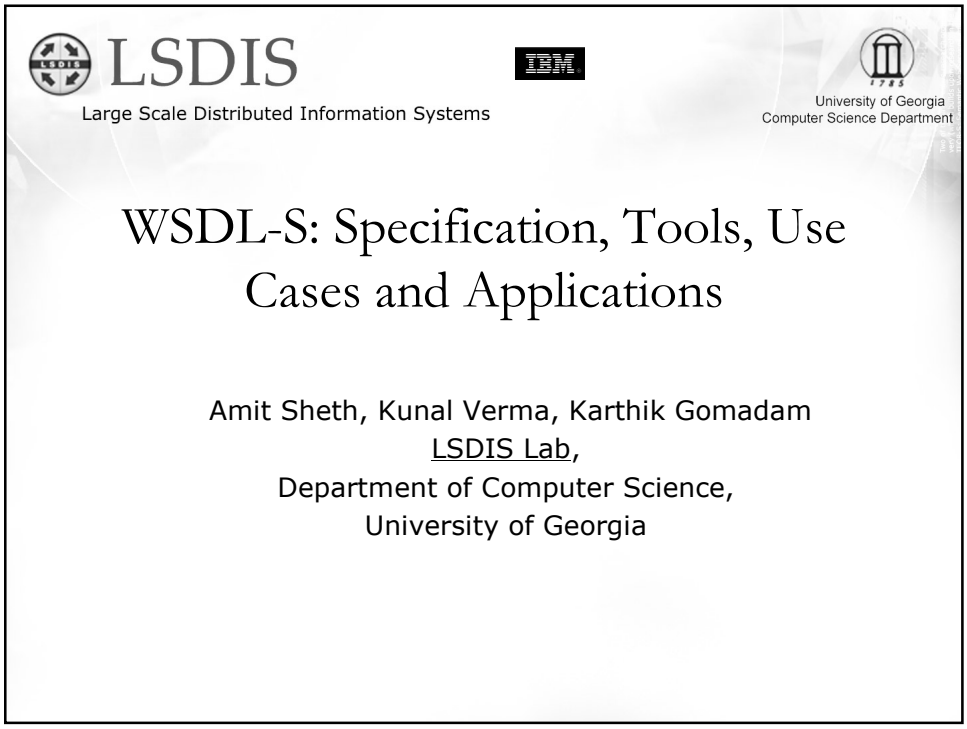
3-2006

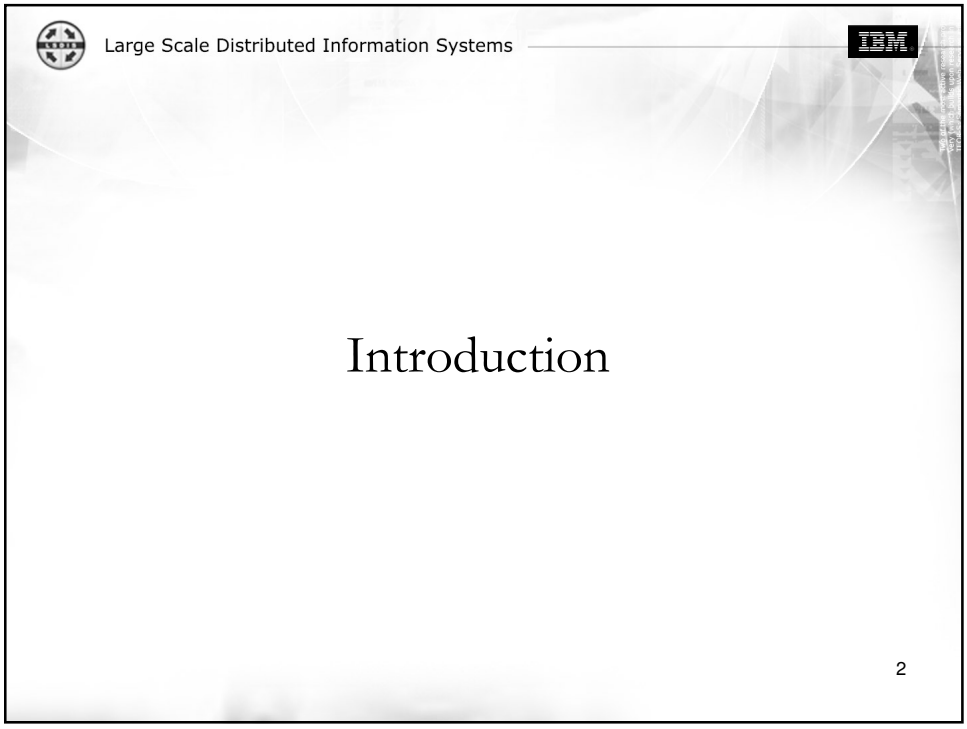
WSDL-S: Specification, Tools, Use Cases and Applications

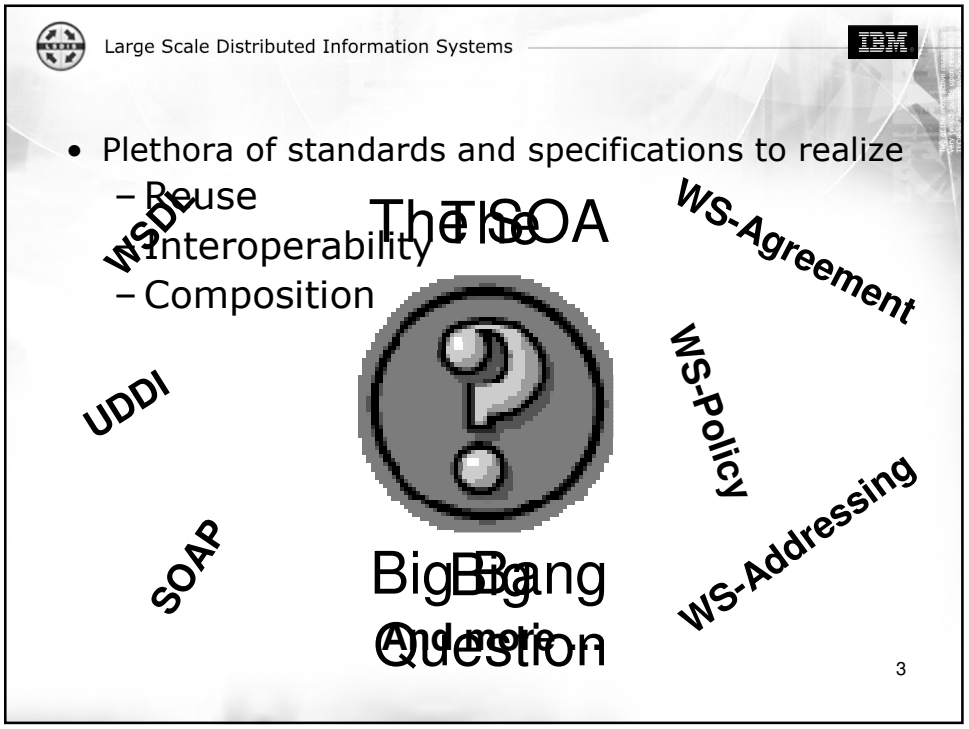
Amit P. Sheth
Wright State University - Main Campus, amit@sc.edu

