Web Services Standards and Best Practices within WebSphere

Greg Truty
Senior Technical Staff Member, WebSphere Web Services
SWG co-chair WS-* Strategy

gtruty@us.ibm.com

Objective

- The following session will describe:
  - WebSphere's view on the standards industry
  - WebSphere *(IBMs)* current state in implementing web services standards
  - What WebSphere is doing to help achieve interoperability
  - Best practices customers should know for implementing interoperable web services

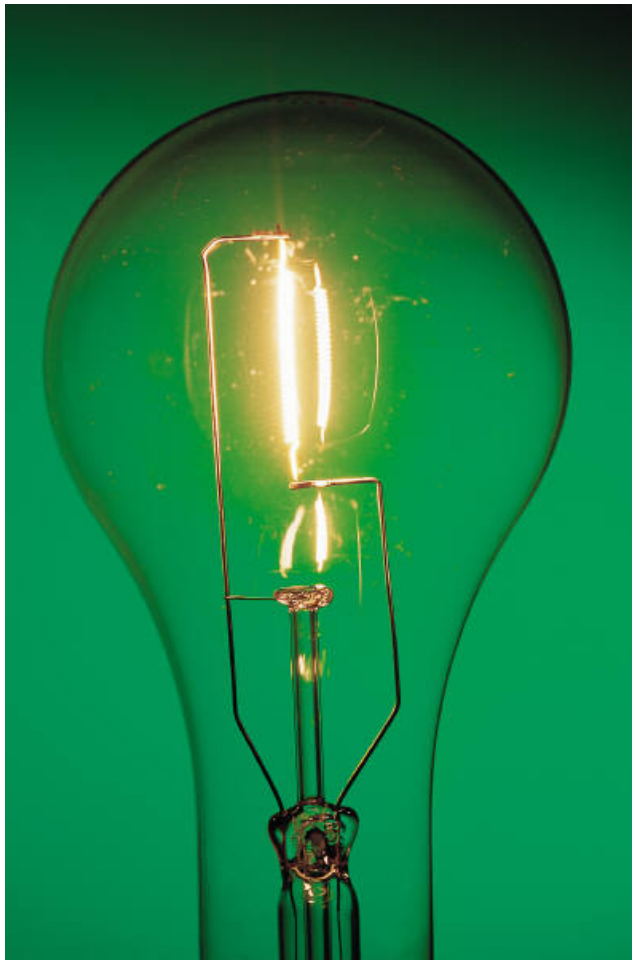
WebSphere's view on the standards industry

Differences between Specifications and Standards

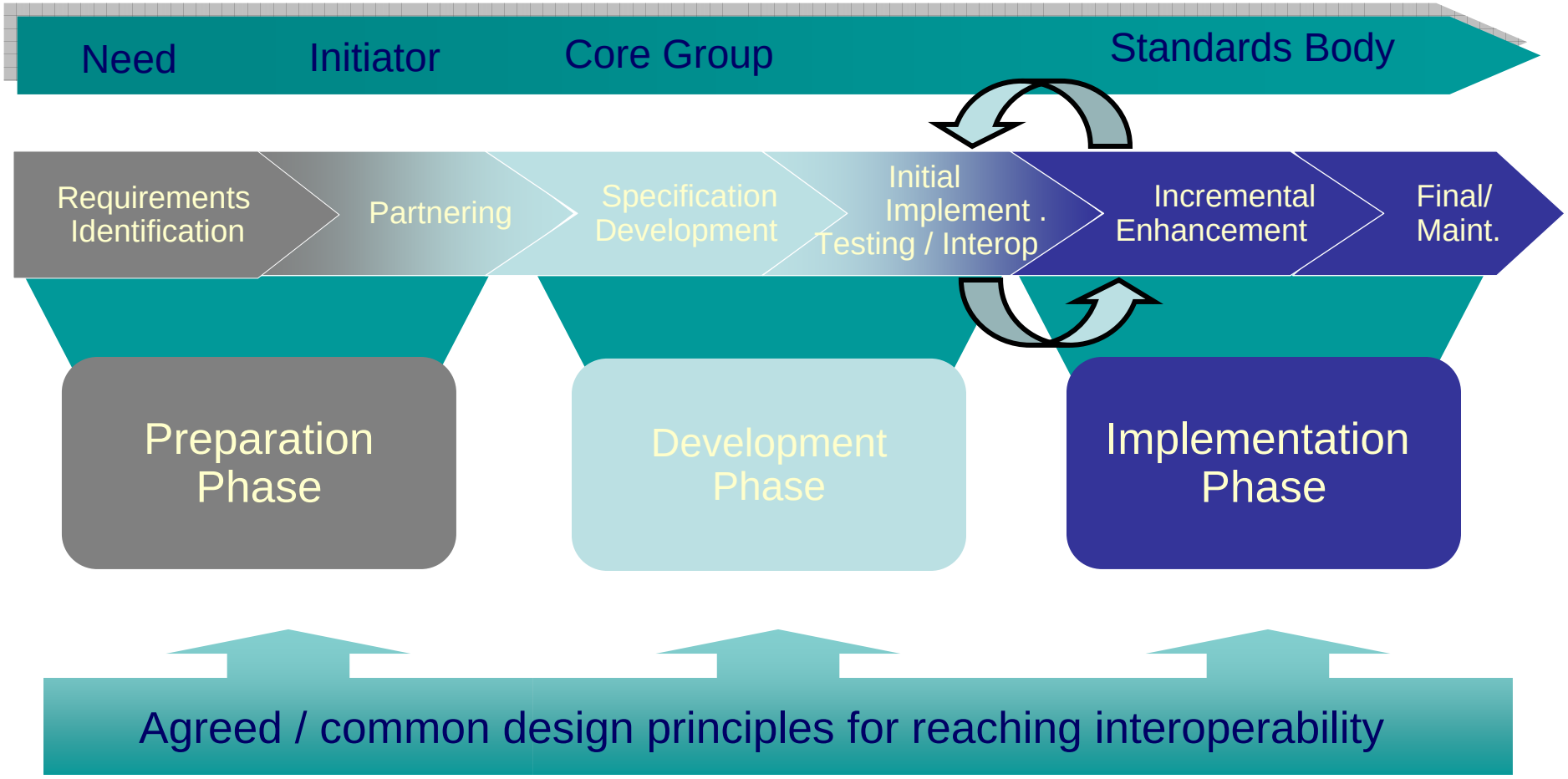
- A specification outlines APIs/data/formats from one or more vendors working together
  - Proposed by vendors to address a gap
  - Often supported by some vendor's software
- Usually submitted to a standards body
  - More companies get involved, specification changes, sometimes merged with competing specifications
  - Goes through multiple drafts and reviews
- Standard is agreed to by the standards body's committee members and is declared a 'version x' standard

Standardization underlies the ability to innovate

- **Collaboration** is key to innovation
- Open standards are essential to collaborative innovation: **interoperability is key**
- **Both proprietary** and **open source** are important forms of software development
- Patents should be granted only for what is new



