a/2002/02/20/rest.html>
- ❑ <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>
- ❑ Second Generation Web Services
 - <http://www.xml.com/pub/a/2002/02/06/rest.html>
- ❑ The REST architectural style : presentation by Robert Wilson