REST Web Services

Majrul Ansari
Java & Web Technology Trainer

My Introduction

Would love to go on Mars if I get an opportunity! ☺

What are we going to learn?

- How Java can be used for developing RESTful Web Services
- Different technologies that contribute in the development process
  - HTTP, XML, JSON, …
- JAX-RS, the standard API
  - Handling Request
  - Generating Response
  - Support for XML & JSON
  - Content Negotiation
  - Client API
  - Exception Handling
  - Support for customisation

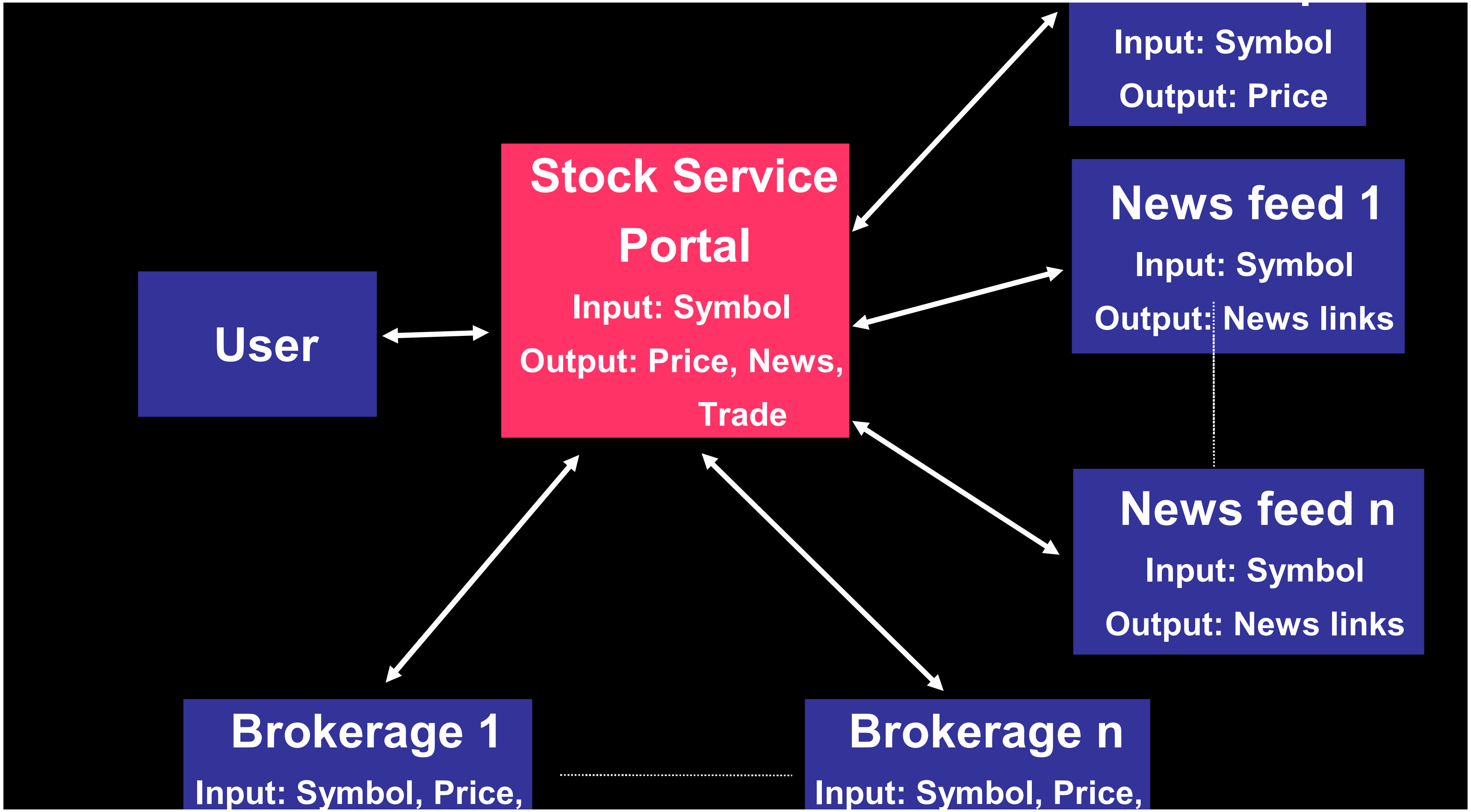
Introduction

Web Services are self-contained, self-describing, modular applications that can be published, located, and invoked over a network--generally, the Web.