Phases in Standardization



Standards Process - Ideal

- Industry customers provide comprehensive, unchanging requirements
- Standards bodies create unambiguous specifications
  - Orthogonal to all other specifications (or at worst, with the full knowledge of and cooperation of complementary standards efforts
  - Foreseeing all possible usages
- Developers follow specifications exactly
  - Never need “best practices” because using the specification properly **is** the best practice

Standards Process - Reality

- Constantly changing requirements
  - Often discovered as a result of the standards they are supposed to drive
  - Using an iterative process is not enough, since so many different standards groups are working independently
- Standards are coupled at many different levels
  - May sometimes be ambiguous or unclear
  - Cannot really be tested until they are already established
  - Cannot rely on other standards to stand still
- Developers do not know which standards to use, how to use them or how to tell if a product actually supports standards

Not all Standards are Created Equally

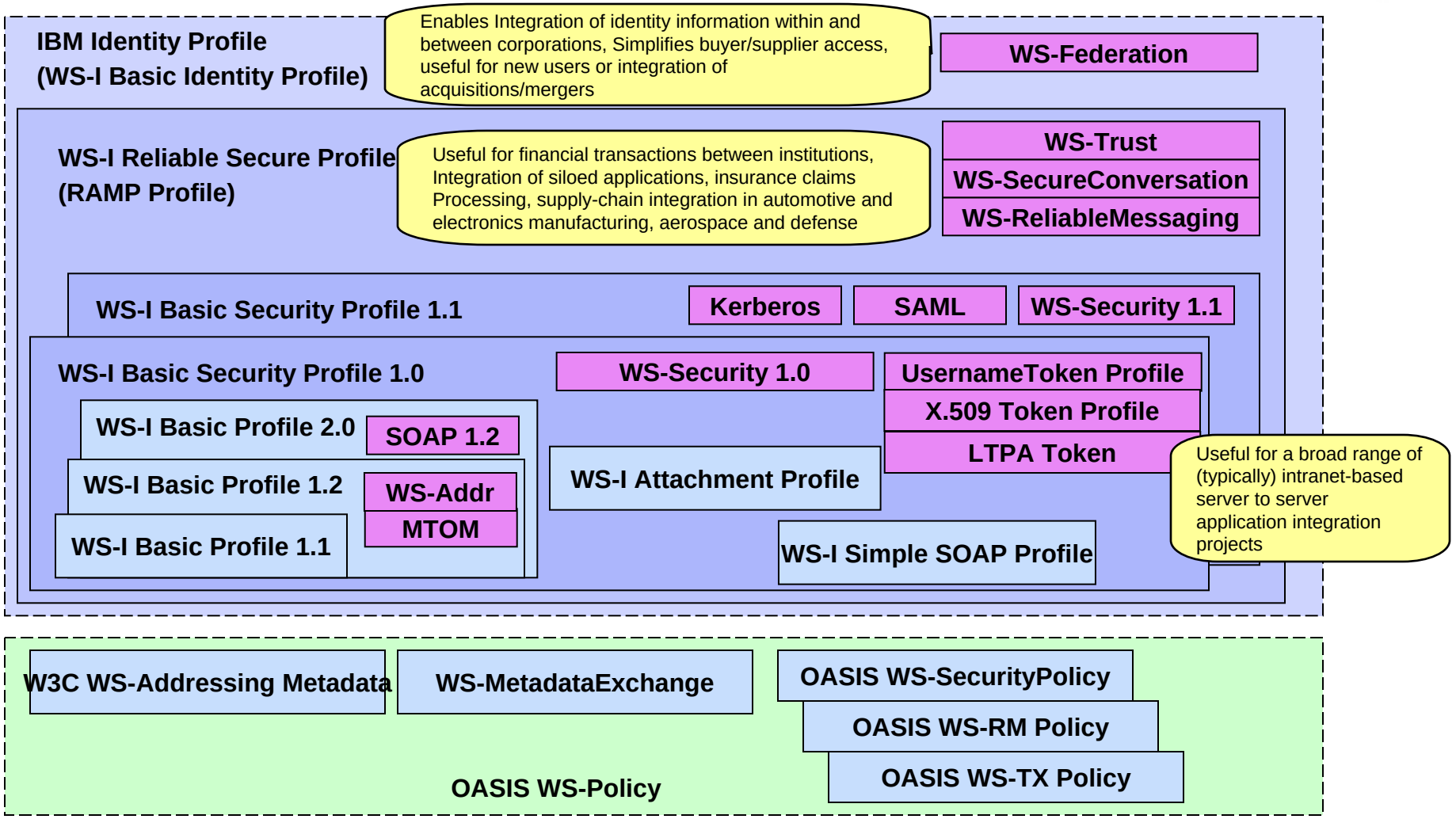
- Standards bodies vary by process, rigor, respect and adoption rate
  - And specific standards vary with respect to adoption
- Key standards bodies for SOA:
  - W3C, OASIS
  - WS-I
    - Consortium to promote interoperability of web services standards
- Most established, proven SOA standards:
  - XML, XML Schema, SOAP, WSDL, WS-Security
- Opinions vary as to whether Java Community Process (JCP) is a standards body

Java Community Process

- Does this matter in the standards domain?
  - YES!
  - Interoperability Scenarios exist in a context.
    - e.g. *a business problem needs to be solved*
  - The programming model exposed to an end-user may dictate behavior
    - e.g. *a WS-ReliableMessaging create_sequence request may be issued during a request*
  - The programming model may define limitations in what can/cannot be done
    - e.g. *JAX-RPC only defines a subset of mappings*

WebSphere's approach to Web Services

Making Standards *Simple* and *Useful* – Roadmap



WebSphere Web Service Standards Status

Web Services Standard	Introduced in	W3C or OASIS Status	Profile
WS-Security	5.0.2, 6.0, 6.1, 6.1 FeP for WS*	OASIS approved 1.0 – OASIS approved 1.1 -	BSP 1.0 BSP 1.1
W3C WS-Addressing	6.1	W3C recommend – 8/2005	BP 1.2, 2.0
WS-Notification	6.1	OASIS approved – 10/2006	
WS-Atomic Transaction	6.0 (submission)	OASIS approved – 6/2007	IBM BPMP
WS-BusinessActivity	6.1 (submission)	OASIS approved – 6/2007	IBM BPMP
WS-ReliableMessaging	6.1 FeP WS	OASIS approved – 6/2007	WS-I RSP 1.0
WS-SecureConversation	6.1 FeP WS (submission)	OASIS approved – 3/2007	WS-I RSP 1.0
WS-Trust	6.1 FeP WS (submission)	OASIS approved – 3/2007	WS-I RSP 1.0
W3C MTOM	6.1 FeP WS	W3C recommend – 11/2004	BP 1.2, 2.0
W3C SOAP 1.2	6.1 FeP WS	W3C recommend – 6/2003	BP 2.0
WS-DistributedManagement 1.1	6.1 FeP WS	OASIS approved – 8/2006	IBM SMP
Kerberos Token Profile		OASIS approved – 2/2006	BSP 1.0
WS-Addressing Metadata		W3C recommend – 9/2007	
WS-Policy		W3C recommend – 9/2007	
WS-SecurityPolicy		OASIS approved - 7/2007	