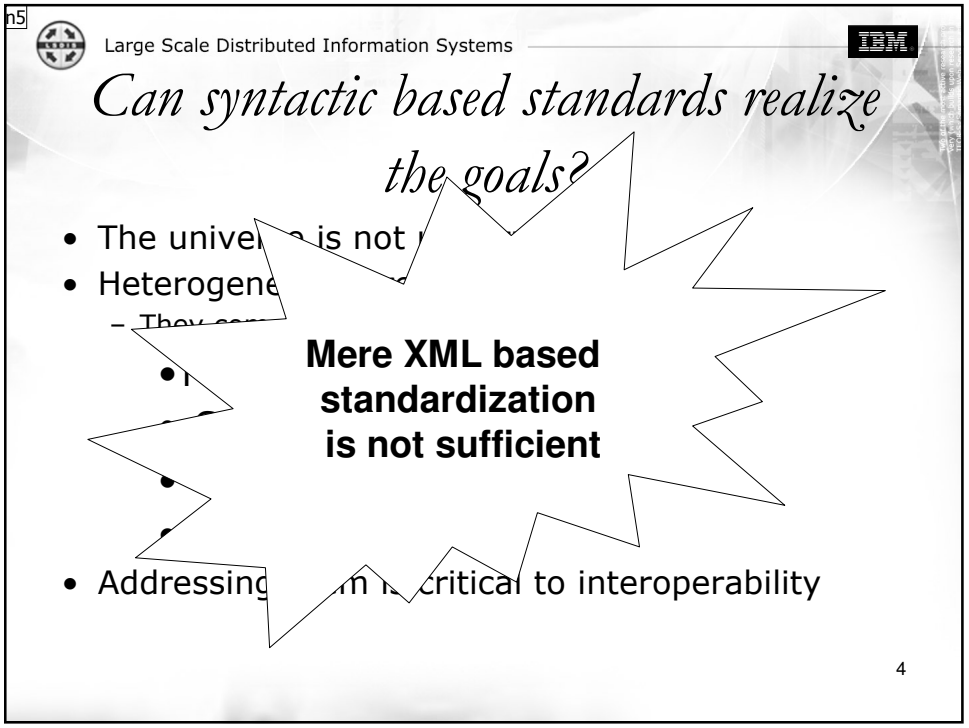
Kunal Verma
Wright State University - Main Campus

Karthik Gomadam
Wright State University - Main Campus


LSDIS
Large Scale Distributed Information Systems
IBM
University of Georgia
Computer Science Department
WSDL-S: Specification, Tools, Use Cases and Applications
Amit Sheth, Kunal Verma, Karthik Gomadam
LSDIS Lab,
Department of Computer Science,
University of Georgia

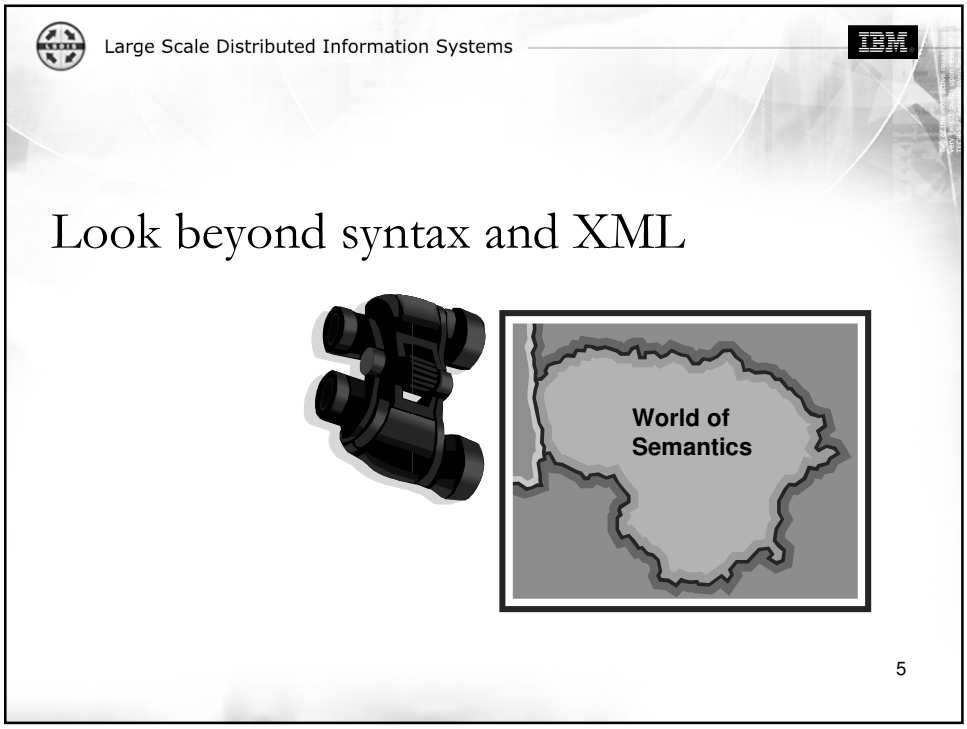

Large Scale Distributed Information Systems
IBM
Introduction
2

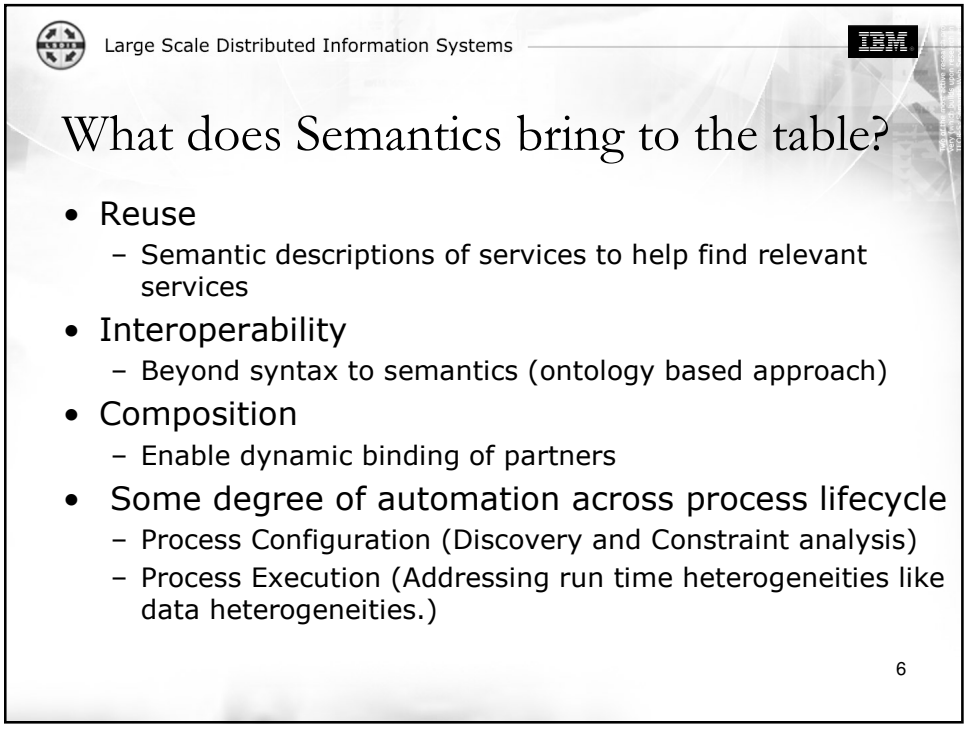

Large Scale Distributed Information Systems
IBM
• Plethora of standards and specifications to realize
– Reuse
– Interoperability
– Composition
WSDL
The Issue
WS-Agreement
UDDI
WS-Policy
SOAP
Big Bang
And more...
WS-Addressing
The SOA Question
3