Another definition

- A Web service is a software application identified by a URI whose interfaces and binding are capable of being defined, described and discovered by XML artifacts and supports direct interactions with other software applications using XML based messages via Internet-based protocols

Overview



Input: Symbol

Output: Price

Stock Service Portal

Input: Symbol

Output: Price, News, Trade

News feed 1

Input: Symbol

Output: News links

User

News feed n

Input: Symbol

Output: News links

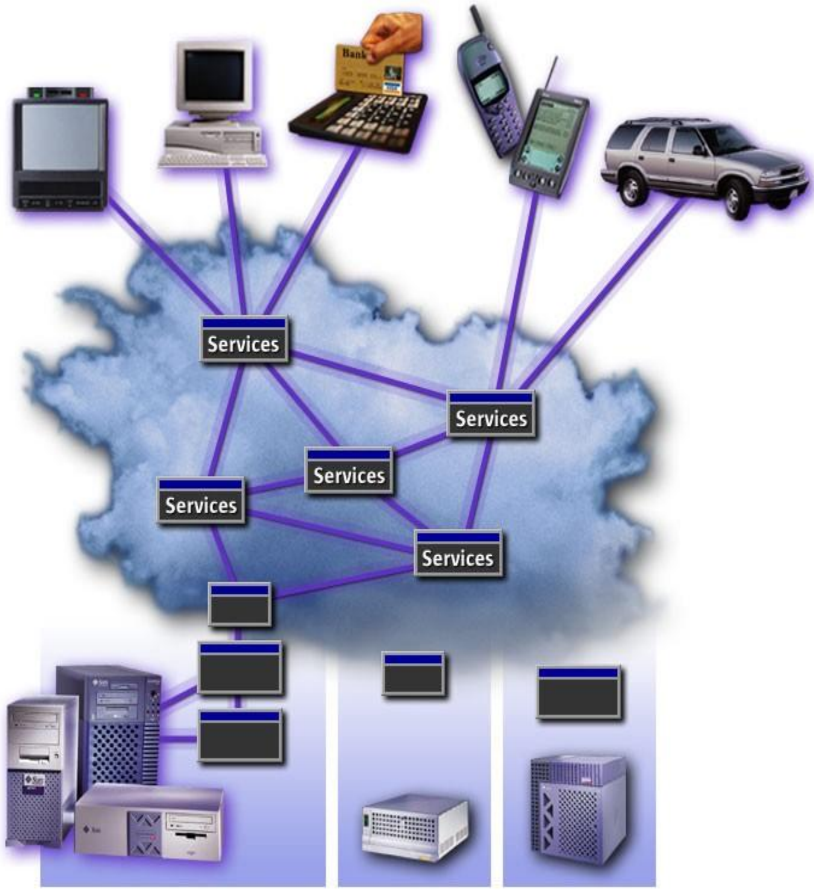
Brokerage 1

Input: Symbol, Price,

Brokerage n

Input: Symbol, Price,

Why WebServices?



- □ Platform neutral
- □ Accessible using standards and are Interoperable
- □ Simplifies enterprise integration

Role of XML & JSON

- □ XML & JSON is popularly used for representing data transferred over the network between the Service provider and the Consumer
- □ Different APIs are used in Java to handle XML & JSON parsing

XML Support in Java

- JAXP (Java XML Parsing API)
    - A thin and lightweight API for parsing and transforming XML documents
- Allows for pluggable parsers and transformers
- Supports parsing of XML using:
    - SAX (Event driven)
    - DOM (Tree based)
    - StAX (Pull based)

About JAXB

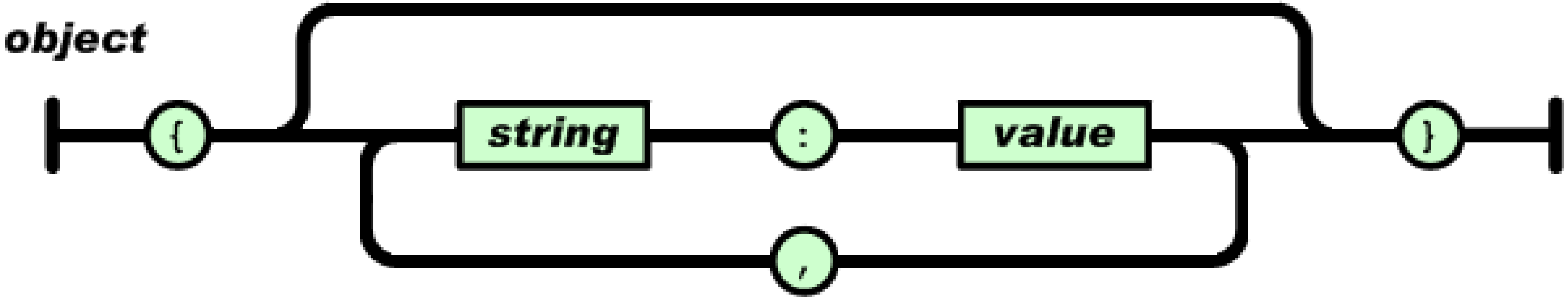
- Provides API, tools, and a framework that automate the mapping between XML documents and Java objects
  - Provides compiler that compiles XML schema to Java classes
- Provides an efficient and standard way of mapping between XML and Java code
  - Programmers don't have to create application specific Java classes anymore themselves
- Programmers do not have to deal with XML structure, instead deal with meaningful business data
  - *getData()* method as opposed to *getAttributes()*

Cont’d…

- In a sense JAXB is high-level language while JAXP/SAX/DOM are assembly language for XML document management
- JAXB automates XML to Java binding so you can easily access your data
  - So that means we don’t have to use any parser explicitly in our code

