A Pragmatic Introduction to REST

QCon London 2008
Stefan Tilkov, stefan.tilkov@innoq.com

Stefan Tilkov

http://www.innoQ.com

stefan.tilkov@innoq.com

http://www.innoq.com/blog/st/

http://www.InfoQ.com

REST vs. ... ?

REST vs. SOAP?

REST vs. SOA?

REST vs. WS-*?

Not today

(At least we'll try)

First, let's define some things

What is SOA?

3 Possible Definitions

Take your pick

SOA: An Approach to Business/IT Alignment

A different approach to an enterprise's IT architecture ...

... driven by business, not technology

... focusing on shared and re-used functionality

... aligning business and IT

... relying on strong governance

SOA: An Approach to Business/IT Alignment

... can be implemented using any architecture, technology, or set of products

2

SOA: A Technical Architecture

Services with clearly defined interfaces

... autonomous and with explicit boundaries

... relying on shared schema, not shared code

... programming language-independent

... separating interface and implementation

... containing multiple specific operations

SOA: A Technical Architecture

… *somewhat* technology-independent – can be built with e.g. CORBA, DCE RPC, DCOM, RMI, or Web services.

3

SOA = Web Services

Business data as XML messages

... sent in a SOAP body

... enriched with metadata in SOA headers

... described with WSDL and XML Schema

... configured through WS-Policy

... registered in a UDDI registry

SOA = Web Services

... implemented using technologies and products from the WS-* universe

Web Services Standards Overview

Interoperability Issues

Basic Profile 1.1 · WS-I · Final Specification

▲ **Basic Profile –** The Basic Profile 1.1 provides implementation guidelines for how related set of non-proprietary Web Service specifications should be used together for best interoperability.

Basic Profile 1.2 · WS-I · Working Group Draft

▲ **Basic Profile –** The Basic Profile 1.2 builds on Basic Profile 1.1 by incorporating Basic Profile 1.1 errata, requirements from Simple SOAP Binding Profile 1.0, and adding support for WS-Addressing and MTOM.

Basic Profile 2.0 · WS-I · Working Group Draft

▲ **Basic Profile –** The Basic Profile 2.0 is an update of WS-I BP that includes a profile of SOAP 1.2.

Attachments Profile 1.0 · WS-I · Final Specification

▲ **Attachments Profile –** The Attachment Profile 1.0 complements the Basic Profile 1.1 to add support for interoperable SOAP Messages with attachments-based Web Services.

Simple SOAP Binding Profile 1.0 · WS-I · Final Specification

▲ **Simple SOAP Binding Profile –** The Simple SOAP Binding Profile consists of three Basic Profile 1.0 requirements related to the serialization of the envelope and its representation in the message.

Basic Security Profile 1.0 · WS-I · Board Approval Draft

▲ **Basic Security Profile** defines the WS-I Basic Security Profile 1.0, based on a set of non-proprietary Web services specifications, along with clarifications and amendments to those specifications which promote interoperability.

REL Token Profile 1.0 · WS-I · Working Group Draft

▲ **REL Token Profile** is based on a non-proprietary Web services specification, along with clarifications and amendments to that specification which promote interoperability.

SAML Token Profile 1.0 · WS-I · Working Group Draft

▲ **SAML Token Profile** is based on a non-proprietary

Business Process Specifications

Business Process Execution Language for Web Services 1.1 (BPEL4WS) · 1.1 · BEA Systems, IBM, Microsoft, SAP, Siebel Systems · OASIS-Standard

▲ **Business Process Execution Language for Web Services 1.1** (BPEL4WS) provides a language for the formal specification of business processes and business interaction protocols using Web Services.

WS-Choreography Model Overview 1.0 · W3C · Working Draft

▲ **WS-Choreography Model Overview** defines the format and structure of the (SOAP) messages that are exchanged, and the sequence and conditions in which the messages are exchanged.

Web Service Choreography Interface (WSCI) · 1.0 · W3C · Sun Microsystems, SAP, BEA Systems and Intalio · Note

▲ **Web Service Choreography Interface** (WSCI) describes how Web Service operations can be choreographed in the context of a message exchange in which the Web Service participates.

Web Service Choreography Description Language (CDL4WS) · 1.0 · W3C · Candidate Recommendation

▲ **Web Service Choreography Description Language** (CDL4WS) is to specify a declarative, XML based language that defines from a global viewpoint the common and complementary observable behaviour, where message exchanges occur, and when the jointly agreed ordering rules are satisfied.

Business Process Execution Language for Web Services 2.0 (BPEL4WS) · 2.0 · OASIS, BEA Systems, IBM, Microsoft, SAP, Siebel Systems · Committee Draft

▲ **Business Process Execution Language for Web Services 2.0** (BPEL4WS) provides a language for the formal specification of business processes and business interaction protocols using Web Services.

Business Process Management Language (BPML) · 1.1 · BPMI.org · Final Draft

▲ **Business Process Management Language** (BPML) provides a meta-language for expressing business processes and supporting entities.

XML Process Definition Language (XPDL) · 2.0 · Final

▲ **XML Process Definition Language** (XPDL) provides an XML file format that can be used to interchange process models between tools.

Management Specifications

Management Using Web Services (WSDM-MUWS) · 1.0 · OASIS · OASIS-Standard

▲ **Web Service Distributed Management:** Management Using Web Services (WSDM-MUWS) defines how an IT resource connected to a network provides manageability interfaces such that the IT resource can be managed locally and from remote locations using Web services technologies.

Management Of Web Services (WSDM-MOWS) · 1.0 · OASIS · OASIS-Standard

▲ **Web Service Distributed Management:** Management Of Web Services (WSDM-MOWS) addresses management of the components that form the network, the Web services endpoints, using Web services protocols.

WS-Management · AMD, Dell, Intel, Microsoft and Sun Microsystems · Published Specification

▲ **WS-Management** describes a general SOAP-based protocol for managing systems such as PCs, servers, devices, Web services and other applications, and other manageable entities.

Service Modeling Language · IBM, BEA, BMC, Cisco, Dell, HP, Intel, Microsoft, Sun · Draft Specification

▲ **Service Modeling Language** (SML) is used to model complex IT services and systems, including their structure, constraints, policies, and best practices.

Presentation Specifications

Web Services for Remote Portlets (WSRP) · 2.0 · OASIS · Committee Draft

▶ **Web Services for Remote Portlets** (WSRP) defines a set of interfaces and related semantics which standardize interactions with components providing user-facing markup, including the processing of user interactions with that markup.

Metadata Specifications

WS-Policy · 1.5 · W3C · Working Draft

▲ **WS-Policy** describes the capabilities and constraints of the policies on intermediaries and endpoints (e.g. business rules, required security tokens, supported encryption algorithms, privacy rules).

WS-PolicyAssertions · 1.1 · BEA Systems, IBM, Microsoft, SAP · Public Draft

▲ **WS-PolicyAssertions** provides an initial set of assertions to address some common needs of Web Services applications.

WS-PolicyAttachment · 1.2 · W3C · W3C Member Submission

▲ **WS-PolicyAttachment** defines two general-purpose mechanisms for associating policies with the subjects to which they apply; the policies may be defined as part of existing metadata about the subject or the policies may be defined independently and associated through an external binding to the subject.

WS-Discovery · Microsoft, BEA Systems, Canon, Intel and webMethods · Draft

▲ **WS-Discovery** defines a multicast discovery protocol for dynamic discovery of services on ad-hoc and managed networks.

WS-MetadataExchange · 1.1 · BEA Systems, Computer Associates, IBM, Microsoft, SAP, Sun Microsystems and webMethods · Public Draft

▲ **WS-MetadataExchange** enables a service to provide metadata to others through a Web services interface. Given only a reference to a Web service, a user can access a set of WSDL/SOAP operations to retrieve the metadata that describes the service.

