Web Services & Service-Oriented Programming (SOP)

Service-Oriented Programming (SOP)

- SOP
 - A programming paradigm that uses "services" as the building block to develop applications
- Approach
 - Develop services (web)
 - Make use of services by invoking (calling) others
 - Combine services to make additional services

What is a Web Service?

- A service (high-level description)
 - Similar to a method -> a program calls a method and gets some results back
- A web service (high-level description)
 - A service that can be invoked by a program via the internet.
- A web service is different from a web application
 - A web *application* is for use by humans
 - Such as http://www.weather.com
 - A web *service* is for use by programs
 - Such as Twitter APIs: <u>https://dev.twitter.com/rest/reference/get/followers/ids</u>

What is a Web Service?



What is a Web Service?

- Defined by World Wide Web Consortium (W3C):
 - A software system designed to support interoperable machine-to-machine interaction over a network.
 - It has an interface described in a machine-processable format (specifically WSDL).
 - Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

Examples of Web Services

- A self-contained business task
 - A money withdrawal or funds deposit service for a bank
- A full-fledged business process with multiple tasks
 - Automated purchasing of office supplies with approvals at different levels
- An application
 - A complete life insurance application
- A service-enabled resource
 - Access to a remote database containing patient medical records

What Web Services Bring to SOP

- Traditional OOP constrained to *homogeneity*
 - Same data types, programming languages, development platform, operating systems
- Web services handle *heterogeneity*
 - Makes extensive use of XML because XML has become ubiquitous
 - Accessibility through functionalities available on the web (standard networking and http)
- SOP

- Builds on web services to support *software reuse*

Consuming a Web Service

- The process of invoking a web service is known as *consuming* a web service.
 - Option 1 : The consuming program (client) sends a
 SOAP message to the web service. Both consumer and service use the same WSDL interface that describes the services provided. Contents of the sent/received SOAP message use XML format.
 - Option 2 (*RESTful* service) : The consuming program sends a *HTTP* request operation (GET, PUT, POST, DELETE) based on the service to be invoked. The web service replies with an HTTP response.

SOAP based Web Service

Web Services **publish** their location and services (WSDL Interfaces) in a registry (UDDI). Clients **consume** services using the same WSDL Interface via a SOAP message. Web Services can act as both publishers and consumers.

Consumer needs to create a *remote proxy* which implements the WSDL Interface *and* handles all the responsibilities for sending and receiving SOAP messages.



Consuming a RESTful Web Service

- In a *Representation State Transfer* (REST) style architecture requests and responses are built around the transfer of representations of *resources*.
- REST recognizes everything as a resource and each resource implements a standard uniform interface (typically HTTP interface).
- Resources have names and addresses (URLs)
- Each resource has one or more representation (like JSON or XML) and resource representations move across the network usually over HTTP.

REST(ful) based Web Service



Consuming a RESTful Web Service

- All interactions between a client and a web service are done with simple operations. Most web interactions are done using HTTP and just four operations:
 - retrieve information (HTTP GET)
 - create information (HTTP PUT)
 - update information (HTTP POST)
 - delete information (HTTP DELETE)

Consuming a RESTful Web Service

- To invoke a RESTful API all you need is a browser or HTTP stack and pretty much every device or machine connected to a network has that.
- Example of consuming Yahoo's Finance API using Java's HTTP centric classes:
 - <u>https://gist.github.com/kocsenc/fd7febfda2f6eb8</u>
 <u>dffb4</u>