WS-I Profile Status

Basic Interoperability

- Basic Profile 1.1, Attachments Profile 1.0 – *closed*
- Basic Profile 1.2 * – ***Target closure 1H2008***
  - MTOM, WS-Addressing
- Basic Profile 2.0 * – ***Target closure 2H2008***
  - MTOM, WS-Addressing, SOAP 1.2

Reliable / Secure

- Reliable Secure Profile 1.0 – ***Target closure 1Q2009***
  - WS-ReliableMessaging, WS-SecureConversation, WS-Addressing

Secure

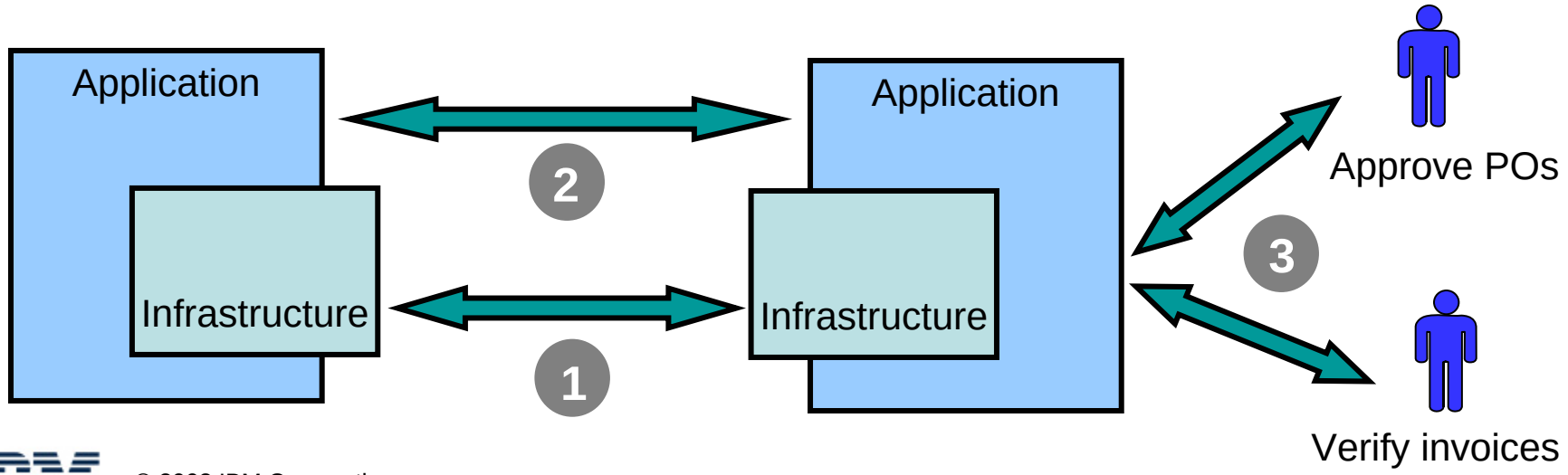
- Basic Security Profile 1.0
  - Username Token Profile, X.509 Token Profile – *closed*
- Basic Security Profile 1.1* – ***Target closure 2H2008***
  - Kerberos Token Profile, SAML Token Profile

*** BP 1.2, BP 2.0, BSP 1.1 - 5 interoperable implementations required for closure**

What WebSphere is doing to help achieve interoperability

Interoperability Layer

1. Technical Interoperability (Wire-level)
   - Messages are exchanged securely and reliably from sending and receiving infrastructure
   - Receiving infrastructure responsible for delivering the message payload to application
2. Semantic Interoperability (Application level)
   - Application knows the business context to which the payload belongs
   - Payload is valid from an application perspective
   - Application successfully processes payload
3. Organizational Interoperability (Process Level)
   - Application notifies appropriate users that are responsible for verification and approval steps and tracks deadlines