Universal Description, Discovery and Integration (UDDI) · 3.0.2 · OASIS · OASIS-Standard

▲ **Universal Description, Discovery and Integration (UDDI)** defines a set of services supporting the description and discovery of businesses, organizations, and other Web services providers, the Web services they make available, and the technical interfaces which may be used to access those services.

Web Service Description Language 2.0 SOAP Binding · 2.0 · W3C · Working Draft

▲ **Web Service Description Language SOAP Binding** describes the concrete details for using WSDL 2.0 in conjunction with SOAP 1.1 protocol.

Web Service Description Language 2.0 Core · 2.0 · W3C · Candidate Recommendation

▲ **Web Service Description Language 2.0 Core** is an XML-based language for describing Web services and how to access them. It specifies the location of the service and the operations (or methods) the service exposes.

Web Service Description Language 1.1 · 1.1 · W3C · Note

▲ **Web Service Description Language 1.1** is an XML-based

Reliability Specifications

WS-ReliableMessaging · 1.1 · OASIS · Committee Draft

▲ **WS-ReliableMessaging** describes a protocol that allows Web services to communicate reliable in the presence of software component, system, or network failures. It defines a SOAP binding that is required for interoperability.

WS-Reliable Messaging Policy Assertion (WS-RM Policy) · 1.1 · OASIS · Committee Draft

▲ **Web Services ReliableMessaging Policy Assertion** (WS-RM Policy) describes a domain-specific policy assertion for WS-ReliableMessaging that that can be specified within a policy alternative as defined in WS-Policy Framework.

WS-Reliability · 1.1 · OASIS · OASIS-Standard

▲ **WS-Reliability** is a SOAP-based protocol for exchanging SOAP messages with guaranteed delivery, no duplicates, and guaranteed message ordering. WS-Reliability is defined as SOAP header extensions and is independent of the underlying protocol. This specification contains a binding to HTTP.

Security Specifications

WS-Security · 1.1 · OASIS · OASIS-Standard

▲ **WS-Security** is a communications protocol providing a means for applying security to Web Services.

WS-SecurityPolicy · 1.1 · IBM, Microsoft, RSA Security, VeriSign · Public Draft

▲ **WS-SecurityPolicy** defines how to describe policies related to various features defined in the WS-Security specification.

WS-Security: SOAP Message Security · 1.1 · OASIS · Public Review Draft

▲ **WS-Security: SOAP Message Security** describes enhancements to SOAP messaging to provide message integrity and confidentiality. Specifically, this specification provides support for multiple security token formats, trust domains, signature formats and encryption technologies. The token formats and semantics for using these are defined in the associated profile documents.

WS-Security: Username Token Profile · 1.1 · OASIS · Public Review Draft

▲ **WS-Security: Username Token Profile** describes how a Web Service consumer can supply a Username Token as a means of identifying the requestor by username, and optionally using a password (or shared secret, etc.) to authenticate that identity to the Web Service producer.

WS-Security: Kerberos Binding · 1.0 · Microsoft, IBM, OASIS · Working Draft

▲ **WS-Security: Kerberos Binding** defines how to encode Kerberos tickets and attach them to SOAP messages. As well, it specifies how to add signatures and encryption to the SOAP message, in accordance with WS-Security, which uses and references the Kerberos tokens.

WS-Federation · 1.0 · IBM, Microsoft, BEA Systems, RSA Security, and VeriSign · Initial Draft

▲ **WS-Federation** describes how to manage and broker the trust relationships in a heterogeneous federated environment including support for federated identities.

WS-Security: SAML Token Profile · 1.1 · OASIS · Public Review Draft

▲ **WS-Security: SAML Token Profile** defines the use of Security Assertion Markup Language (SAML) v1.1 assertions in the context of WSS: SOAP Message Security including for the purpose of securing SOAP messages and SOAP message exchanges.

WS-Trust · BEA Systems, Computer Associates, IBM, Layer 7 Technologies, Microsoft, Netegrity, Oblix, OpenNetwork, Ping Identity Corporation, Reactivity, RSA Security, VeriSign and Westbridge · Technology - Initial Draft

▲ **WS-Trust** describes a framework for trust models that enables Web Services to securely interoperate. It uses WS-Security base mechanisms and defines additional primitives and extensions for security token exchange to enable the issuance and dissemination of credentials within different trust domains.

WS-Security: X.509 Certificate Token Profile · 1.1 · OASIS · Public Review Draft

▲ **WS-Security: X.509 Certificate Token Profile** describes the use of the X.509 authentication framework with the WS-Security: SOAP Message Security specification.

WS-SecureConversation · BEA Systems, Computer Associates, IBM, Layer 7 Technologies, Microsoft, Netegrity, Oblix, OpenNetwork, Ping Identity Corporation, Reactivity, RSA Security, VeriSign and Westbridge · Technology -Public Draft

▲ **WS-SecureConversation** specifies how to manage and authenticate message exchanges between parties including security context exchange and establishing and deriving session keys.

Transaction Specifications

WS-Coordination · 1.1 · OASIS · Working Draft

▲ **WS-Coordination** describes an extensible framework for providing protocols that coordinate the actions of distributed applications.

WS-Business Activity · 1.1 · OASIS · Working Draft

▲ **WS-Business Activity** provides the definition of the business activity coordination type that is to be used with the extensible coordination framework described in the WS-Coordination specification.

WS-Atomic Transaction · 1.1 · OASIS · Committee Draft

▲ **WS-Atomic Transaction** defines protocols that enable existing transaction processing systems to wrap their proprietary protocols and interoperate across different hardware and software vendors.

WS-Composite Application Framework (WS-CAF) · 1.0 · Arjuna Technologies, Fujitsu, IONA, Oracle and Sun Microsystems · Committee Specification

▲ **WS-Composite Application Framework** (WS-CAF) is a collection of three specifications aimed at solving problems that arise when multiple Web Services are used in combination. It proposes standard, interoperable mechanisms for managing shared context and ensuring business processes achieve predictable results and recovery from failure.

WS-Context (WS-CTX) · 1.0 · Arjuna Technologies, Fujitsu, IONA, Oracle and Sun Microsystems · Committee Draft

▲ **WS-Context** (WS-CTX) is intended as a lightweight mechanism for allowing multiple Web Services to share a common context.

WS-Coordination Framework (WS-CF) · 1.0 · Arjuna Technologies, Fujitsu, IONA, Oracle and Sun Microsystems · Committee Draft

▲ **WS-Coordination Framework** (WS-CF) allows the management and coordination in a Web Services interaction of a number of activities related to an overall application.

WS-Transaction Management (WS-TXM) · 1.0 · Arjuna Technologies, Fujitsu, IONA, Oracle and Sun Microsystems · Committee Draft

▲ **WS-Transaction Management** (WS-TXM) defines a core infrastructure service consisting of a Transaction Service for Web Services.

Resource Specifications

Web Services Resource Framework (WSRF) · 1.2 · OASIS · OASIS-Standard

▲ **Web Services Resource Framework** (WSRF) defines a family of specifications for accessing stateful resources using Web Services.

WS-BaseFaults (WSRF) · 1.2 · OASIS · Working Draft

▲ **WS-BaseFaults** (WSRF) defines a base set of information that may appear in fault messages. WS-BaseFaults defines an XML schema type for base faults, along with rules for how this base fault type is used and extended by Web Services.

WS-ServiceGroup (WSRF) · 1.2 · OASIS · Working Draft

▲ **WS-ServiceGroup** (WSRF) defines a means by which Web Services and WS-Resources can be aggregated or grouped together for a domain specific purpose.

WS-ResourceProperties · 1.2 · OASIS · Working Draft

▲ **WS-ResourceProperties** specifies the means by which the definition of the properties of a WS-Resource may be declared as part of the Web Service interface. The declaration of the WS-Resource properties represents a projection of or a view on the WS-Resource state.