Support for JSON in Java

- A lightweight simple key:value pair based datastructure
  - Alternative to XML
- Before Java EE 7, we didn't had any standard API for JSON parsing in Java



JSON Parsing

- Java EE 7 is the first version to introduce a standard API for parsing JSON called as JSONP
- Alternatively we have been using third party JSON parsing APIs
  - Jackson (Widely used)
  - gson
  - json-simple
  - …
- JSONB is available from Java EE 8 onwards

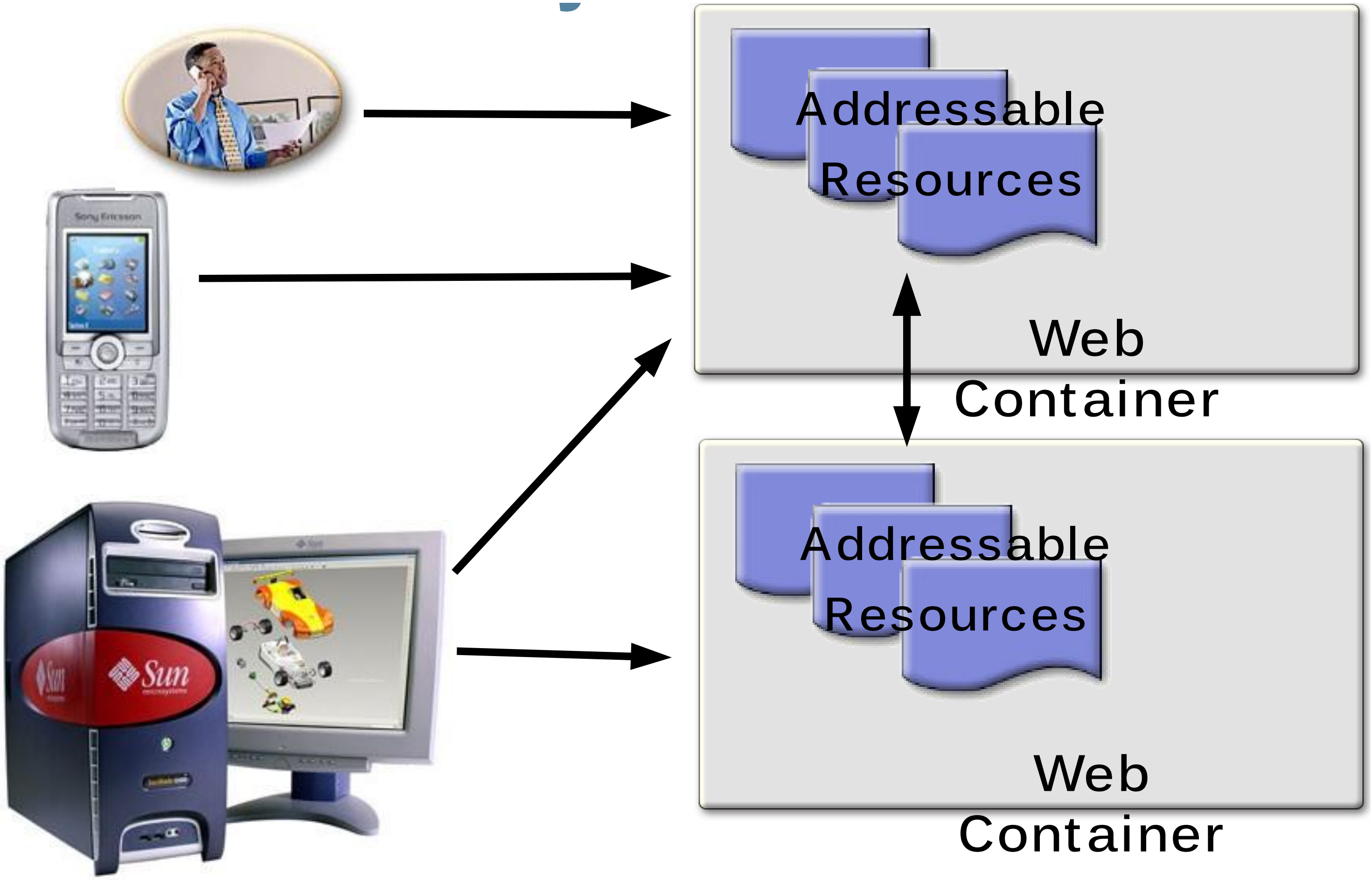
Representational State Transfer

Time to take some REST

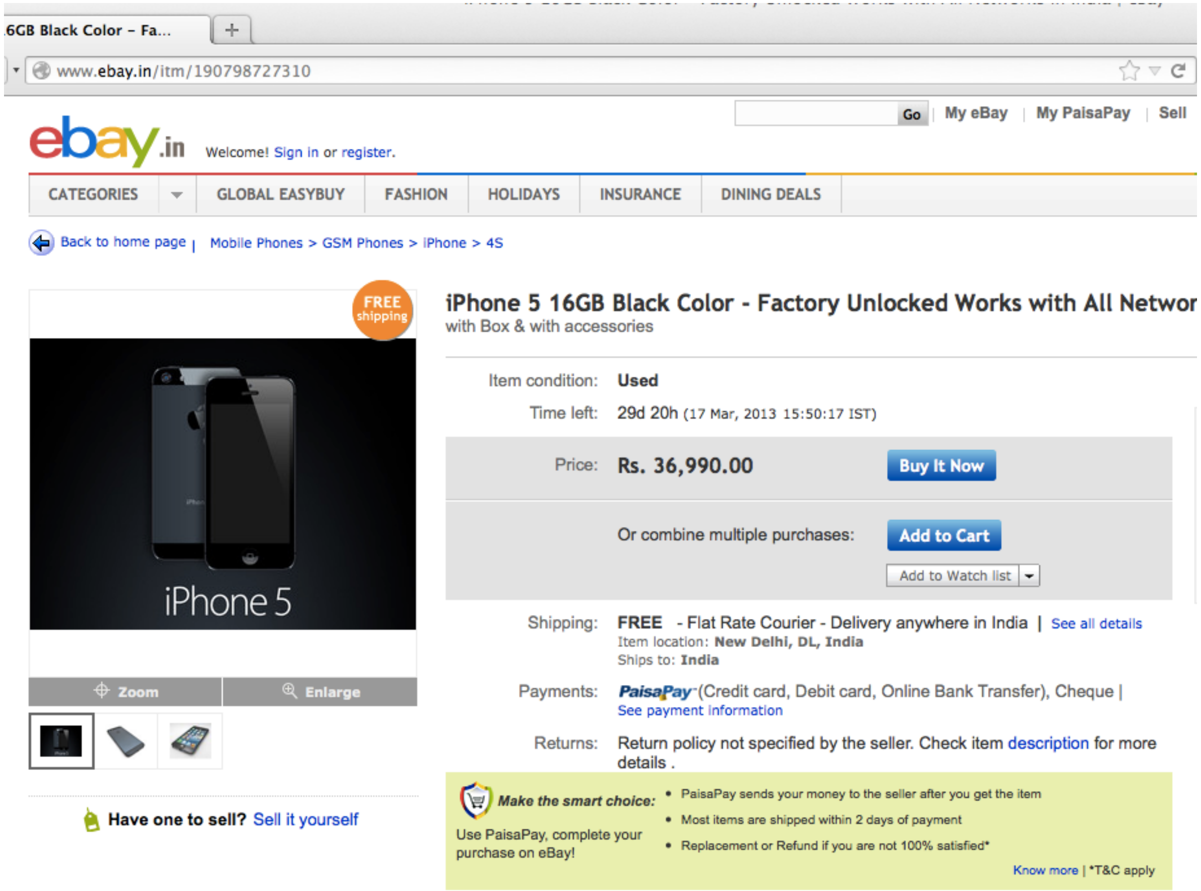
But why REST?

Like SOAP, why not SHAMPOO, CONDITIONER, … ☺

The Web Wide World



Everything is an URL



Welcome! Sign in or register.

Back to home page | Mobile Phones > GSM Phones > iPhone > 4S

iPhone 5 16GB Black Color - Factory Unlocked Works with All Networ
with Box & with accessories

Item condition:	Used
Time left:	29d 20h (17 Mar, 2013 15:50:17 IST)
Price:	Rs. 36,990.00
	Or combine multiple purchases:
Shipping:	FREE - Flat Rate Courier - Delivery anywhere in India | See all details Item location: New Delhi, DL, India Ships to: India
Payments:	PaisaPay (Credit card, Debit card, Online Bank Transfer), Cheque | See payment information
Returns:	Return policy not specified by the seller. Check item description for more details .

Make the smart choice:

Use PaisaPay, complete your purchase on eBay!

- PaisaPay sends your money to the seller after you get the item
- Most items are shipped within 2 days of payment
- Replacement or Refund if you are not 100% satisfied*

Know more | *T&C apply

Have one to sell? Sell it yourself

REST

- Representational State Transfer it is
- RESTful services are stateless
- RESTful services have a uniform interface
- REST-based architectures are built from resources (pieces of information) that are uniquely identified by URIs
- REST is considered as the rebirth of HTTP, in a way somewhat similar to the 1995 Karan Arjun movie in Bollywood ;-)
- The guy behind REST is **Roy Fielding**