Large Scale Distributed Information Systems
IBM
Can syntactic based standards realize the goals?
• The univer is not
• Heterogene
Mere XML based standardization is not sufficient
• Addressing is critical to interoperability
4

Slide 4

n5 Process lifecycle semantics can help with some degree of automation in all the steps
1. Configuration and execution
namelessnerd, 2/1/2006


Large Scale Distributed Information Systems
IBM
Look beyond syntax and XML
World of Semantics
5


Large Scale Distributed Information Systems
IBM
What does Semantics bring to the table?
• Reuse
– Semantic descriptions of services to help find relevant services
• Interoperability
– Beyond syntax to semantics (ontology based approach)
• Composition
– Enable dynamic binding of partners
• Some degree of automation across process lifecycle
– Process Configuration (Discovery and Constraint analysis)
– Process Execution (Addressing run time heterogeneities like data heterogeneities.)
6

n7

Large Scale Distributed Information Systems

IBM

Juxtaposing the various approaches to SWS

Specification	Description	Formalism
OWL-S	An OWL based upper ontology semantically representing Web services.	Description Logics
SWSL (Semantic Web Service Language)	A combination First Order Logic and rules to represent Web services.	First order logic, Different variants of rule languages (Horn, HiLog etc.)
WSDL-S	Use of extensibility elements in WSDL to annotate elements with terms in ontologies.	Agnostic (examples typically use description logics, but use of UML for conceptual modeling is also recognized)
WSMO	A F-Logic based conceptual model for representing Web services.	F-Logic

Each of these are W3C Member submissions

7

Large Scale Distributed Information Systems

IBM

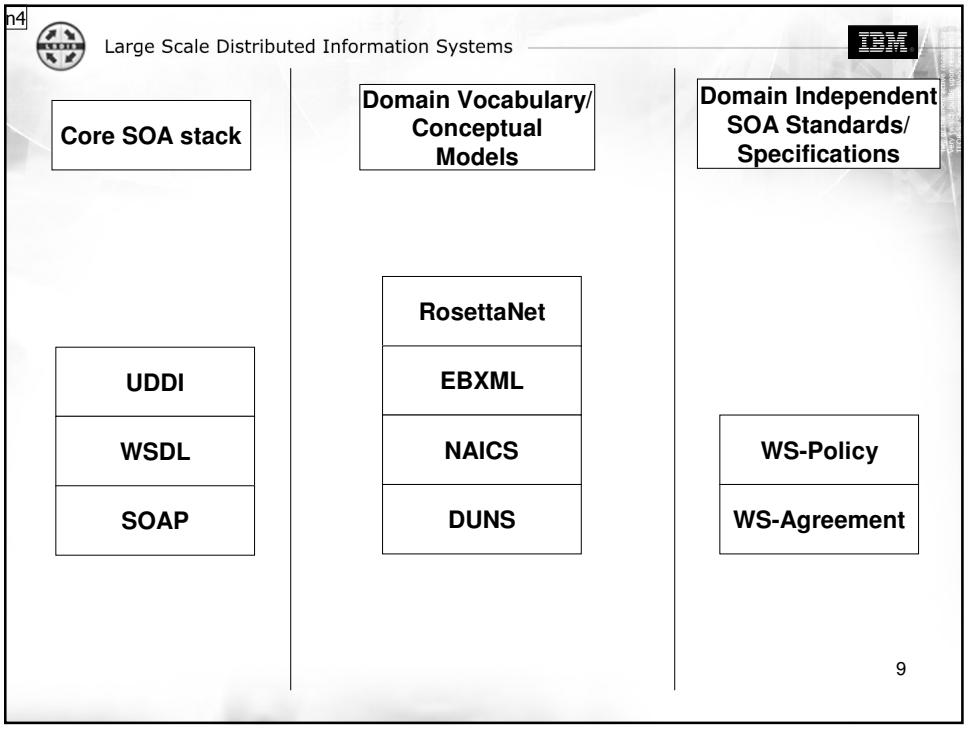
Semantics to Web Services: The ingredients

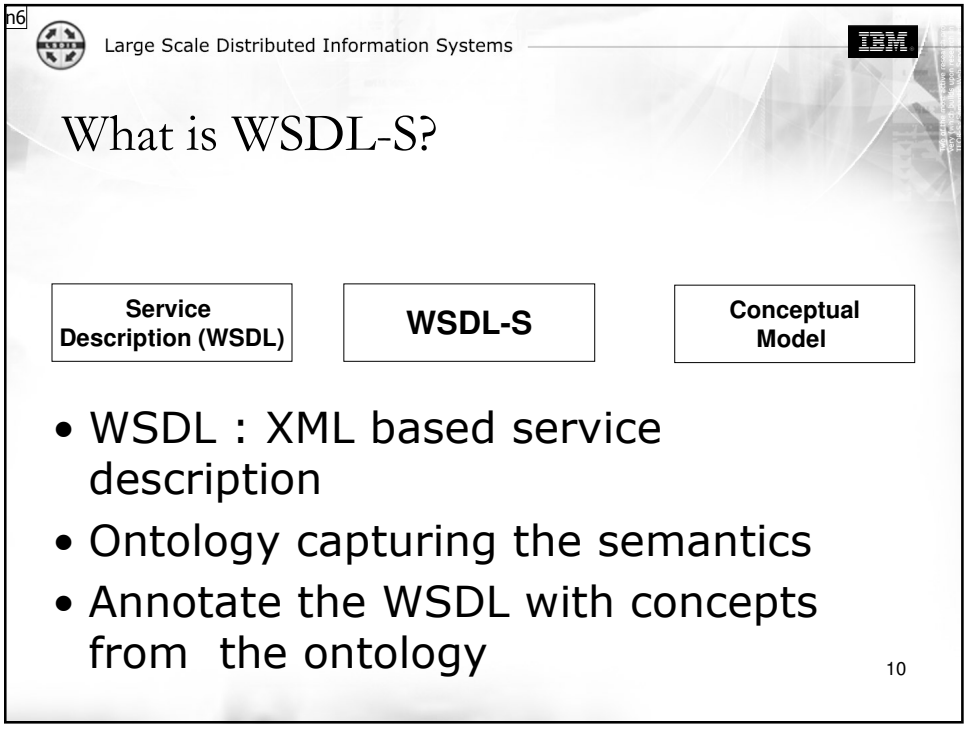
- Conceptual Model
  - An agreed upon model that captures the semantics of domain.
- XML based service description
  - Standards and specifications like WSDL for web service description, WS-Agreement for capturing agreements etc.
- Annotate the service description

8

Slide 7

n7 What are the common ideas across them?
namelessnerd, 2/3/2006


n4
Large Scale Distributed Information Systems
IBM
Core SOA stack
Domain Vocabulary/ Conceptual Models
Domain Independent SOA Standards/ Specifications
UDDI
WSDL
SOAP
RosettaNet
EBXML
NAICS
DUNS
WS-Policy
WS-Agreement
9