WS-ResourceLifetime · 1.2 · OASIS · Working Draft

▲ **WS-ResourceLifetime** is to standardize the terminology, concepts, message exchanges, WSDL and XML needed to monitor the lifetime of, and destroy WS-Resources. Additionally, it defines resource properties that may be used to inspect and monitor the lifetime of a WS-Resource.

WS-Transfer · W3C · W3C Member Submission

▲ **WS-Transfer** describes a general SOAP-based protocol for accessing XML representations of Web service-based resources.

Resource Representation SOAP Header Block (RRSHB) · W3C · Recommendation

▲ **Resource Representation SOAP Header Block** (RRSHB) complements MTOM by defining mechanisms for describing and conveying non-XML resource representations in a SOAP 1.2 message.

ecifications

WS-Notification is a family of related white papers and specifications that define a standard Web services approach to notification using a topic-based publish/subscribe pattern.

WS-BrokeredNotification · 1.3 · OASIS · OASIS-Standard

▲ **WS-BrokeredNotification** defines the interface for the NotificationBroker. A NotificationBroker is an intermediary, which, among other things, allows publication of messages from entities that are not themselves service providers.

WS-BaseNotification · 1.3 · OASIS · OASIS-Standard

▲ **WS-BaseNotification** standardizes the terminology, concepts, operations, WSDL and XML needed to express the basic roles involved in Web services publish and subscribe for notification message exchange.

WS-Eventing · W3C · Public Draft

▲ **WS-Eventing** defines a baseline set of operations that allow Web services to provide asynchronous notifications to interested parties.

WS-Addressing – Core · 1.0 · W3C · Recommendation

▲ **WS-Addressing – Core** provides transport-neutral mechanisms to address Web services and messages. This specification defines XML elements to identify Web service endpoints and to secure end-to-end endpoint identification in messages.

WS-Enumeration describes a general SOAP-based protocol for enumerating a sequence of XML elements that is suitable for traversing logs, message queues, or other linear information models.

WS-Topics · 1.3 · OASIS · OASIS-Standard

▲ **WS-Topics** defines three topic expression dialects that can be used as subscription expressions in subscribe request messages and other parts of the WS-Notification system.

WS-Addressing – WSDL Binding · 1.0 · W3C · Candidate Recommendation

▲ **WS-Addressing – WSDL Binding** defines how the abstract properties defined in Web Services Addressing – Core are described using WSDL.

WS-Addressing – SOAP Binding · 1.0 · W3C · Recommendation

▲ **WS-Addressing – SOAP Binding** provides transport-neutral mechanisms to address Web services and messages.

SOAP

SOAP · 1.2 · W3C · Recommendation

SOAP · 1.1 · W3C · Note

▲ **SOAP** is a lightweight, XML-based protocol for exchange of information in a decentralized, distributed environment.

SOAP Message Transmission Optimization Mechanism (MTOM) · 1.0 · W3C · Recommendation

▲ **SOAP Message Transmission Optimization Mechanism** describes an abstract feature for optimizing the transmission and/or wire format of a SOAP message.

tions

XML – Extensible Markup Language is a pared-down version of SGML, designed especially for Web documents. It allows one to create own customized tags, enabling the definition, transmission, validation, and interpretation of data between applications and between organizations.

XML 1.0 · 1.0 · W3C · Recommendation

▲ **XML** – Extensible Markup Language is a pared-down version of SGML, designed especially for Web documents. It allows one to create own customized tags, enabling the definition, transmission, validation, and interpretation of data between applications and between organizations.

Namespaces in XML · 1.1 · W3C · Recommendation

▲ **Namespaces in XML** provides a simple method for qualifying element and attribute names used in XML documents by associating them with namespaces identified by IRI references.

XML Information Set · 1.0 · W3C · Recommendation

▲ **XML Information Set** is an abstract data set to provide a consistent set of definitions for use in other specifications that need to refer to the information in a well-formed XML document.

XML Schema · 1.1 · W3C · Working Draft

▲ **XML Schema** – XML Schema Definition Language is an XML language for describing and constraining the content of XML documents.

XML binary Optimized Packaging (XOP) · 1.0 · W3C · Recommendation

▲ **XML binary Optimized Packaging (XOP)** is an XML language for describing and constraining the content of XML documents.

Describing Media Content of Binary Data in XML (DMCBDX) · W3C · Note

▲ **Describing Media Content of Binary Data in XML (DMCBDX)** specifies how to indicate the content-type associated with binary element content in an XML document and to specify, in XML Schema, the expected content-type(s) associated with binary element content.

Dependencies

Messaging Specifications

SOAP 1.1 · SOAP 1.2 · SOAP Message Transmission Optimization Mechanism · WS-Notification · WS-BaseNotification · WS-Topics · WS-BrokeredNotification · WS-Addressing – Core · WS-Addressing – SOAP Binding · WS-Addressing – WSDL Binding · WS-Eventing · WS-Enumeration

→ Resource · Security · Metadata

Metadata Specifications

WS-Policy · WS-PolicyAssertions · WS-PolicyAttachment · WS-Discovery · WS-MetadataExchange · Universal Description, Discovery and Integration · Web Service Description Language 1.1 · Web Service Description Language 2.0 Core · Web Service Description Language 2.0 SOAP Binding

→ Security · Messaging

Security Specifications

WS-Security · WS-Security: SOAP Message Security · WS-Security: Kerberos Binding · WS-Security: SAML Token Profile · WS-Security: X.509 Certificate Token Profile · WS-Security: Username Token Profile · WS-SecurityPolicy · WS-Trust · WS-Federation · WS-SecureConversation

→ Reliability · Messaging · Metadata

Reliability Specifications

WS-ReliableMessaging · WS-Reliability · WS-Reliable Messaging Policy Assertion

→ Transaction · Basic Profile · Security · Metadata

Resource Specifications

Web Service Resource Framework · WS-BaseFaults · WS-ServiceGroup · WS-ResourceProperties · WS-ResourceLifetime · WS-Transfer · Resource Representation SOAP Header Block (RRSHB)

→ Transaction · Security · Messaging

Management Specifications

WS-Management · Management Of Web Services · Management Using Web Services · Service Modeling Language

→ Metadata · Resource · Security · Messaging

Business Process Specifications

Business Process Execution Language for Web Services · Web Service Choreography Description Language · Web Service Choreography Interface · WS-Choreography Model Overview · Business Process Management Language · Business Process Execution Language for Web Serv. 2.0 · XML Process Definition Language

→ Reliability · Security · Messaging · Transaction

Transaction Specifications

WS-Business Activity · WS-Atomic Transaction · WS-Coordination · WS-Composite Application Framework · WS-Transaction Management · WS-Context · WS-Coordination Framework

→ Metadata · Messaging · Reliability · Security

Presentation Specifications

Web Services for Remote Portlets

→ Reliab. · Secur. · Mess.

http://www.innoq.com/resources/ws-standards-poster/

innoQ Deutschland GmbH
Halskestraße 17
D-40880 Ratingen
Phone +49 2102 77162-100
info@innoq.com · www.innoq.com

innoQ Schweiz GmbH
Gewerbestrasse 11
CH-6330 Cham
Phone +41 41 743 0111

Why is SOA so hard to define?

A Web service *is 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.*

W3C Web Services Architecture WG

http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/

*"**Service Oriented Architecture** is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations."*

OASIS SOA Reference Model

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm

*"An **Economy** is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations."*

Nick Gall, VP, Gartner

http://tech.groups.yahoo.com/group/service-orientated-architecture/message/9065

What is REST?