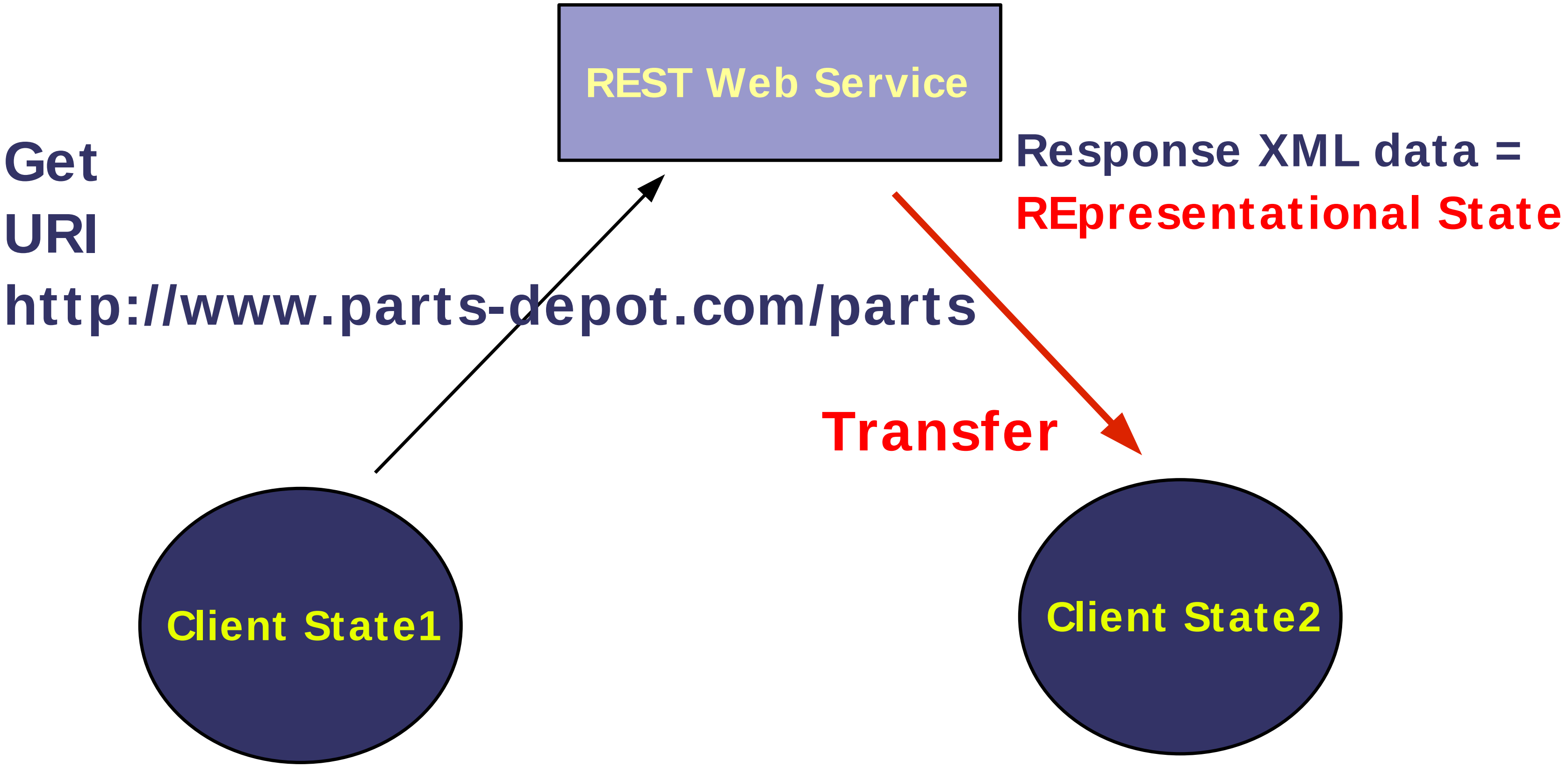
Cont’ d…

- REST is an architectural principle applied for developing distributed applications. RPC, RMI, CORBA, DCOM, SOAP/WS-* and now REST very much existed/exist for this very specific requirement
- The architectural principles defined are as follows:
  - Addressable resources
  - A uniform, constrained interface
    - GET, POST, PUT, and DELETE
  - Representation oriented
  - Communicate statelessly
    - Each request from client to server must contain all the information necessary to understand the requesT
  - HATEOAS (Hypermedia As the Engine Of Application State)

REST

- In REST system, resources are manipulated through the exchange of “representations” of the resources
  - For example, a purchase order resource is represented by an XML or JSON document
  - In a RESTful purchasing system, each purchase order is made through a combination of HTTP POST method with XML document, which represents the order, sent to a unique URI