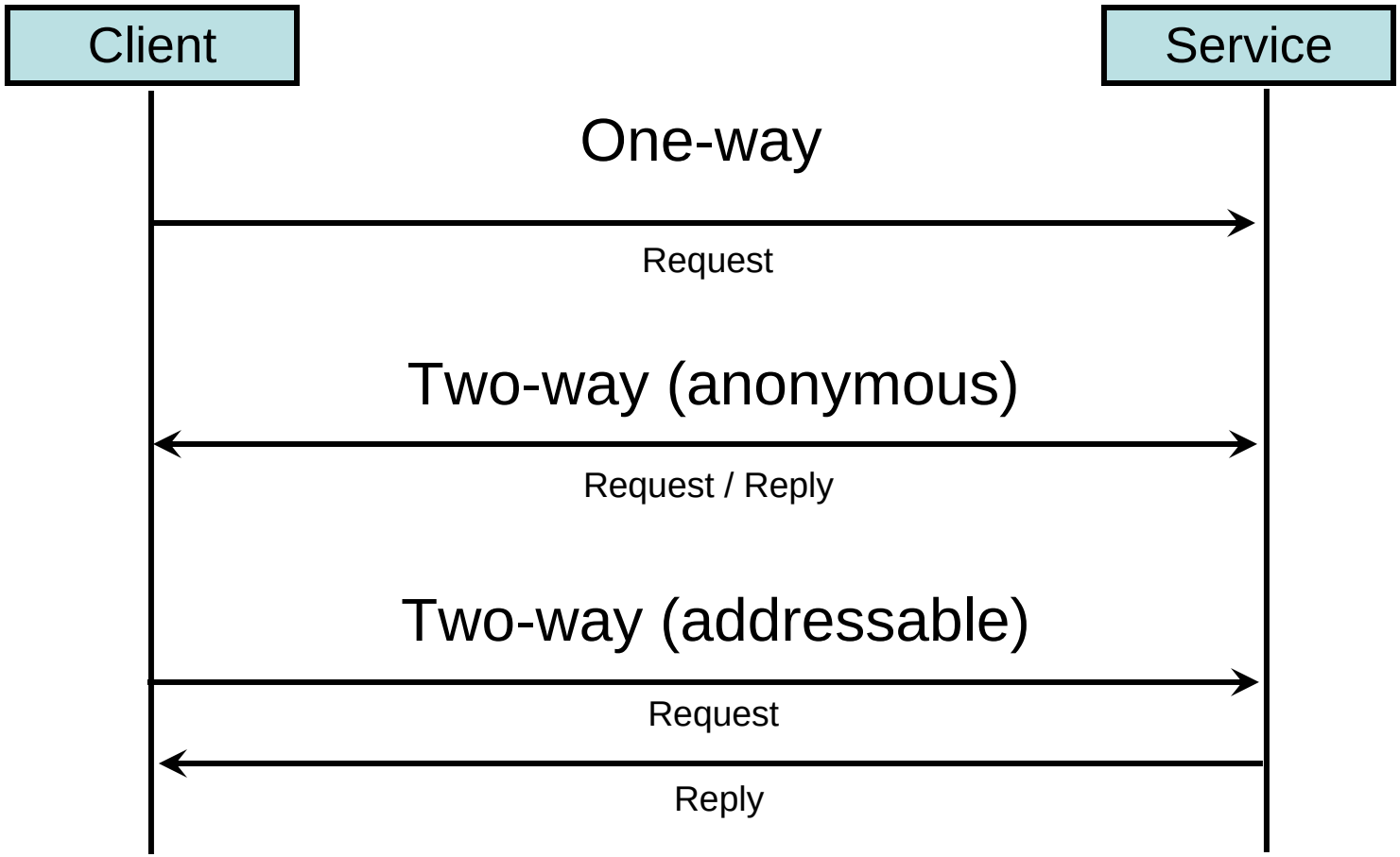
Interoperability Testing Methodology

- Technical interoperability (Wire-level)
  - Simple Message Exchange Pattern Test Suite
  - Samples shipped with WebSphere Web Services Feature Pack
- Technical & Semantic interoperability
  - ACORD Insurance Scenario
    - Jointly developed and tested with Microsoft
  - WS-I Supply Chain Management Sample
    - Donated to WS-I Sample Application Group

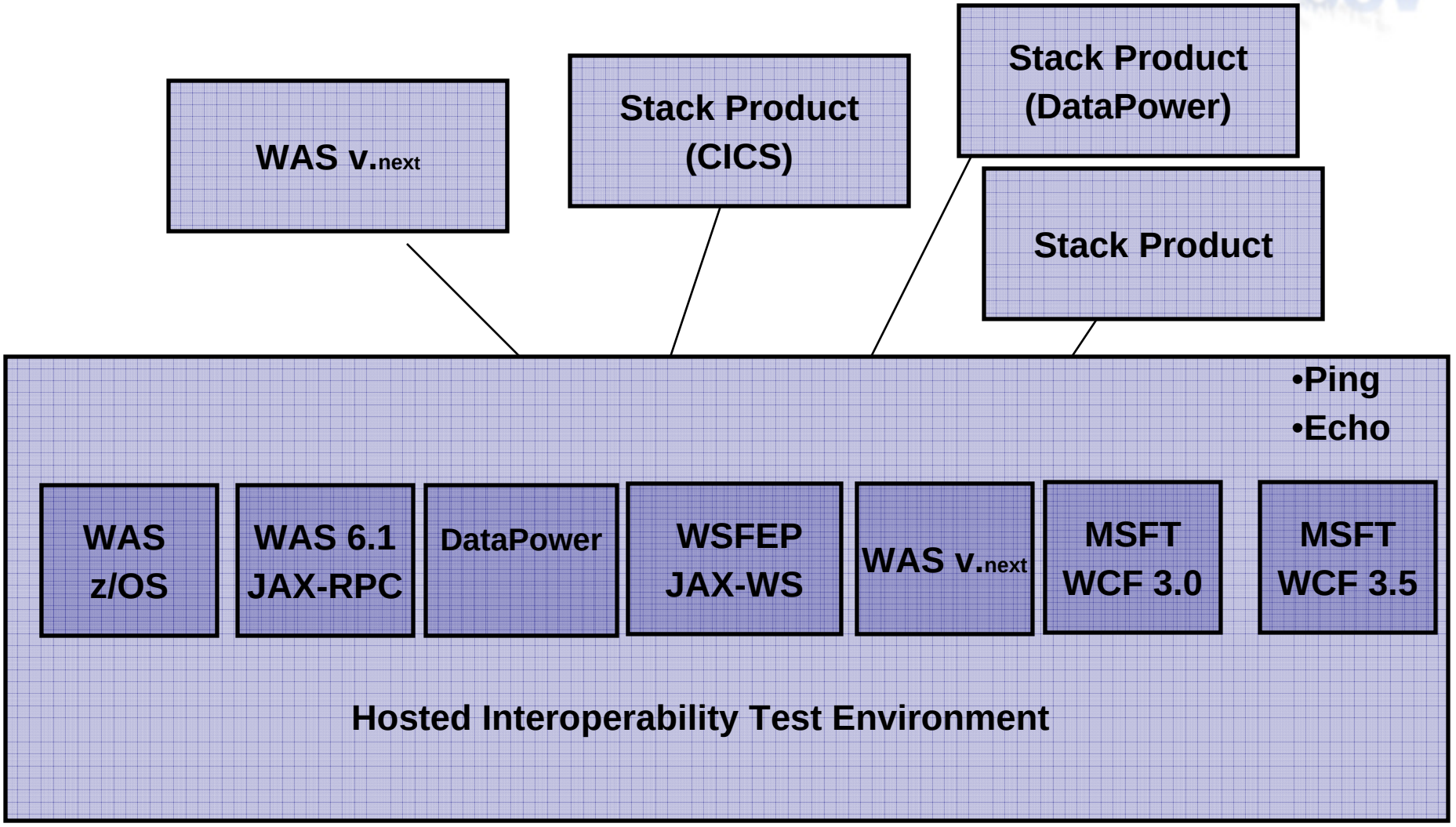
Internal & External testing

- Internal
  - Focused scenario testing with Microsoft WCF 3.0/3.5
  - Testing with IBM stack products (e.g. CICS)
  - Integration with DataPower
- External (testing with other vendors)
  - WS-I testing (Microsoft, BEA, Oracle, SAP, Sun, Novell)
  - Web Services Test Forum (WSTF)
  - Other standards based interoperability testing