3 definitions again

REST: An Architectural Style

One of a number of "architectural styles"

... described by Roy Fielding in his dissertation

... defined via a set of *constraints* that have to be met

... architectural principles underlying HTTP, defined *a posteriori*

... with the Web as one particular instance

See: http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

2

REST: The Web Used Correctly

A system or application architecture

... that uses HTTP, URI and other Web standards "correctly"

... is "on" the Web, not tunneled through it

... also called "WOA", "ROA", "RESTful HTTP"

3

REST: XML without SOAP

Send plain XML (w/o a SOAP Envelope) via HTTP

... violating the Web as much as WS-*

... preferably use GET to invoke methods

... or tunnel everything through POST

... commonly called "POX"

Only option 1 is the right one (because Roy said so)

But we'll go with option 2 (and equate "REST" with "RESTful HTTP usage")

and avoid option 3 like the plague

REST Explained in 5 Easy Steps

1. Give Every "Thing" an ID

```
http://example.com/customers/1234

http://example.com/orders/2007/10/776654

http://example.com/products/4554

http://example.com/processes/sal-increase-234
```

2. Link Things To Each Other

```
<order self='http://example.com/orders/1234'>
  <amount>23</amount>
  <product ref='http://example.com/products/4554' />
  <customer ref='http://example.com/customers/1234' />
</order>
```

3. Use Standard Methods

GET	retrieve information, possibly cached
PUT	Update or create with known ID
POST	Create or append sub-resource
DELETE	(Logically) remove

4. Allow for Multiple "Representations"

```
GET /customers/1234
Host: example.com
Accept: application/vnd.mycompany.customer+xml

<customer>...</customer>
```

```
GET /customers/1234
Host: example.com
Accept: text/x-vcard

begin:vcard
...
end:vcard
```

5. Communicate Statelessly

```
GET /customers/1234
Host: example.com
Accept: application/vnd.mycompany.customer+xml
```

```
<customer><order ref='./orders/46'</customer>
```

shutdown

update software

replace hardware

startup

```
GET /customers/1234/orders/46
Host: example.com
Accept: application/vnd.mycompany.order+xml
```

```
<order>...</order>
```

time

Consequences

OrderManagementService

+ getOrders()
+ submitOrder()
+ getOrderDetails()
+ getOrdersForCustomers()
+ updateOrder()
+ addOrderItem()
+ cancelOrder()

CustomerManagementService

+ getCustomers()
+ addCustomer()
+ getCustomerDetails()
+ updateCustomer()
+ deleteCustomer()

«interface»
Resource

GET
PUT
POST
DELETE

/orders

GET - list all orders
PUT - unused
POST - add a new order
DELETE - unused

/orders/{id}

GET - get order details
PUT - update order
POST - add item
DELETE - cancel order

/customers

GET - list all customers
PUT - unused
POST - add new customer
DELETE - unused

/customers/{id}

GET - get customer details
PUT - update customer
POST - unused
DELETE - delete customer

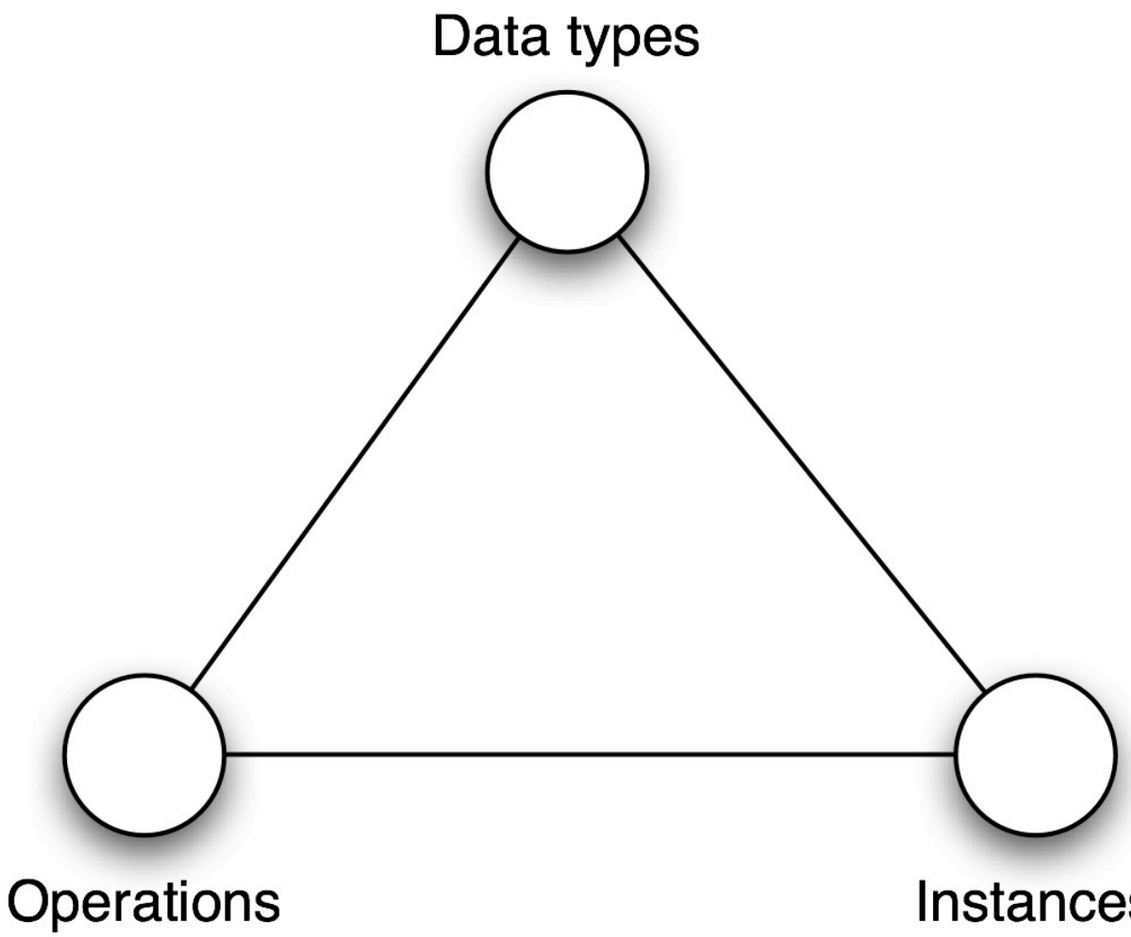
/customers/{id}/orders

GET - get all orders for customer
PUT - unused
POST - add order
DELETE - cancel all customer orders

Cheating?

Maybe.

many



many

very few

(one per service)

