CICS & SOA

What you don’t know might help you
Will Yates – CICS Test Architect
SOA is not just web services

A service-oriented architecture (SOA) is an architectural pattern in computer software design in which application components provide services to other components via a communications protocol, typically over a network. The principles of service-orientation are independent of any vendor, product or technology.
CICS essential design concepts

- Pseudo – conversational programming
- Communication Area – COMMAREA or Channels & Containers
- Program to program communication
Two models of CICS SOA Integration

CICS TS

Terminal

Presentation Logic

Business Logic

Data Access Logic

Service end-point

Service client

Service end-point

Integration logic
Two models of CICS SOA Integration
SOA Story: Enterprise SOA

SOA
WS-AT
WSRR
WS-DL
WS-Sec
SOAP
High Ceremony
Service Description
Service Governance

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
SOA Story: Restful SOA

Situational Apps
Mobile Enablement
Resource Orientated

Service Description
Service Governance

REST
HTTP
ATOM

PHP

SOAP
WS-Sec
WSRR
WSDL
WS-AT
High Ceremony

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
SOA Story: CICS TS v3 era

Service Description
Service Governance
CICS TS v3

WS-AT
WSRR
WSDL
WS-Sec
SOAP

High Ceremony

Situational Apps
Mobile Enablement
Resource Orientated

REST
HTTP
PHP
ATOM
SOA Story: CICS TS v4 era

CICS TS V4.2
Feature Pack for Mobile Extensions

REST
HTTP
PHP
ATOM

SOAP
WS-Sec
WSRR
WSDL
WS-AT
High Ceremony

Situational Apps
Mobile Enablement
Resource Orientated

Service Description
Service Governance

CICS TS v4
SOA Story: CICS v5 era

• Language mapping enhancements to WSBIND

• New Base Function
  – Mobile Extensions Feature Pack

• New enablement function
  – Websphere Liberty Profile embedded into CICS runtime
    • Web enabled servlets
    • JAX-RS
    • JAX-WS
Mobile: Another stage in computing history

Mobile is different:
- Ubiquitous computing
- Mixed models
- A different type of app
- Revolutionary to business models

Host / Mainframe

Client / Server
Web / Desktop

Mobile / wireless / cloud

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Mobile internet usage (UK 2014)

68% of adults used the internet ‘on the go’

- 58% used a mobile phone or smartphone
- 43% used a portable computer
- 15% used an Ebook reader or other handheld device

Source: UK Office for National Statistics
Worldwide 4G network coverage

64%

Of the world's population will be covered by 4G-LTE networks by the end of 2020

Source: GSMA Intelligence
OK I get it mobile is huge! But how does that affect which SOA technology I pick? What choices do I have?
SOA pick and mix

Language Protocol

- JSON
- SOAP
- XML
- HTML

Service invocation

- REST
- WS - *

Transport

- HTTP
- HTTPS
- Message Queue
- CTG

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
SOA pick and mix

Language Protocol

- JSON
- SOAP
- XML
- HTML

Service invocation

- REST
- WS - *

Transport

- HTTP
- HTTPS
- Message Queue
- CTG

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
SOA pick and mix

Language Protocol

- JSON
- SOAP
- XML
- HTML
- REST
- WS - *

Service invocation

- HTTP
- HTTPS
- Message Queue
- CTG

Transport

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
SOA pick and mix

Language Protocol

- JSON
- SOAP
- XML
- HTML
- REST
- WS

Service invocation

- HTTP
- HTTPS
- Message Queue
- CTG

Transport

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Language formats

- heavyweight, human readable, text based format for data
- More complex with custom types
- Requires dedicated parser

{ "JSON": "REST" }

- Lightweight, human readable, text based format for data
- Simple Structure
- Native JavaScript support
- Widely adopted by the industry

<SOAP><xml/></SOAP>
Not just JSON or SOAP

• Neither technology demands a particular data format
  – Raw XML
  – HTML
  – Binary Data
    • PDF etc

• HTTP media type can inform the client of the response data

• What can your client consume – who is the customer?
SOA pick and mix

Language Protocol

- JSON
- SOAP
- XML
- HTML

Service invocation

- REST
- WS - *

Transport

- HTTP
- HTTPS
- Message Queue
- CTG

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Service invocation (REST or WS-*)

- The key motivator of a RESTful SOA is simplicity and ubiquity
  - Deliver content in the simplest possible way
  - HTTP is available everywhere; it’s like the air around us

- With Enterprise SOA (WS-*), it’s not the body (user data) that distinguishes the QoS, it’s the richness of the WS-* headers
  - WS-Security is about choice in the decision of encryption, identity tokens and digital signatures
  - WS-Addressing is about transport-neutral mechanisms of describing addresses
  - Even the WS-I standards are about allowing maximum flexibility within a fully agreed-upon framework of standards

- You might want to choose WS-* where you don’t have overall control of connectivity.
  - WS-ReliableMessaging and SOAP over JMS are about choice in how you obtain qualities of service
REST: Accessibility for Developers

- A RESTful Web service is formed like a sentence – it simplifies how developers access services
  - **Verb** = HTTP Action (GET, POST, PUT, DELETE)
  - **Noun** = the URI of the Service (the document)
  - **Adjective** = MIME type of the resulting document

```
GET
PUT
POST
DELETE
```

verb

noun

adjective

http://server.com/users/100210

json
REST: Accessibility for Developers

- A RESTful Web service is formed like a sentence – it simplifies how developers access services
  - **Verb** = HTTP Action (GET, POST, PUT, DELETE)
  - **Noun** = the URI of the Service (the document)
  - **Adjective** = MIME type of the resulting document

GET http://server.com/

```
<operation>
  <function>getCustomerDetails</function>
  <customerNumber>100210</customerNumber>
</operation>
```
JSON vs SOAP/XML web services

• JSON is lightweight?
  – Complexity of message structures
    • A complex message in JSON representation will be more costly to process than a simple message in XML representation
    • HTTP(S) connectivity base-line cost is equivalent

JavaScript and JSON work together well by design out of the box
XML is very powerful but needs requires a dedicated parser

• Versioning of a service can be costly with SOAP/XML
  – Rolling out changes to back-level clients can be costly
    • Regenerate and deploy WSDL
• JSON schemas are more forgiving e.g. default values fields
Merging Enterprise SOA & RESTful SOA

• Developing for both Enterprise SOA and RESTful SOA enables the exploitation of distinct content pools:
  – Services generated inside your enterprise
  – Services generated outside the enterprise
• Utilizing the best of both worlds allows you to take advantage of all the communities served by your business

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Attributes and Advantages of a RESTful SOA

• **Simplicity**
  – Many decisions pre-made, constrained choices
    • Protocol (HTTP), encryption model (HTTPs)
    • Identity token exchange (Basic-Auth, or standard HTTP schemes)

• **Ubiquity**
  – Use the HTTP infrastructure and other technologies like JavaScript that already exist

• **Effortless use of services**
  – Single, well-understood programming model (Javascript)
  – Lots of examples on the web
    • Copy-cut-and-paste programming to use services
  – Someone should be able to use a RESTful SOA Service without knowing they’re doing it!

• **Cacheability, Scalability, Testability, Secureability, Navigability**
Summary

• SOA is here to stay
  – Exposing services not just functions
  – SOAP like JSON and REST is just an enablement technology

• Importance of choosing a technology to expose your services that can be consumed by the client

• CICS implements key technologies to allow you to ‘take your pick’ to build your SOA strategy