Overview



REST Web Service

Get

URI

http://www.parts-depot.com/parts

Response XML data =

REpresentational State

Transfer

Client State1

Client State2

ID is a URI

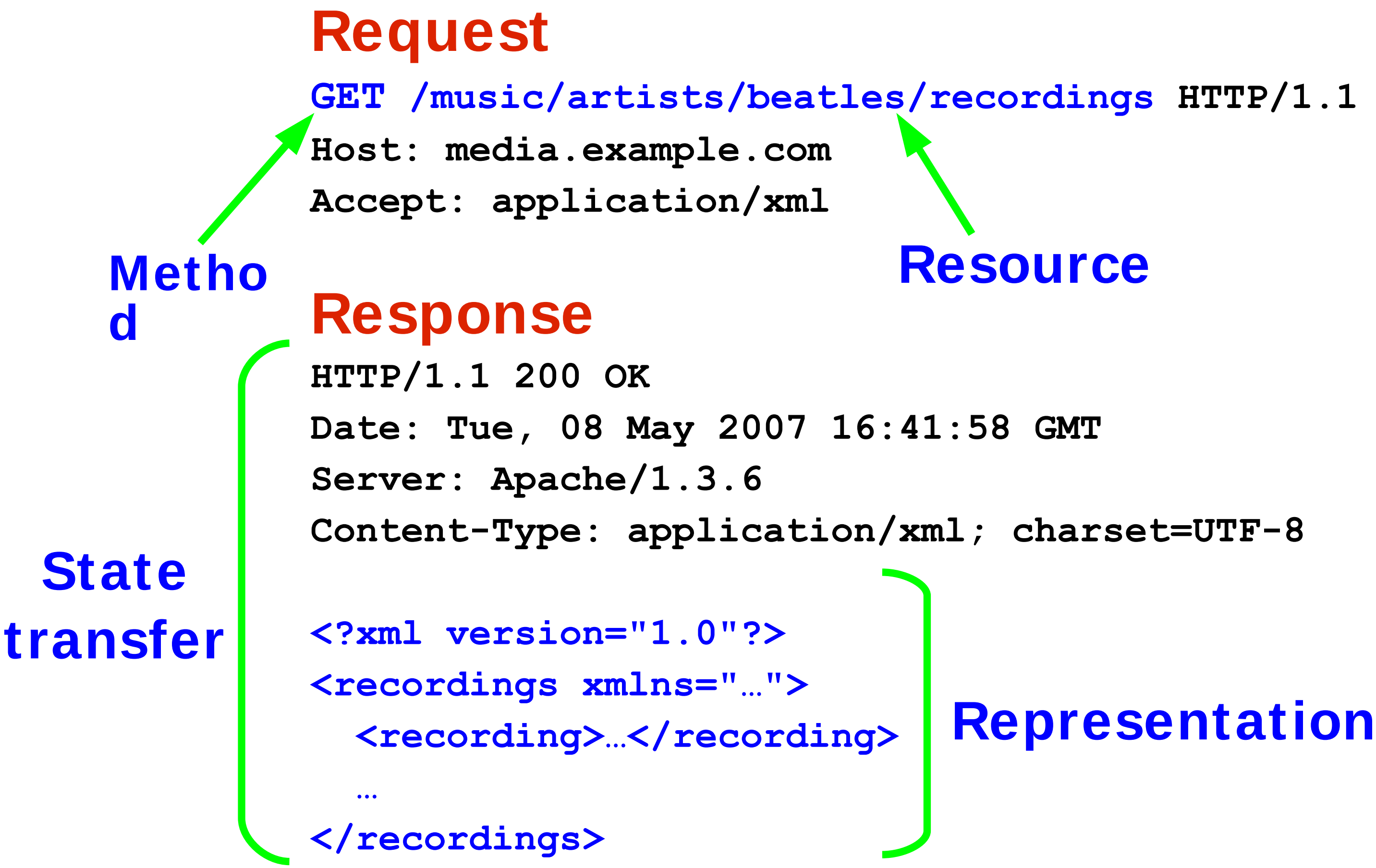
- Every Resource has an ID

http://example.com/widgets/foo
http://example.com/customers/bar
http://example.com/customers/bar/orders/2
http://example.com/orders/101230/customer

Back to HTTP

- Offer data in a variety of formats
  - XML
  - JSON
  - XHTML
- Support content negotiation
  - Accept header
    - GET /foo
    - Accept: application/json
  - URI based
    - GET /foo.json

RESTing



Request

```
GET /music/artists/beatles/recordings HTTP/1.1
Host: media.example.com
Accept: application/xml
```

Method

Resource

Response

```
HTTP/1.1 200 OK
Date: Tue, 08 May 2007 16:41:58 GMT
Server: Apache/1.3.6
Content-Type: application/xml; charset=UTF-8

<?xml version="1.0"?>
<recordings xmlns="…">
  <recording>…</recording>
  …
</recordings>
```

State transfer

Representation

CRUD HTTP

CRUD Operations	4 main HTTP methods Verb	 Noun
Create (Single)	POST	Collection URI
Read (Multiple)	GET	Collection URI
Read (Single)	GET	Entry URI
Update (Single)	PUT	Entry URI
Delete (Single)	DELETE	Entry URI

GET

- GET to retrieve information
  - GET /music/beatles
- Cacheable

POST

- POST to add new information
  - POST /music/beatles

PUT

- PUT to update information
  - PUT /songs/beetles/123-567890

DELETE

- Remove the data
  - DELETE /songs/beatles/idontknow

REST vs SOAP

- "Traditional" SOAP-based web service
  - Few URIs (nouns), many custom methods (verbs)
    - **musicPort.getRecordings("beatles")**
  - Uses HTTP as transport for SOAP messages
- RESTful web service
  - Many resources (nouns), few fixed methods(verbs)
    - **GET /music/artists/beatles/recordings**
  - HTTP is the protocol

SOAP vs REST

- SOAP based web services is about services SOA
  - Stock quote service
  - quoteService.**purchase**(“oracle”, 2013, 6.0f);
- REST is Resource-Oriented Architecture (ROA)
  - Stock quote resource
  - Resources are manipulated by exchanging representations
  - POST /stocks/quotes/oracle