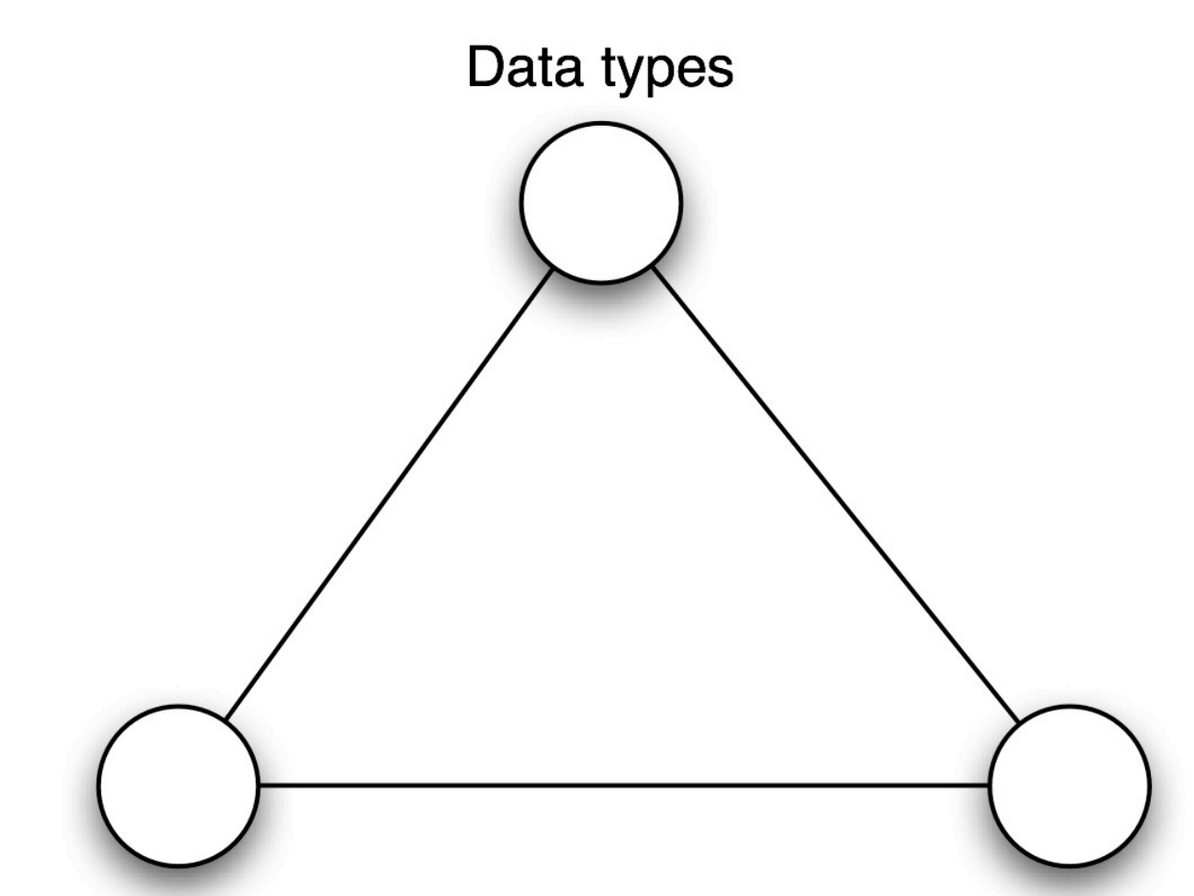
OrderManagementService

+ getOrders()
+ submitOrder()
+ getOrderDetails()
+ getOrdersForCustomers()
+ updateOrder()
+ addOrderItem()
+ cancelOrder()

CustomerManagementService

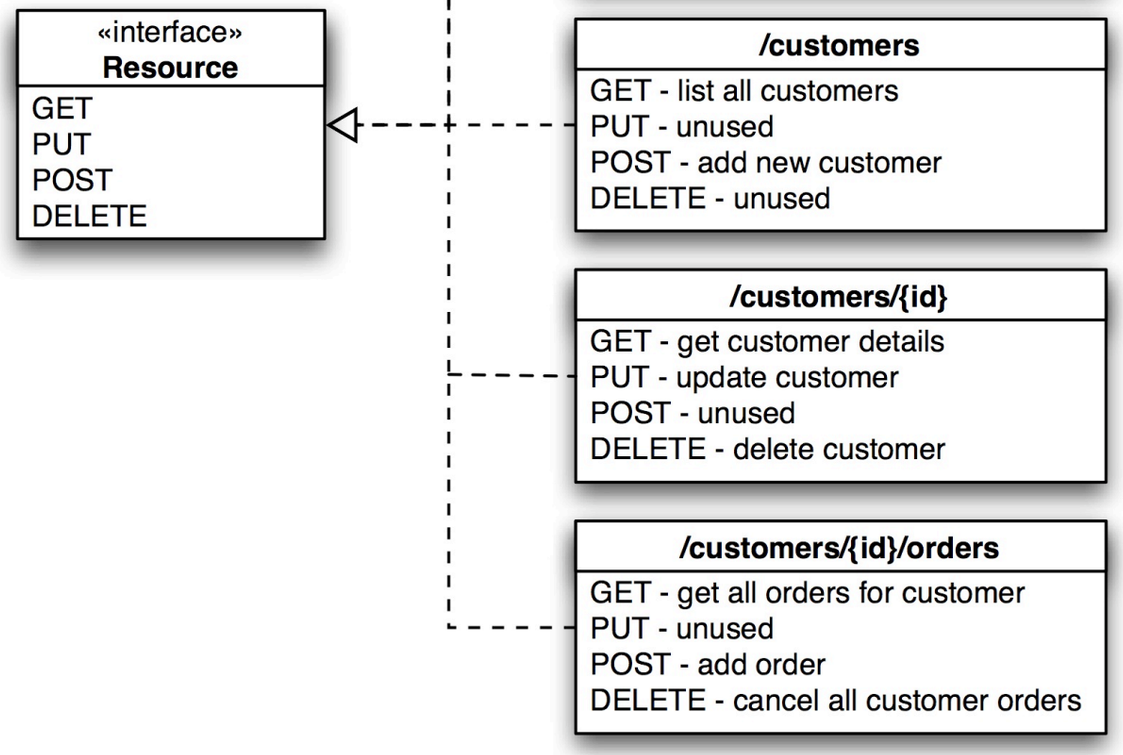
+ getCustomers()
+ addCustomer()
+ getCustomerDetails()
+ updateCustomer()
+ deleteCustomer()

many



very few
(fixed)

many



Designing a RESTful Application

Identify resources & design URIs

Select formats (or create new ones)

Identify method semantics

Select response codes

See: http://bitworking.org/news/How_to_create_a_REST_Protocol

What's cool about REST?

A very rough analogy (in pseudocode)

```
interface Resource {
      Resource(URI u)
      Response get()
      Response post(Request r)
      Response put(Request r)
      Response delete()
}
```

generic

Any HTTP client
(Firefox, IE, curl, wget)

Any HTTP server

Caches

Proxies

Google, Yahoo!, MSN

```
class CustomerCollection : Resource {
      ...
      Response post(Request r) {
            id = createCustomer(r)
            return new Response(201, r)
      }
      ...
}
```

Anything that knows
your app

specific

generic

```
interface Resource {
      ...
}
```

Anything that understands HTTP

```
class AtomFeed : Resource {
      AtomFeed get()
      post(Entry e)
      ...
}
```

Any feed reader

Any AtomPub client

Yahoo! Pipes

```
class CustomerCollection : AtomFeed {
      ...
}
```

Anything that knows your app

specific

Some HTTP Features

Verbs (in order of popularity):

GET, POST

PUT, DELETE

HEAD, OPTIONS, TRACE

Standardized (& meaningful) response codes

Content negotiation

Redirection

Caching (incl. validation/expiry)

Compression

Chunking

RESTful HTTP Advantages

Universal support (programming languages, operating systems, servers, ...)