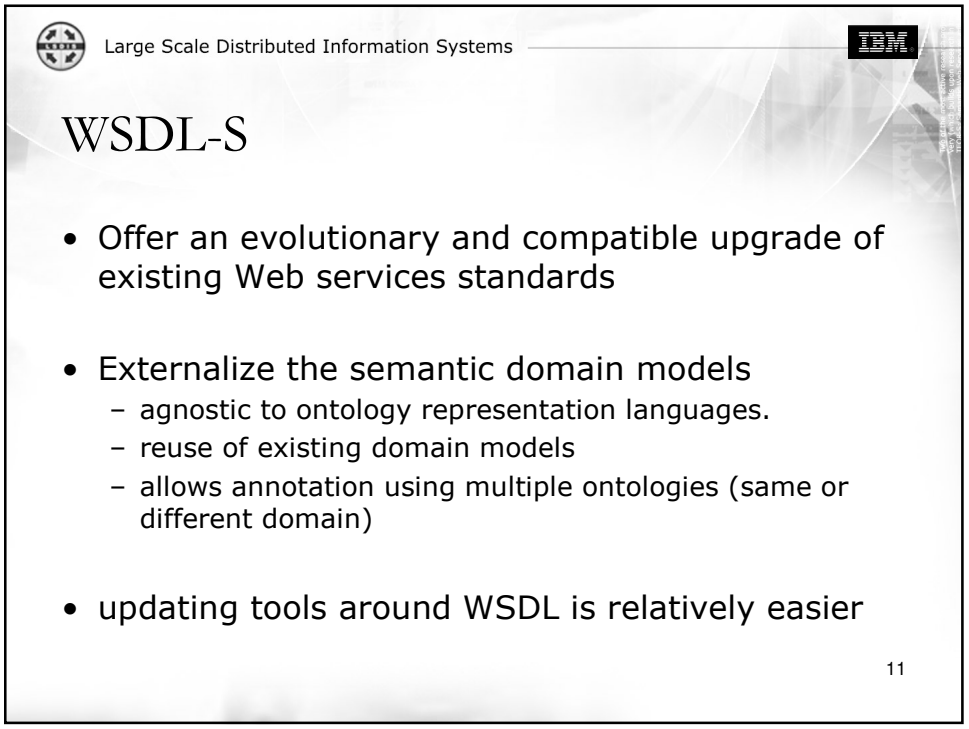

n6
Large Scale Distributed Information Systems
IBM
What is WSDL-S?
Service Description (WSDL)
WSDL-S
Conceptual Model
• WSDL : XML based service description
• Ontology capturing the semantics
• Annotate the WSDL with concepts from the ontology
10

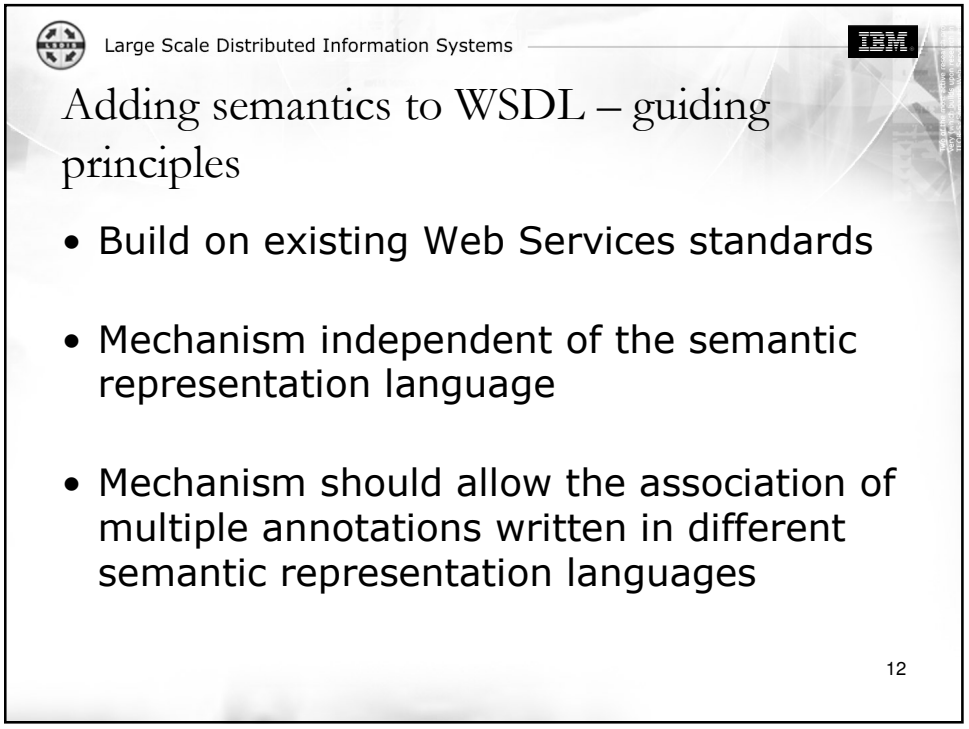
Slide 9

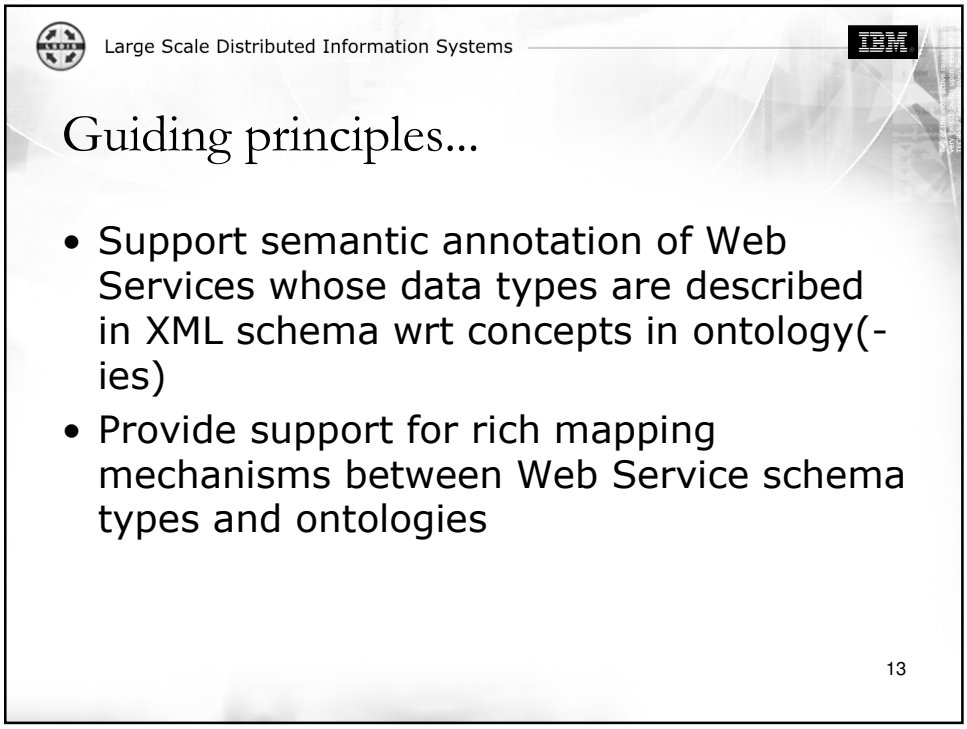
n4 supports higher degree of automation
namelessnerd, 2/1/2006