JAX-RS Design Goals

- Support REST concepts
  - Everything is a resource
  - Every resource is address'able via URI
  - HTTP methods provides uniform interface > Representations (formats)
- Support High level and Declarative programming model
  - Use @ annotation in POJOs
- Generate or hide the boilerplate code
  - No need to write boilerplate code for every app

Implementations of JAX-RS

- Jersey – reference implementation of JAX-RS
  - Comes with Glassfish, other Java EE 6 servers
- Other open source implementations of JAX-RS
  - Apache CXF
  - JBoss RESTEasy
  - Restlet

Development Tools

- IDE – for general purpose RESTful Web service development
  - NetBeans, Eclipse, Intellij IDEA
- Client tools – for sending HTTP requests
  - RESTClient
  - Lot’s of plugins available for browsers
- Several command line tools
  - curl
- soapUI

SOA

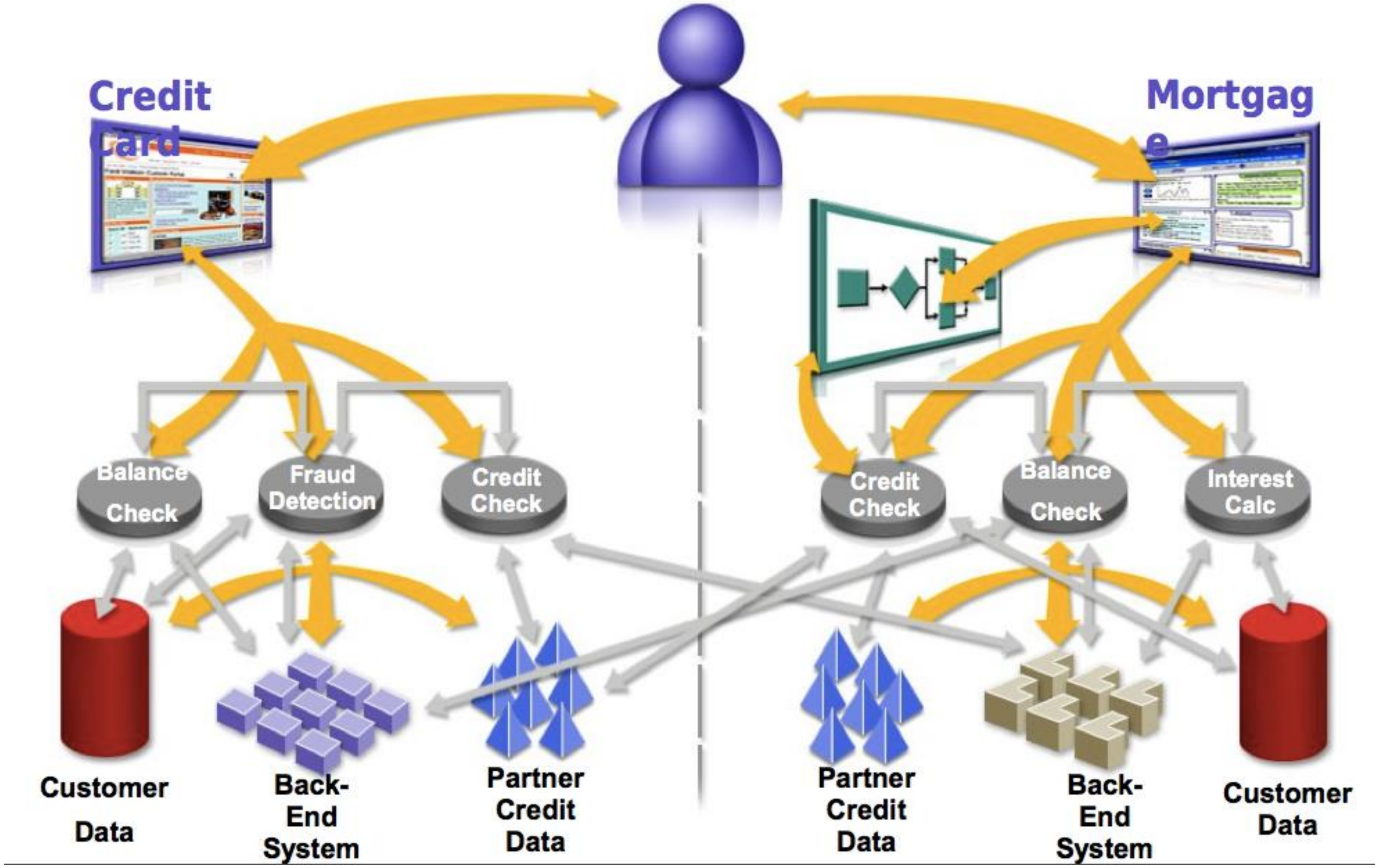
Service Oriented Architecture

- An architectural principle for structuring systems into coarse-grained *services*
- Technology-neutral best practice
- Emphasizes the loose coupling of services
- New services are created from existing ones in a synergistic fashion
- Strong service definitions are critical
- Services can be re-composed when business requirements change