Proven scalability

Real web integration for machine-2-machine communication

Support for XML, but also other formats

REST and Web Services

Web Services Issues

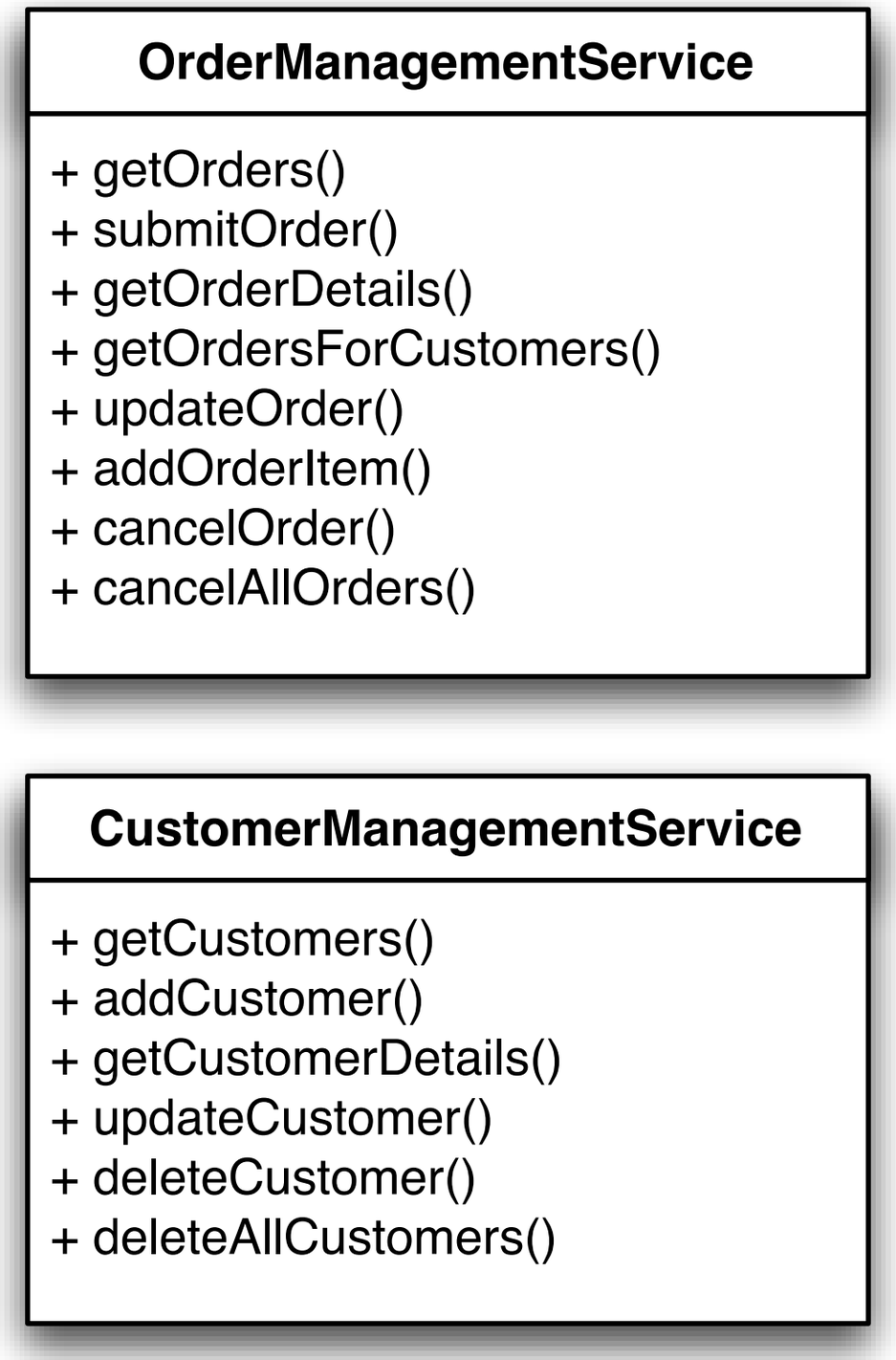
Web Services are "Web" in name only

WS-* tends to ignore the web

Abstractions leak, anyway

Protocol independence is a bug, not a feature

Web Services



A separate interface (façade) for each purpose

As known CORBA, DCOM, RMI/EJB

Often used for SOA ("CORBA w/ angle brackets)

Application-specific protocol

Contribution to the Net's Value

2 URLs

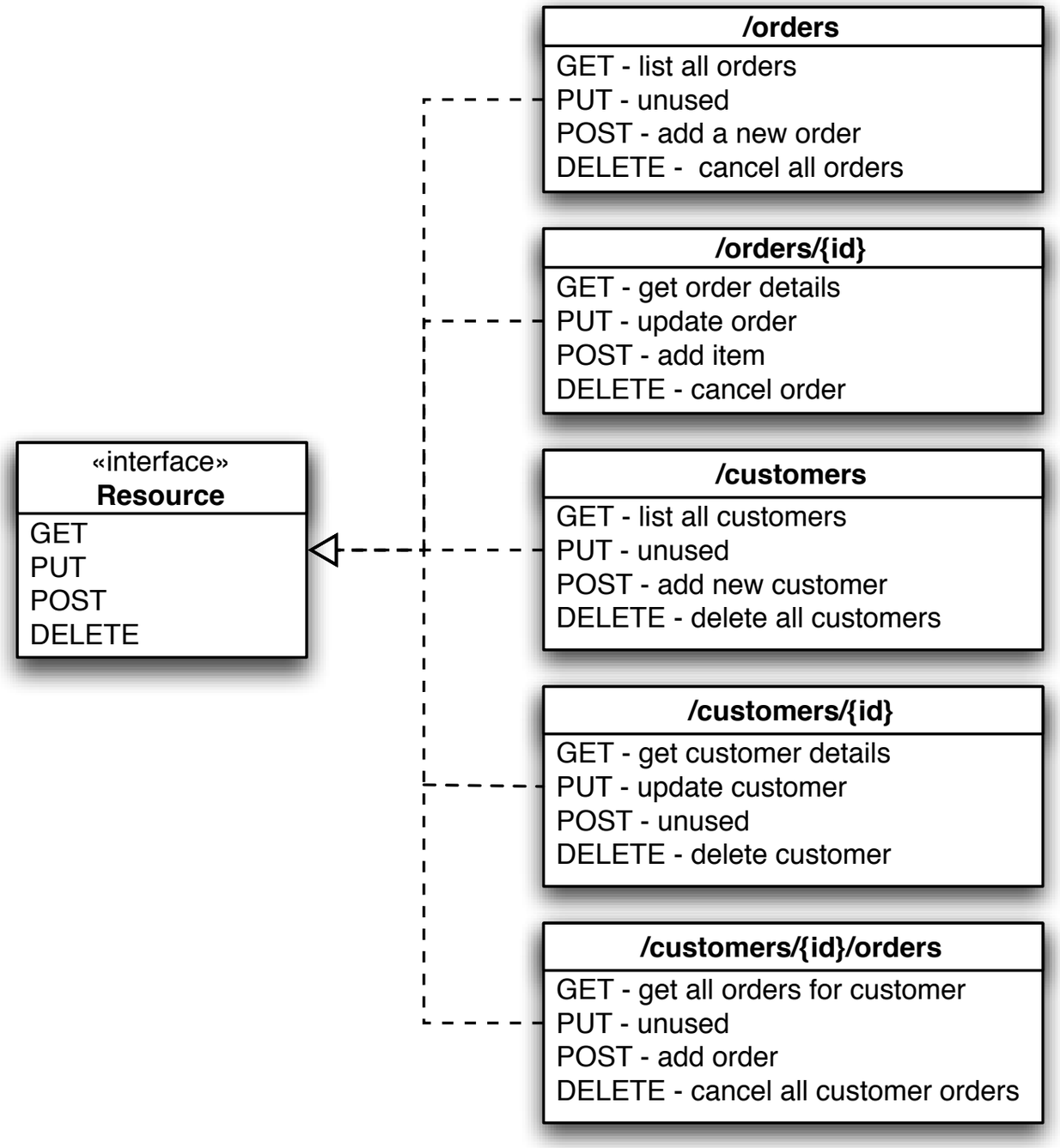
http://example.com/customerservice

http://example.com/orderservice

1 method

POST

REST Approach



A single *generic* (uniform) interface for everything

Generic verbs mapped to resource semantics

A standard application protocol (e.g. HTTP)

Contribution to the Net's Value

Millions of URLs

every customer

every order

4-6 supported methods per resource

GET, PUT, POST, DELETE, OPTIONS, HEAD

Cacheable, addressable, linkable, ...

REST for SOA

Business	SOA as an approach to business/IT alignment	
Architecture	Technical SOA	REST
Technology	SOAP, WSDL, WS-*	(RESTful) HTTP, URI, ...