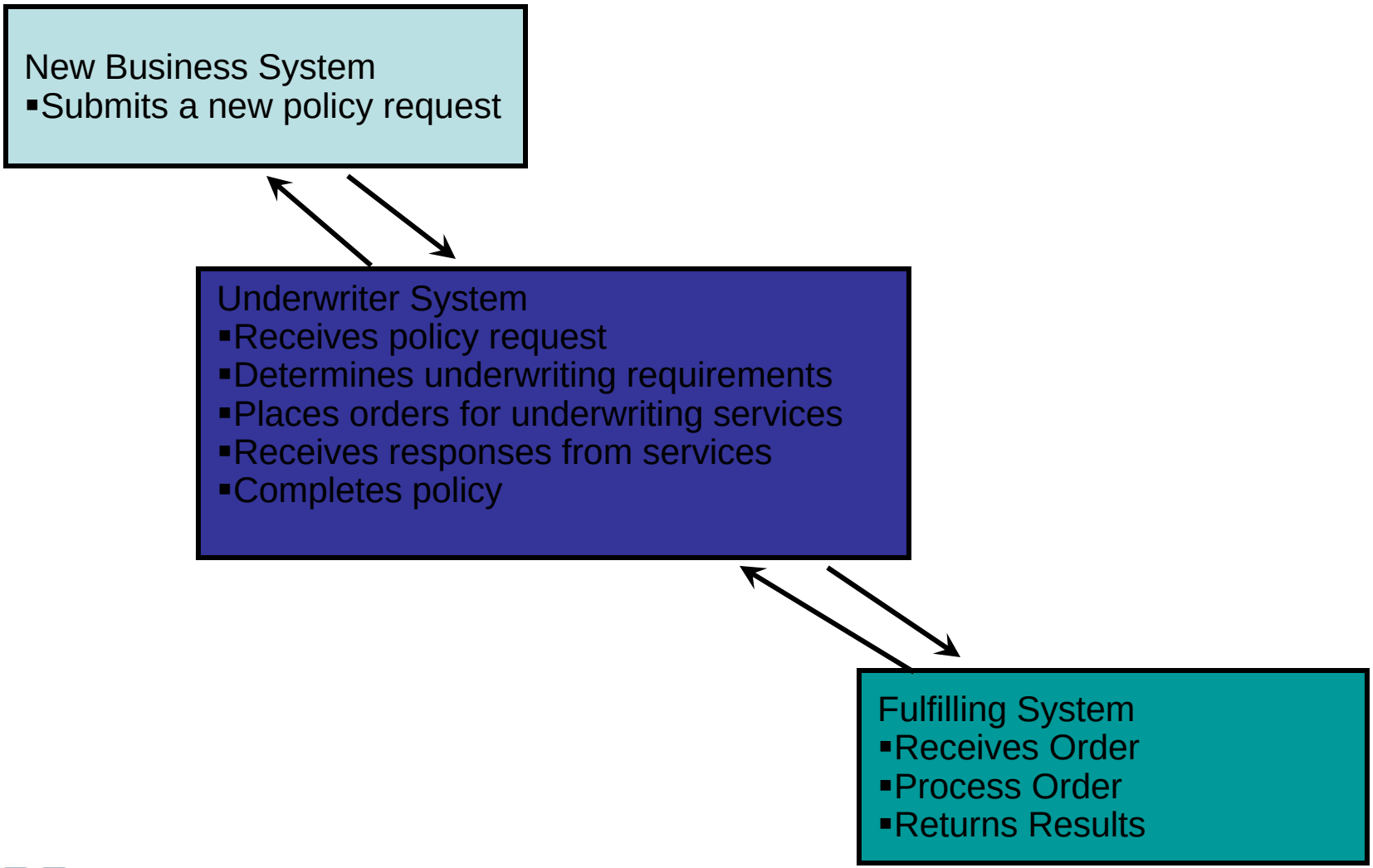
Simple Message Exchange Patterns

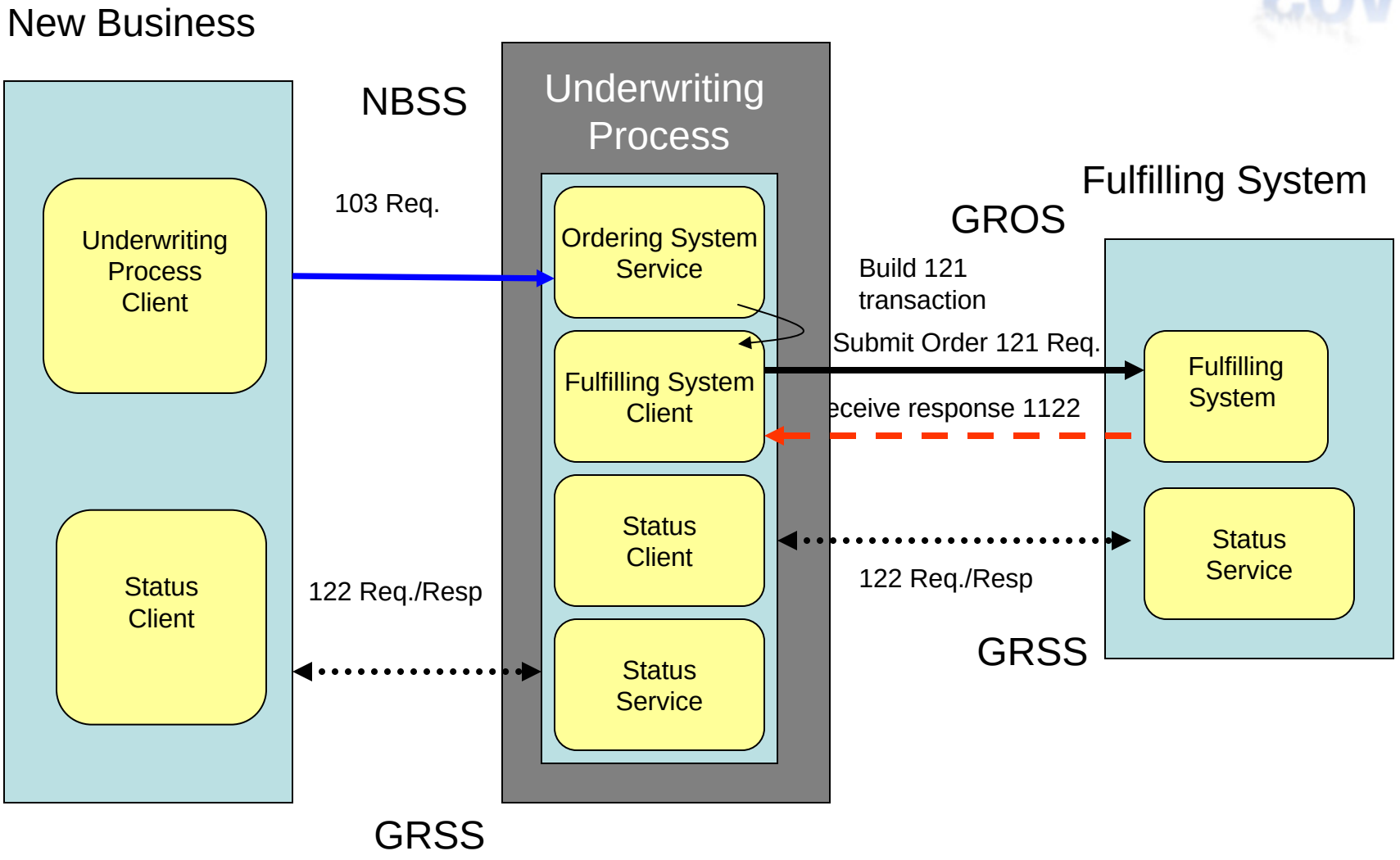


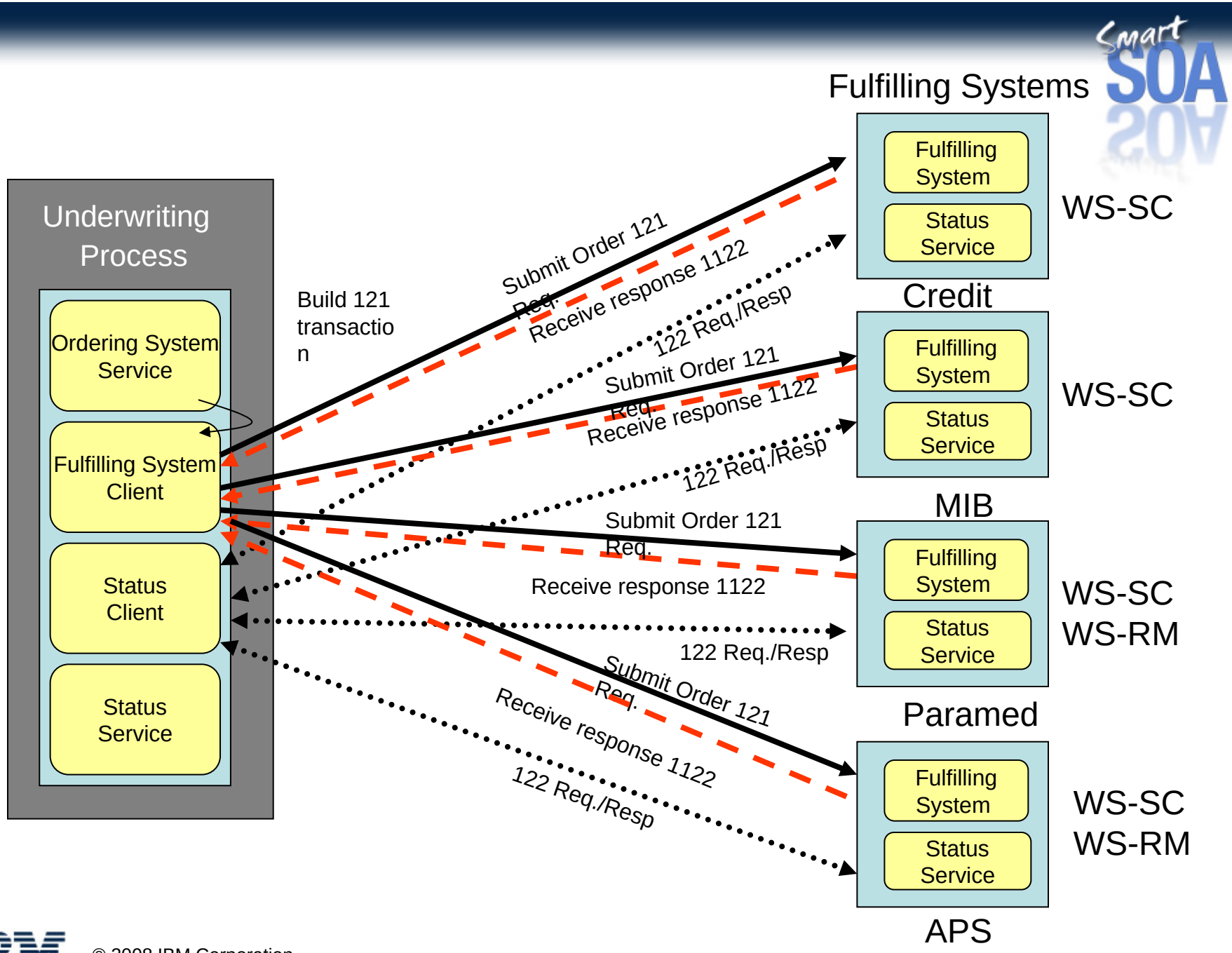
Internal Interoperability Test environment



ACORD Insurance Underwriting Scenario

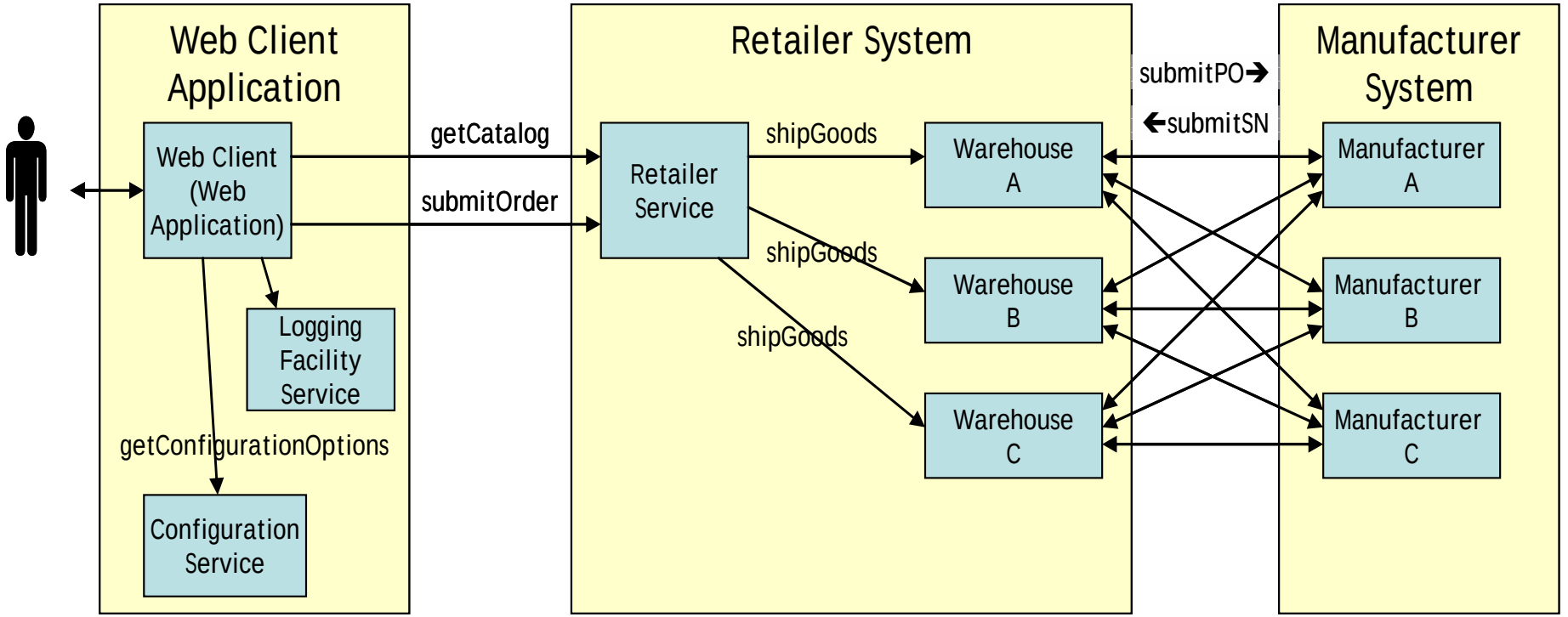






WS-I Secure Supply Chain Management Sample

Used to test BSP 1.0 interoperability for WAS 6.1



WAS/WCF Interoperability Test Scenarios

- SOAP 1.1/1.2
- WS-Security 1.0/1.1
- Username Token Profile
- Kerberos Token Profile
- X.509 Token Profile
- WS-Addressing
- WS-Addressing Metadata
- MTOM/XOP (for SOAP 1.1/1.2)
- WS-Reliable Messaging 1.0/1.1
- WS-SecureConversation 1.0 (submission spec and OASIS approved version)
- WS-Trust (submission spec and OASIS approved version)
- WS-Atomic Transaction (original AT support and OASIS approved version)
- WS-Policy (including OASIS WS-SX Security Policy, RM Policy, and TX policy)

Web Services Interoperability (WS-I)

What is WebSphere doing to make progress in WS-I ?