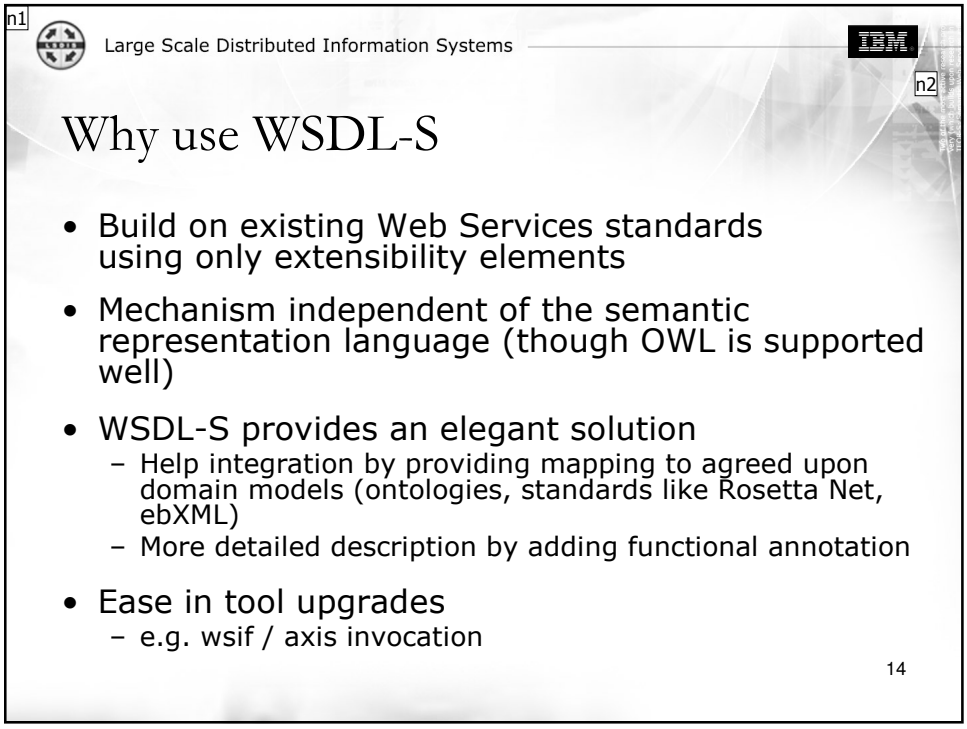
Slide 10

n6 structural semantic interop.
namelessnerd, 2/1/2006


Large Scale Distributed Information Systems
IBM
WSDL-S
• Offer an evolutionary and compatible upgrade of existing Web services standards
• Externalize the semantic domain models
– agnostic to ontology representation languages.
– reuse of existing domain models
– allows annotation using multiple ontologies (same or different domain)
• updating tools around WSDL is relatively easier
11


Large Scale Distributed Information Systems
IBM
Adding semantics to WSDL – guiding principles
• Build on existing Web Services standards
• Mechanism independent of the semantic representation language
• Mechanism should allow the association of multiple annotations written in different semantic representation languages
12

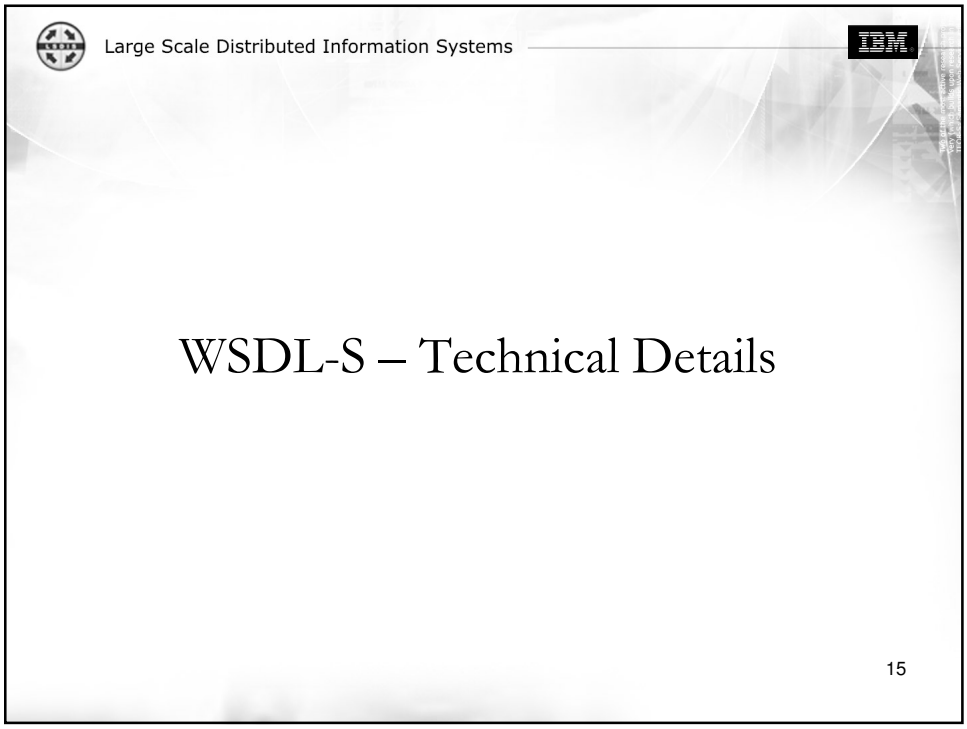

Large Scale Distributed Information Systems
IBM
Guiding principles...
• Support semantic annotation of Web Services whose data types are described in XML schema wrt concepts in ontology(-ies)
• Provide support for rich mapping mechanisms between Web Service schema types and ontologies
13

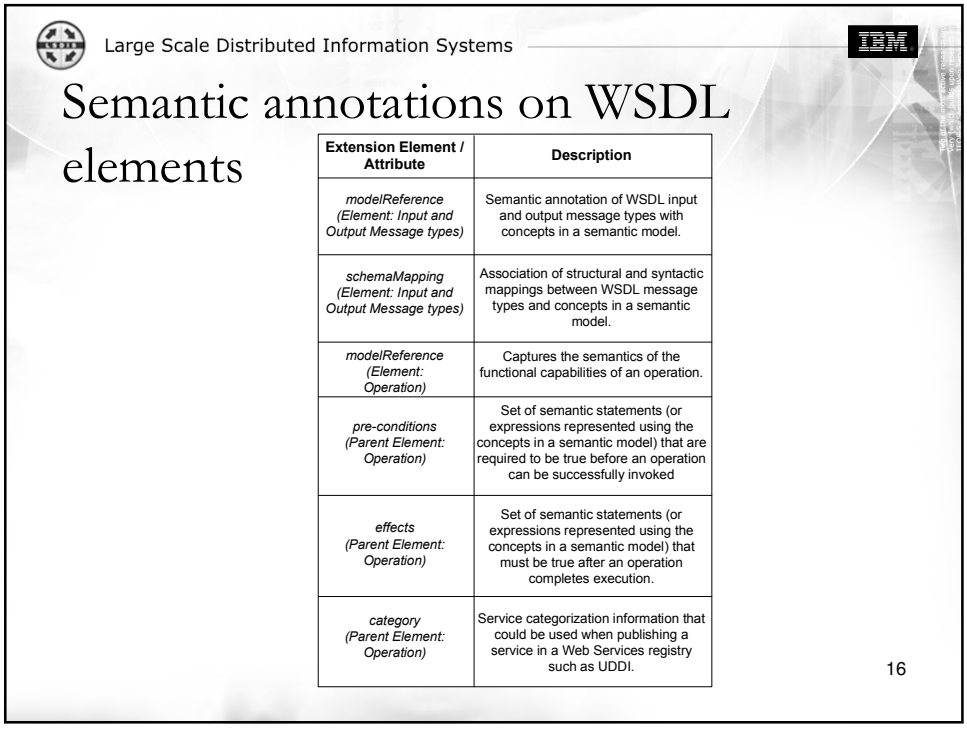

n1
Large Scale Distributed Information Systems
IBM
n2
Why use WSDL-S
• Build on existing Web Services standards using only extensibility elements
• Mechanism independent of the semantic representation language (though OWL is supported well)
• WSDL-S provides an elegant solution
– Help integration by providing mapping to agreed upon domain models (ontologies, standards like Rosetta Net, ebXML)
– More detailed description by adding functional annotation
• Ease in tool upgrades
– e.g. wsif / axis invocation
14