REST as an alternative way to achieve SOA goals

Why You Should Care

WS-* Roots

The Enterprise

RPC, COM, CORBA, RMI, EJB

Transaction Systems

Controlled Environment

Top-down Approach

REST Roots

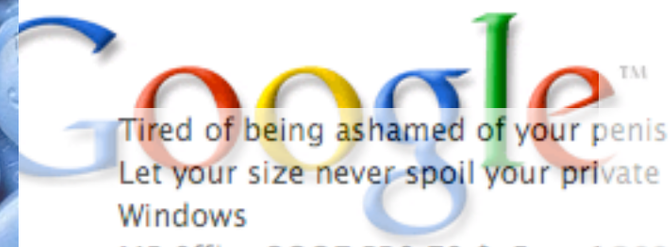
The Internet

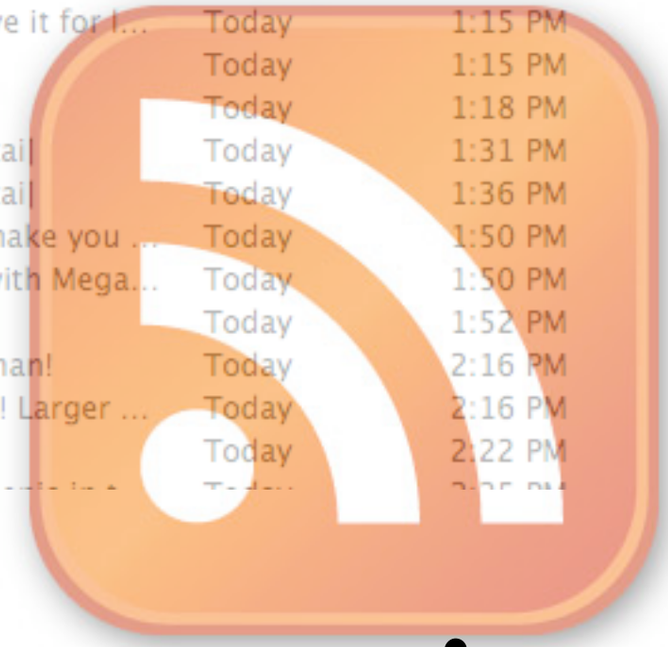
Text formats

Wire Standards

FTP, POP, SMTP

Bottom-up Approach

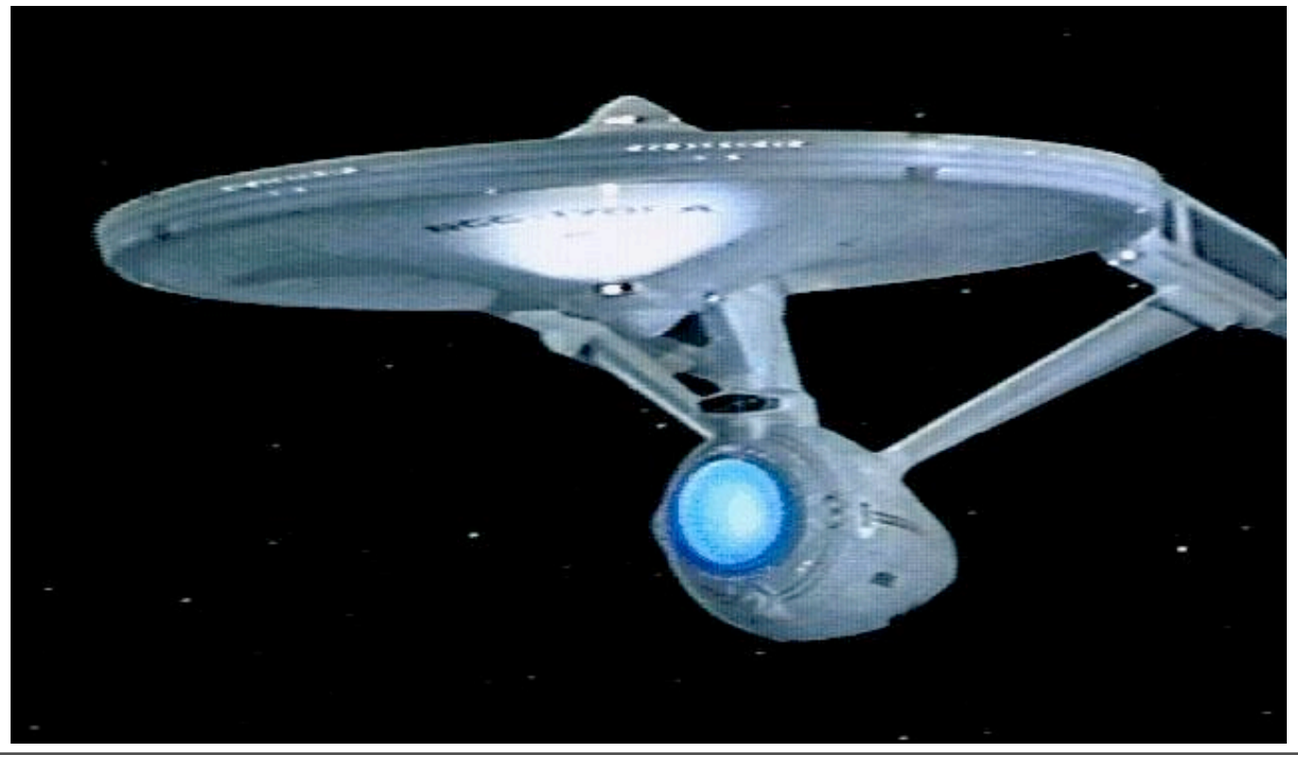




Octavio C. Dudley
Octavio R. Dudley
Luella Bates
Terrance Wells
Dean Snyder
Marguerite E. Lord
Marguerite J. Lord
Sales
Lee J. Lowry
Lee G. Lowry
//±q

Tired of being ashamed of your penis size? Leave it for l... Today 1:15 PM
Let your size never spoil your private life! Today 1:15 PM
Windows Today 1:18 PM
M5 0ff|ce 2OO7 PR0 79 $, 5ave 1O99.9S 0ff Retai| Today 1:31 PM
M5 0ff|ce 2OO7 PR0 79 $, 5ave 1O99.9S 0ff Retai| Today 1:36 PM
Ordinary men have ordinary sex. Megadik will make you ... Today 1:50 PM
Take care of you and your penis! Enlargement with Mega... Today 1:50 PM
dookit rcd trichosis Today 1:52 PM
Prove your manliness! Take MegaDik and be a man! Today 2:16 PM
Don't be embarrassed every time you get naked! Larger ... Today 2:16 PM
éöxÈç·é±ÆàÅ«Ü· Today 2:22 PM

Internet vs. Enterprise



What's the difference between the Internet and a typical enterprise?

Internet vs. Enterprise

One is a gigantic, uncontrollable anarchy of heterogeneous systems with varying quality that evolve independently and constantly get connected in new and unexpected ways.

The other is a worldwide, publicly accessible series of interconnected computer networks that transmit data by packet switching using the standard Internet Protocol (IP).

REST Support

Everybody	HTTP Servers, Clients, Proxies, Libraries, ...
DHH & The Rails Community	Ruby on Rails
Google	Base GData Calendar Document Lists Blogger Notebook Picasa
Amazon	Simple Storage Service (S3) Queue Service Flexible Payment Search
Sun	JSR 311 Jersey
IBM	Abdera Project Zero
Microsoft	Astoria WCF

Advanced Stuff

Description

What's the WSDL equivalent in REST?

There is none ...

XSD (95% of WSDL) is available to you, anyway

Of the remaining 5%, 90% is just silly

Why would you want to describe the uniform interface over and over again?