- Salim Zeitouni from the WebSphere Interoperability team now chairs the Sample Applications group
- Working closely with Microsoft to define the scenarios for BSP 1.1
- Charles LeVay from the WebSphere Interoperability team driving completion of test assertions, test scenarios, and test tools for RSP 1.0
- IBM Emerging technology group driving completion of test assertions, test scenarios and test tools for BP 1.2, BP 2.0

Web Services Test Forum (WSTF)

Problem Statement

- Web Services are becoming the dominant integration technology
  - “lingua franca” of SOA
  - underpin many enterprise architectures/roadmaps
- Most (all?) large enterprise environments are heterogeneous
- Unknown but significant risk around interoperability
  - WS interoperability largely untested except for simple compositions of technologies, e.g.:
    - SOAP 1.1
    - WS-Security
  - Projects may be delayed or fail due to unexpected interoperability issues

WSTF Details

- Almost everything is private (mailing lists, scenario work, votes...)
  - But low bar for entry – want people to discuss issues w/o fear of looking bad
- Unpublished scenarios can be tested, but nothing is made public
- Scenarios are made public only after a vote by the implementers
  - Requires at least 5 implementations + 2/3 ‘yes’ vote
  - Vote provides a simple filtering to prevent diluting the value of the group
    - Shows broad industry support
- Testing
  - Each scenario has a list of endpoints (private and public lists)
  - Endpoints are expected to be “long lived”
  - New implementations (non-members) can test at will using ‘published’ scenarios and endpoints
  - Implementers are encouraged to make their test code/config etc. available
  - Interoperability issues are brought to the appropriate forum(s) by individuals (not the group)

WSTF Status

- Charter and Participant's Agreement are now finalized
- Web site (http://www.wstf.org) is ready
- Founding members are working on initial set of scenarios
  - Very basic ones to test the infrastructure of the group
  - Provide a base-line for more advanced scenarios
- Soliciting next round of members

What Can Customers Do?

- Develop your own interoperability test lab
  - infeasible unless you can dedicate a large amount of resources
- Join the Web Services Test Forum (WSTF)
  - http://www.wstf.org/
  - open forum focused on developing and testing non-trivial WS scenarios
    - no dues, no board, minimum of process
    - consensus driven
    - scenario-based
  - provides an on-going, “shared” test bed available for the entire WS community to use

WSTF Involvement

- Observer
  - 1 (medium technical) person/day month
  - gain visibility into areas of biggest concern
  - scenarios may not match your business requirements
- Participant
  - 3 (highly technical) person/days month/scenario
  - same benefits as observer plus scenarios will more closely match your business requirements
- Evangelist
  - 5 (highly technical) person/days month
  - scenarios tailored to your business requirements
  - WSTF becomes your virtual interop lab

Best practices customers should know for implementing interoperable web services

High Level Best Practices *(top-down)*

- Start with Data Model and build schema
- Create Doc/Lit or Doc/Lit/wrapped WSDL
- Use tooling to validate WSDLs, schemas and instance documents compliant with WS-I Basic Profile
- Create stub service and code implementation
- Test application level interoperability

High Level Best Practices *(bottoms-up)*

- Utilize tools to generate artifacts (schemas, WSDLs)
- Use tooling to validate WSDLs, schemas and instance documents compliant with WS-I Basic Profile
- Deploy service test with clients generated from WSDLs
- Test application level interoperability

Meet in the middle – a combination and reality

- This is the most typical approach
  - top down design is difficult to implement
  - bottom up design generates ugly WSDL/XSD
  - bottom up design has language-specific bindings
  - bottom up implementation not Web service friendly
- Options:
  - Write custom serializers to map directly from XML to existing objects
    - Framework provided in WebSphere Application Server v6
    - JAX-WS Provider makes it easier to use other mapping capabilities
  - Write a facade service which maps from generated Java classes (mapped from XML) to existing code
  - If need to map from WSDL to existing XML, consider using an ESB
- Challenges
  - extra work to keep implementation and WSDL/XSD in sync

Top 10 tips for web services interoperability

- Avoid empty arrays
- Understand how data types map to package names
- Test for null using isNil instead of == null
- Avoid null dates altogether
- Be careful when comparing dates - use compareTo
- Use trace tools to log message exchanges
- Know how to adjust listening ports for your tools
- Use unit testing to ensure client and service handle edge cases correctly
- Use doc/literal per WS-I as the encoding type
- Think in terms of XSD, even if you write code-first

WebSphere / WCF interoperability

WCF Bindings

- basicHttpBinding
  - MEPs: One-way, two-way (anon),
  - SOAP 1.1, WS-Security (optional)
- wsHttpBinding
  - MEPs: One-way, two-way (anon),
  - SOAP 1.2, WS-Addressing, WS-Security (optional), WS-RM (optional), WS-SC (optional)
- wsDualHttpBinding
  - MEPs: two-way (addr)
  - SOAP 1.2, WS-Addressing, WS-Security (optional), WS-RM, WS-SC (optional)
- customBinding – EVERYTHING is Configurable !

SOAP 1.1 / SOAP 1.2 (Action)

- Scenario
  - WAS Client -> WCF Service
- Issue
  - For SOAP 1.1 WCF requires http header to contain SOAPAction
  - For SOAP 1.2 WCF requires action parameter in Content-Type to be defined
- Resolution
  - Define SOAP action in WSDL

    e.g. **<soap:operation soapAction="pingOperation" style="document" />**

  or

  - Define SOAP action in client code

```
// Configure SOAPAction properties
BindingProvider bp = (BindingProvider) (ping._getDescriptor()
                                        .getProxy());
bp.getRequestContext().put(BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
                                        endpointURL);
bp.getRequestContext().put(BindingProvider.SOAPACTION_USE_PROPERTY,
                                        Boolean.TRUE);
bp.getRequestContext().put(BindingProvider.SOAPACTION_URI_PROPERTY,
                                        "pingOperation");
```

MTOM / XOP

- Scenario
  - Attempting to send binary data (jpegs/images) via MTOM
- Issue
  - WCF defaults to limiting attachments to 64k in size
- Resolution
  - Increase default max receive message size for MSFT is 64k