Slide 14

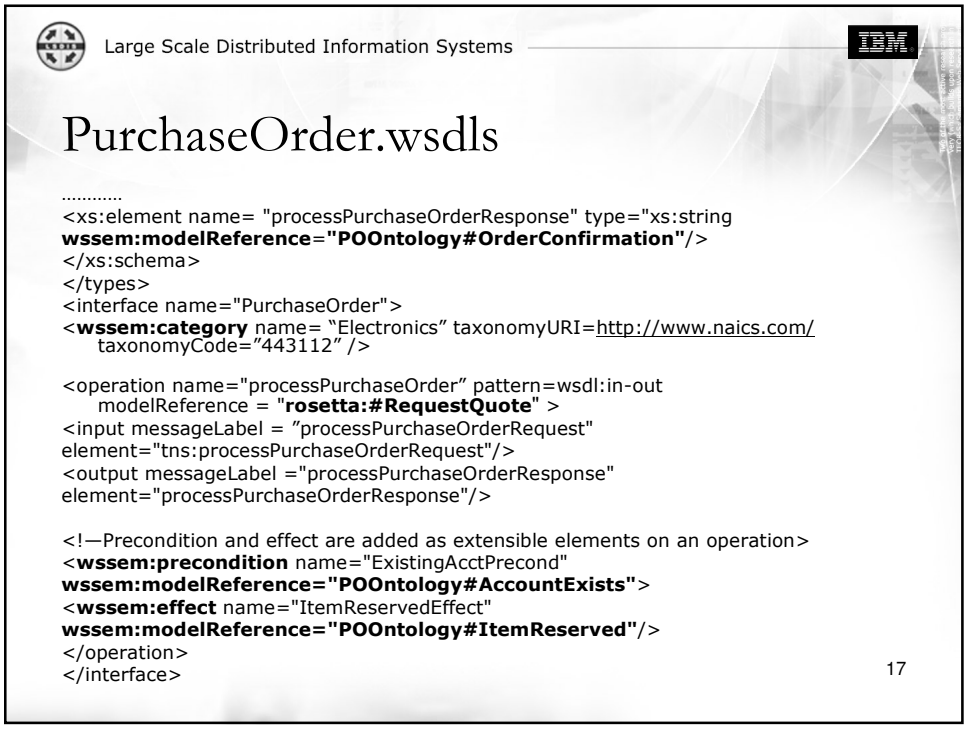
n1 add the four key aspects from CACM paper.
namelessnerd, 1/31/2006

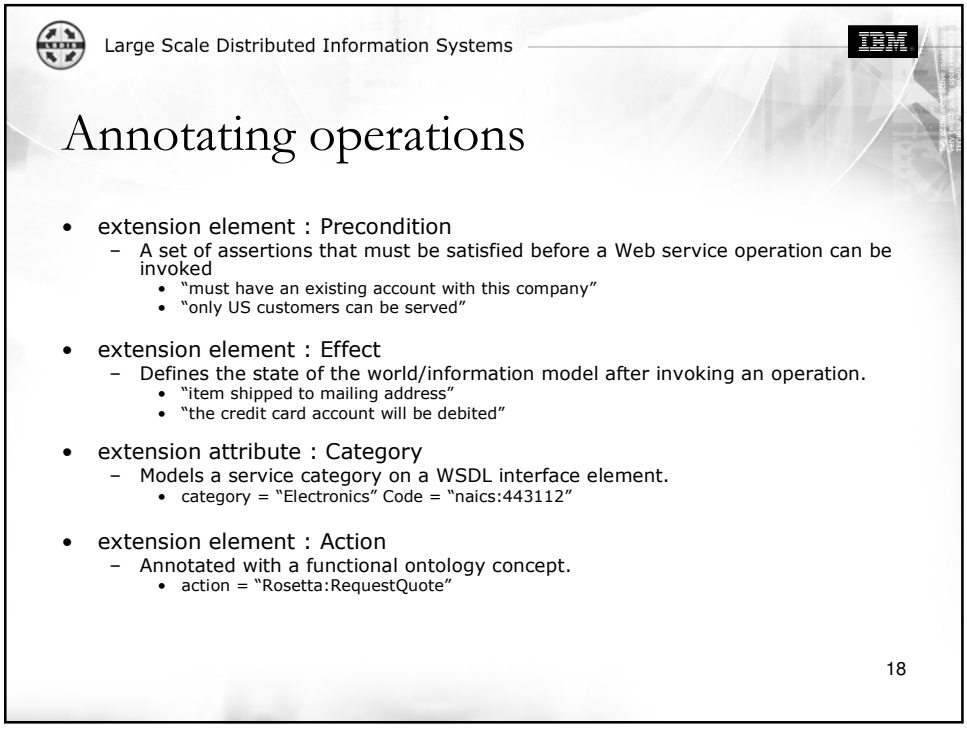
n2 namelessnerd, 1/31/2006





Extension Element / Attribute	Description
modelReference (Element: Input and Output Message types)	Semantic annotation of WSDL input and output message types with concepts in a semantic model.
schemaMapping (Element: Input and Output Message types)	Association of structural and syntactic mappings between WSDL message types and concepts in a semantic model.
modelReference (Element: Operation)	Captures the semantics of the functional capabilities of an operation.
pre-conditions (Parent Element: Operation)	Set of semantic statements (or expressions represented using the concepts in a semantic model) that are required to be true before an operation can be successfully invoked
effects (Parent Element: Operation)	Set of semantic statements (or expressions represented using the concepts in a semantic model) that must be true after an operation completes execution.
category (Parent Element: Operation)	Service categorization information that could be used when publishing a service in a Web Services registry such as UDDI.

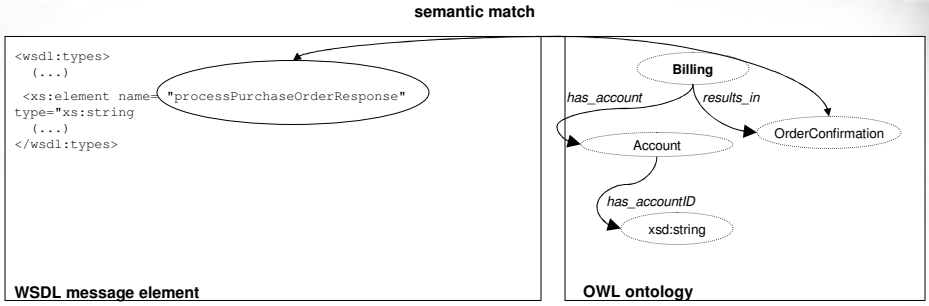

Large Scale Distributed Information Systems
IBM
PurchaseOrder.wsdls
…….
<xs:element name= "processPurchaseOrderResponse" type="xs:string
wssem:modelReference="POOntology#OrderConfirmation"/>
</xs:schema>
</types>
<interface name="PurchaseOrder">
<wssem:category name= "Electronics" taxonomyURI=http://www.naics.com/
taxonomyCode="443112" />
<operation name="processPurchaseOrder" pattern=wsdl:in-out
modelReference = "rosetta:#RequestQuote" >
<input messageLabel = "processPurchaseOrderRequest"
element="tns:processPurchaseOrderRequest"/>
<output messageLabel ="processPurchaseOrderResponse"
element="processPurchaseOrderResponse"/>
<!—Precondition and effect are added as extensible elements on an operation>
<wssem:precondition name="ExistingAcctPrecond"
wssem:modelReference="POOntology#AccountExists">
<wssem:effect name="ItemReservedEffect"
wssem:modelReference="POOntology#ItemReserved"/>
</operation>
</interface>
17