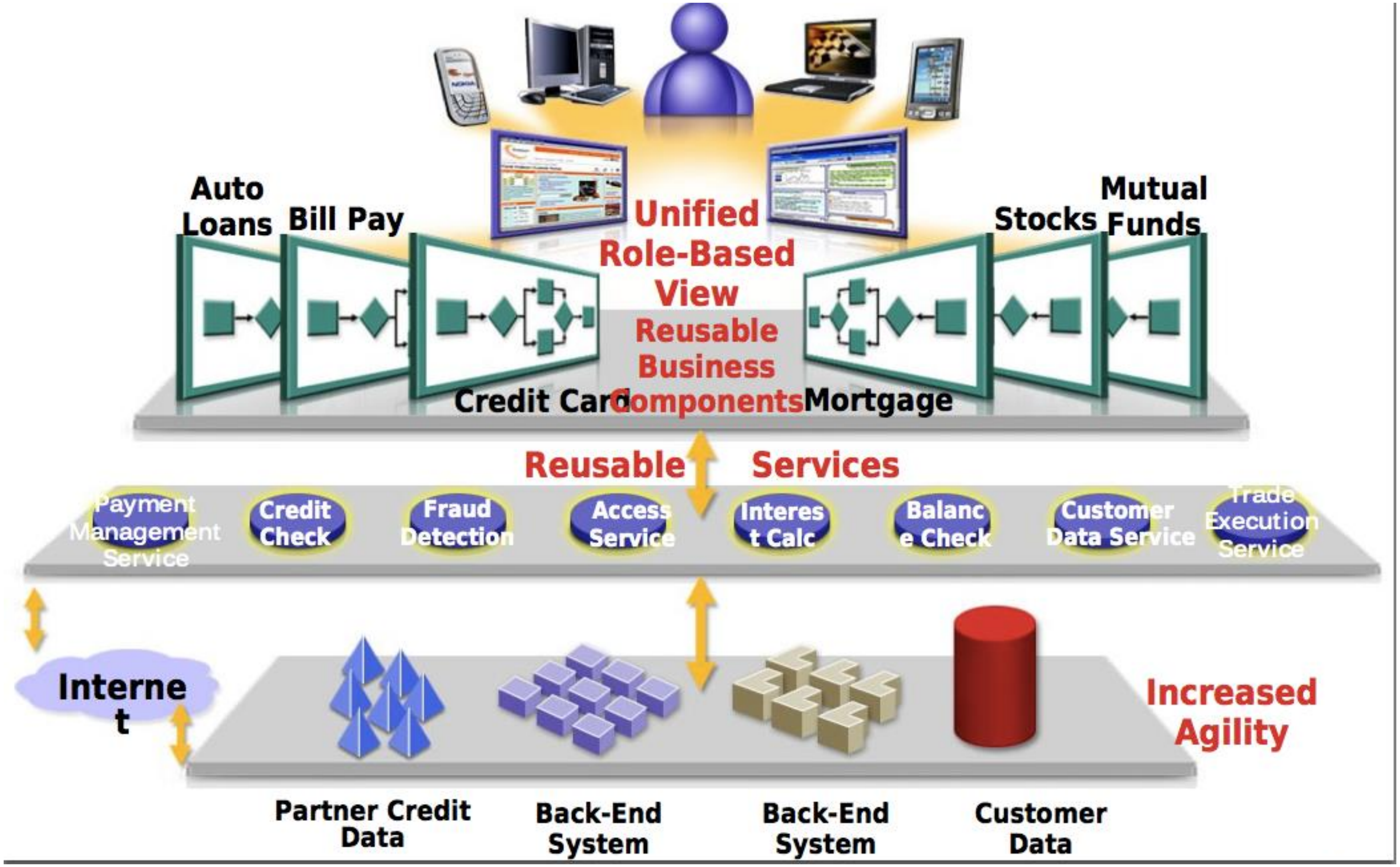
The right approach

- Developers need to build end-to-end applications
  - Front-end user interface
  - Middle-tier business logic
  - Back-end resources
- With the right approach, developers can…
  - Reuse existing parts
  - Build new parts
  - Glue old and new parts together
- With the wrong approach, developers must…
  - Re-implement functionality existing elsewhere
  - Spend massive effort to evolve applications

Before SOA


Credit Card
Mortgag e
Balance Check
Fraud Detection
Credit Check
Credit Check
Balance Check
Interest Calc
Customer Data
Back-End System
Partner Credit Data
Partner Credit Data
Back-End System
Customer Data

SOA Enabled


Auto Loans
Bill Pay
Unified Role-Based View
Reusable Business Components
Stocks
Mutual Funds
Credit Card
Mortgage
Reusable
Services
Payment Management Service
Credit Check
Fraud Detection
Access Service
Interest Calc
Balance Check
Customer Data Service
Trade Execution Service
Internet
Increased Agility
Partner Credit Data
Back-End System
Back-End System
Customer Data

Thank You

Majrul Ansari

contactme@majrul.com