WS-Addressing

- Scenario
  - WAS Client -> WCF Service
- Issue
  - WCF does not correctly handle a wsa:ReplyTo with a valid EPR unless the service is configured for compositeDuplex
- Resolution
  - Configure WCF with both compositeDuplex and oneWay stack elements
  - Recommend customBinding for this configuration

WS-Addressing

- Scenario
  - .Net Service Generation
- Issue
  - svcutil generates the wrong code for ReplyAction, it should NOT be ReplyAction="*" which means don't set any value
- Resolution
  - Either remove the ReplyAction definition altogether, or explicitly set it to the value you are expecting. Therefore, change

```
[System.ServiceModel.OperationContractAttribute( Action = "foo", ReplyAction = "*" )]
```

to

```
[System.ServiceModel.OperationContractAttribute( Action = "foo" )]
```

or

```
[System.ServiceModel.OperationContractAttribute( Action = "foo", ReplyAction = "bar" )]
```

X.509 Token Profile

- Scenario
  - WAS Client -> WCF Service, WCF Client -> WAS Service
- Issue
  - If DN contains multiple attributes
    e.g. cn=abc, ou = raleigh, o=ibm, c=us

  WAS removes the spaces, WCF doesn't like that.
- Resolution
  - Make sure your certificates have a key identifier defined
  - Use key identifier instead of X509issuer Token
  - To be fixed in a future release of WCF
    - http://www.codeprof.com/dev-archive/201/153-118-2012214.shtm

Username Token Profile

- Scenario
  - WAS Client -> WCF Service, WCF Client -> WAS Service
- Issue
  - When using WS-Security to secure the UNT, WCF defaults to a symmetric encryption algorithm (HMAC-SHA1) not supported in WAS. This algorithm is not configurable in the binding
- Resolution
  - Use SSL to secure UNT
  - We are trying to understand how to over-ride this in WCF code

WS-Secure Conversation

- Scenario
  - WAS Client -> WCF Service, WCF Client -> WAS Service
- Issue
  - Enabling signature confirmation in the bootstrap policy, WCF defaults to a symmetric encryption algorithm (HMAC-SHA1) not supported in WAS. This algorithm is not configurable in the binding
- Resolution
  - Don’t use signature confirmation
  - We are trying to understand how to over-ride this in WCF code

WS-Secure Conversation

- Scenario
  - WAS Client -> WCF Service, WCF Client -> WAS Service
- Issue
  - WAS and WCF use different default key sizes for signing
- Resolution
  - Configure key sizes to match

WS-Reliable Messaging (offer)

- Scenario
  - WAS Client -> WCF Service
- Issue
  - WCF Service requires the WAS client to provide an offer with a create sequence request for any 2-way MEP. According to the WS-RM specification, offer is OPTIONAL
- Resolution
  - For WS-RM 1.0, use the WS-ReliableMessaging 1.0 Policy Set
  - For WS-RM 1.1, use the WS-RM SPIs to enable the WAS client to send an offer

WS-Reliable Messaging & WS- Make Connection

- Scenario
  - WAS Client -> WCF Service, WCF Client -> WAS Service
- Issue
  - WCF does not support WS-MakeConnection (WS-MC). WS-MC is used to re-initiate a broken sequence for a 2-way (anonymous) MEP. WCF supports a proprietary message replay scheme.
- Resolution
  - Use WS-ReliableMessaging 1.0 Policy Set which disables MakeConnection and implements replay for interoperability with WCF
  - Use WS-ReliableMessaging Default Policy Set (RM 1.1), but do not use two-way (anonymous) MEPs – remember to use WS-RM SPIs to enable the WAS client to send an offer for two-way (valid address) MEPs

WS-Reliable Messaging (Sequence expiration)

- Scenario
  - WAS Client -> WCF Service
- Issue
  - WAS Client throws exception when it receives an unknown sequence fault due to sequence expiration due inactivity
- Resolution
  - Catch exception and use WS-RM SPIs to send a Create Sequence Request, don't forget to send an offer for two-way MEPs
  - Increase the sequence inactivity timeout in WCF service (default 10 minutes). FYI - WAS sequence inactivity timeout is 24 hours.
  - Working on adding feature in WAS to handle this situation without client code.

WS-ReliableMessaging & (WS-Sec. or WS-SC)

- Scenario
  - WAS Client -> WCF Service, WCF Client -> WAS Service
- Issue
  - By default, WCF does not secure WS-RM control messages e.g. CreateSequeunce. WAS applies the application level security policy to the WS-RM control messages
- Resolution
  - Add and additional behavior to the WCF Client and Service endpoints to encrypt the WS-RM control messages .
    - http://forums.microsoft.com/MSDN/ShowPost.aspx?PostID=913081&SiteID=1

WS-AtomicTransaction

- Scenario
  - WAS Client -> WCF Service
- Issue
  - WS-Coordination message hangs on WCF service due missing WS-Addressing RelatesTo in the WAS message. This is a WAS bug.
- Resolution
  - Resolved by applying an iFix. The iFix will be made available shortly.

Additional Resources

- developerWorks articles

  http://www.ibm.com/developerworks/websphere/library/techarticles/0710_levay/0710_levay.html
  http://www.ibm.com/developerworks/websphere/library/techarticles/0712_levay/0712_levay.html
  http://www.ibm.com/developerworks/websphere/library/techarticles/0711_zeitouni/0711_zeitouni.html
  http://www.ibm.com/developerworks/websphere/library/techarticles/0801_zeitouni/0801_zeitouni.html

- Interoperability Organizations

  http://www.ws-i.org/

  WS-I
  WEB SERVICES
  INTEROPERABILITY
  ORGANIZATION

  http://www.wstf.org/

  WSTF

Other comments

Questions to ask about standards

- Does it address a requirement you have?
- Do you have requirements not covered by the standard, but covered by proprietary alternatives or extensions?
- Do you require interoperability or portability across platforms?
- Is it a standard or a specification? If a specification or draft standard how mature?
- Is it supported on all of your vendors platforms? By some of them? At a consistent version level?
- Is the standard covered by a WS-I profile? Do vendors support the profile?
- Do you need to adopt the latest version of the standard?
- If the vendor support doesn't exist for the standard and exists for an earlier specification, what would it take to migrate?
- Can you put a layer in front of the specification/standard APIs to protect your code from API changes?
- If the alternative is to develop your own equivalent to the standard/spec, don't – usually you are better off using the standard/spec
- Is the standard/spec more complex than what you need?

Questions?

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.
Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM trademarks, see www.ibm.com/legal/copytrade.shtml
AIX, CICS, CICSPlex, DB2, DB2 Universal Database, i5/OS, IBM, the IBM logo, IMS, iSeries, Lotus, OMEGAMON, OS/390, Parallel Sysplex, pureXML, Rational, RCAF, Redbooks, Sametime, Smart SOA, System i, System i5, System z , Tivoli, WebSphere, and z/OS.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
Intel and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.