Large Scale Distributed Information Systems
IBM
Annotating operations
extension element : Precondition
A set of assertions that must be satisfied before a Web service operation can be invoked
"must have an existing account with this company"
"only US customers can be served"
extension element : Effect
Defines the state of the world/information model after invoking an operation.
"item shipped to mailing address"
"the credit card account will be debited"
extension attribute : Category
Models a service category on a WSDL interface element.
category = "Electronics" Code = "naics:443112"
extension element : Action
Annotated with a functional ontology concept.
action = "Rosetta:RequestQuote"
18

Large Scale Distributed Information Systems

IBM

Annotating message types - simple correspondences



1:1 Correspondences

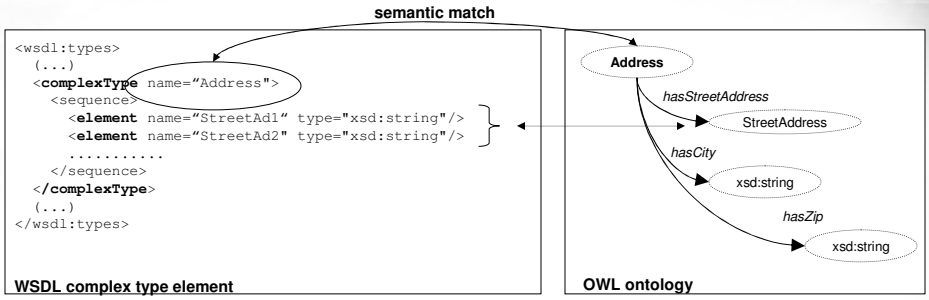
```
<xs:element name= "processPurchaseOrderResponse"
    type="xs:string
    wssem:modelReference="POOntology#OrderConfirmation"/>
```

19

Large Scale Distributed Information Systems

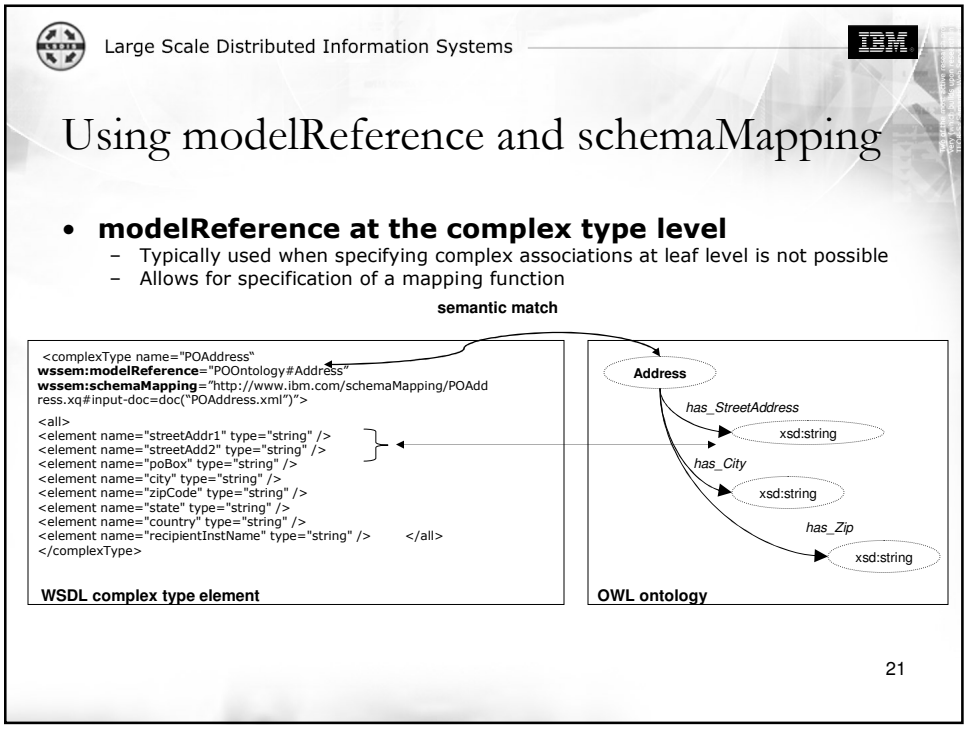
IBM

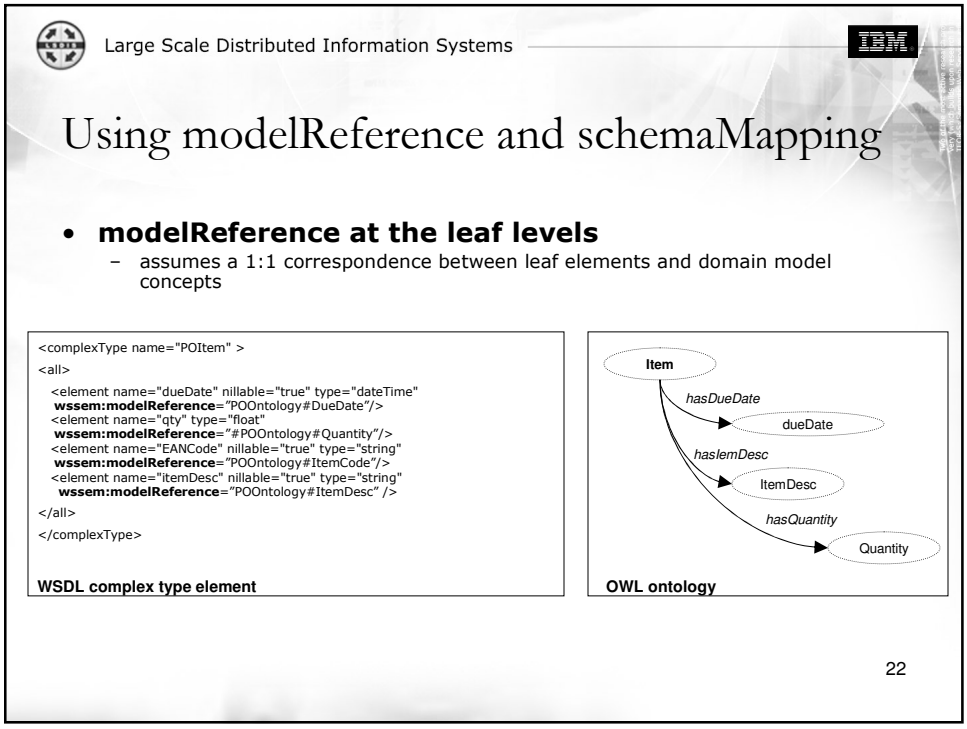
Annotating message types - complex correspondences