... unless you insist

WADL (Web Application Description Language)
https://wadl.dev.java.net/

Use URI Templates to define resource behavior

WADL Example

```
<resources base="http://api.search.yahoo.com/NewsSearchService/V1/">
    <resource path="newsSearch">
        <method name="GET" id="search">
            <request>
                <param name="appid" type="xsd:string" style="query" required="true"/>
                <param name="query" type="xsd:string" style="query" required="true"/>
                <param name="type" style="query" default="all">
                    <option value="all"/>
                    <option value="any"/>
                    <option value="phrase"/>
                </param>
                <param name="results" style="query" type="xsd:int" default="10"/>
                <param name="start" style="query" type="xsd:int" default="1"/>
                <param name="sort" style="query" default="rank">
                    <option value="rank"/>
                    <option value="date"/>
                </param>
                <param name="language" style="query" type="xsd:string"/>
            </request>
            <response>
                <representation mediaType="application/xml" element="yn:ResultSet"/>
                <fault status="400" mediaType="application/xml" element="ya:Error"/>
            </response>
        </method>
    </resource>
</resources>
```

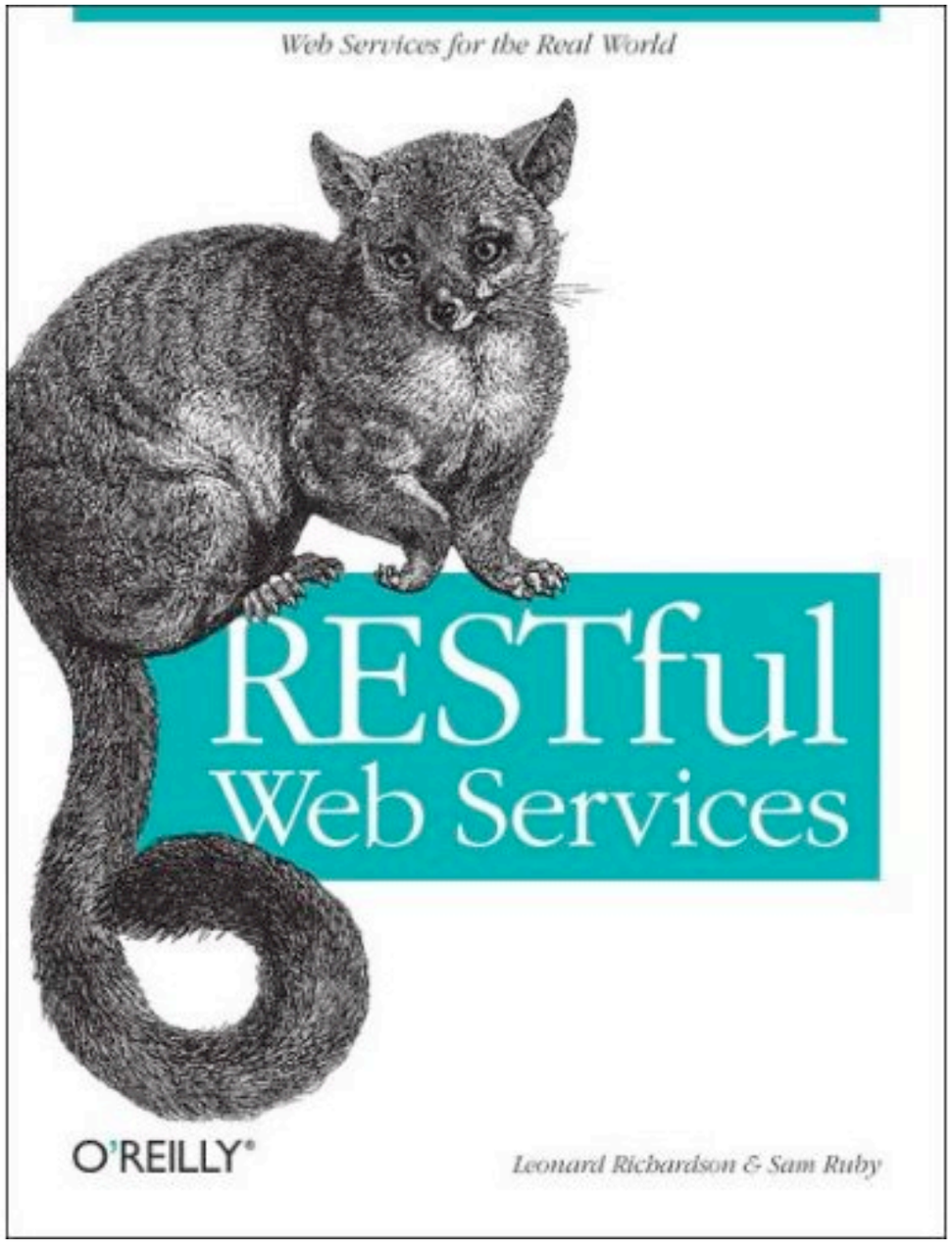
What You Should Do
(in my very humble opinion)

Be skeptical of WS-*
Learn more about REST
Learn to love the URI

Appreciate the Web

If You Want to Know More

http://www.innoq.com/resources/REST



http://www.oreilly.com/catalog/9780596529260/

http://www.infoq.com/

Random Stuff | innoQ | InfoQ | SOA-Expertenwissen | Txp › SOA-E...ertenwissen | innoQ | Partner | banking | awstats | js | 1&1 | O1BC.DE | AR | RailsDocs

InfoQ

Tracking change and innovation in the enterprise software development community

Version 1.4 | En | 中文 | 日本語

174,255 Sep unique visitors

Welcome, Stefan!

Sign out
Preferences
About us
Personal feed
Home

Your Communities

- [x] Java
- [ ] .NET
- [x] Ruby
- [x] SOA
- [ ] Agile
- [x] Architecture

Search

Featured Topics

Performance & Scalability

SOA Governance

Public Beta
Now Available

New & Notable Written for InfoQ by the Community

Contribute News

DynamicJasper: Runtime generation of Jasper Reports

Community Java Topics Open Source

DynamicJasper, an open-source API which provides runtime generation of Jasper Reports, recently released version 1.3. InfoQ took the opportunity to learn more about this product, and what it provides for users. By Ryan Slobojan on Oct 08 Discuss

Presentation: Architecture Evaluation in Practice

Community Architecture Topics Delivering Quality, Enterprise Architecture

Dragos Manolescu shares insights gained from growing ThoughtWorks' architecture evaluation practice and evaluating several architectures for Global 1000 companies. These insights aim at preparing people interested in commiss
participating in, or an
evaluation to tackle th
Marinescu on Oct 08 D

Ruby and the hype cycle

Community Ruby Topics Performance & Scalability, Ruby on Rails, Stories & Case Studies

A recent blog post on a failed Rails project caused a big debate about the viability of Ruby on Rails. A closer look at the post paints a different picture, though. We take a look at the reactions in the Ruby community, and compare this discussion with the upheaval about Twitter earlier this year. By Werner Schuster on Oct 08 3 comments

Adobe Max 2007 North America - Wrap Up

Community Java Topics Rich Client / Desktop, Acquisitions, Rich Internet Apps

Adobe was busy this week showing off their latest work at the 2007 Max Conference. Adobe continues to cater to developers with many of their efforts. The conference came with a number of interesting and exciting announcements for the developer community including: By Jon Rose on Oct 05 Discuss

Exclusive Content

All | Articles | Presentations | Interviews | Books

Steve Sloan on BizTalk Server 2006 R2

InfoQ talked to Steve Sloan, Senior Product Manager, about the BizTalk Server 2006 R2 in the context of SOA. SOA, Oct 04, 2007,

Open Source WS Stacks for Java - Design Goals and Philosophy

InfoQ spoke to the lead developers of the most important open source Java Web-services stacks about their design goals, standards, data binding, XML, interoperability, REST support, and maturity. SOA, Java, Oct 04, 2007, 7

Creating dynamic web applications with JSF/DWR/DOJO

JSF, DWR, and Dojo are all popular technologies in their own right. This article looks at how they can be integrated together in a portal environment. Java, Oct 04, 2007, 1

Architecture Evaluation in

Thank you!

Stefan Tilkov
http://www.innoq.com/blog/st/
stefan.tilkov@innoq.com

innoQ Deutschland GmbH
Halskestraße 17
D-40880 Ratingen
Phone +49 21 02 77 162-100

innoQ Schweiz GmbH
Gewerbestrasse 11
CH-6330 Cham
Phone +41 41 743 01 11

info@innoq.com · www.innoq.com

Photo Credit

http://www.flickr.com/photos/toddography/462611643/

http://www.flickr.com/photos/raveller/159146501/

http://en.wikipedia.org/wiki/Image:Sangreal.jpg