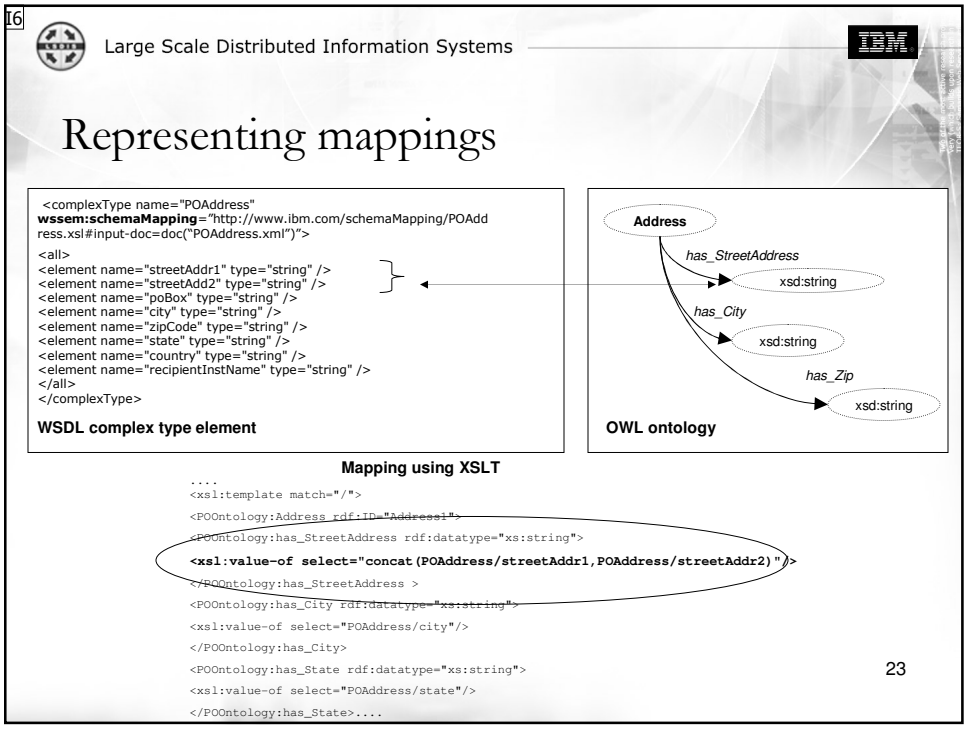


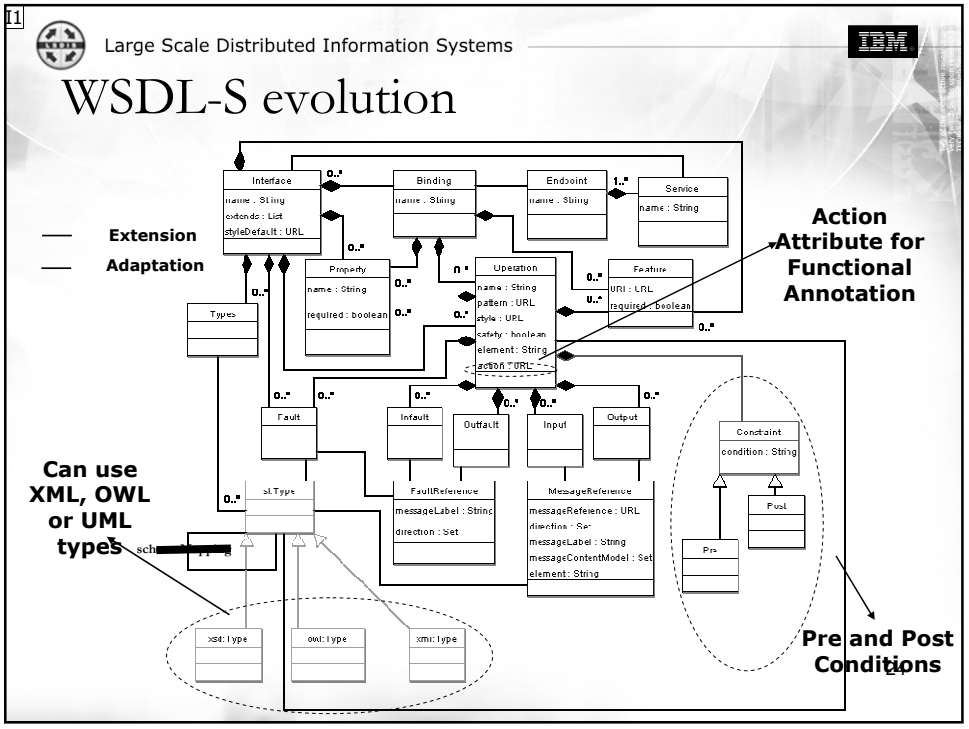
1. modelReference to establish a semantic association
2. schemaMapping to resolve structural heterogeneities beyond a semantic match

20


Large Scale Distributed Information Systems
IBM
Using modelReference and schemaMapping
• modelReference at the complex type level
– Typically used when specifying complex associations at leaf level is not possible
– Allows for specification of a mapping function
semantic match
<complexType name="POAddress"
wssem:modelReference="POOntology#Address"
wssem:schemaMapping="http://www.ibm.com/schemaMapping/POAddress.xq#input-doc=doc("POAddress.xml")">
<all>
<element name="streetAddr1" type="string" />
<element name="streetAdd2" type="string" />
<element name="poBox" type="string" />
<element name="city" type="string" />
<element name="zipCode" type="string" />
<element name="state" type="string" />
<element name="country" type="string" />
<element name="recipientInstName" type="string" /> </all>
</complexType>
WSDL complex type element
Address
has_StreetAddress
xsd:string
has_City
xsd:string
has_Zip
xsd:string
OWL ontology
21


Large Scale Distributed Information Systems
IBM
Using modelReference and schemaMapping
• modelReference at the leaf levels
– assumes a 1:1 correspondence between leaf elements and domain model concepts
<complexType name="POItem" >
<all>
<element name="dueDate" nillable="true" type="dateTime"
wssem:modelReference="POOntology#DueDate"/>
<element name="qty" type="float"
wssem:modelReference="#POOntology#Quantity"/>
<element name="EANCode" nillable="true" type="string"
wssem:modelReference="POOntology#ItemCode"/>
<element name="itemDesc" nillable="true" type="string"
wssem:modelReference="POOntology#ItemDesc" />
</all>
</complexType>
WSDL complex type element
Item
hasDueDate
dueDate
hasItemDesc
ItemDesc
hasQuantity
Quantity
OWL ontology
22


16
Large Scale Distributed Information Systems
IBM
Representing mappings
<complexType name="POAddress"
wssem:schemaMapping="http://www.ibm.com/schemaMapping/POAddress.xsl#input-doc=doc("POAddress.xml")">
<all>
<element name="streetAddr1" type="string" />
<element name="streetAdd2" type="string" />
<element name="poBox" type="string" />
<element name="city" type="string" />
<element name="zipCode" type="string" />
<element name="state" type="string" />
<element name="country" type="string" />
<element name="recipientInstName" type="string" />
</all>
</complexType>
WSDL complex type element
Address
has_StreetAddress
xsd:string
has_City
xsd:string
has_Zip
xsd:string
OWL ontology
Mapping using XSLT
....
<xsl:template match="/">
<POOntology:Address rdf:ID="Address1">
<POOntology:has_StreetAddress rdf:datatype="xs:string">
<xsl:value-of select="concat(POAddress/streetAddr1,POAddress/streetAddr2)"/>
</POOntology:has_StreetAddress >
<POOntology:has_City rdf:datatype="xs:string">
<xsl:value-of select="POAddress/city"/>
</POOntology:has_City>
<POOntology:has_State rdf:datatype="xs:string">
<xsl:value-of select="POAddress/state"/>
</POOntology:has_State>....
23

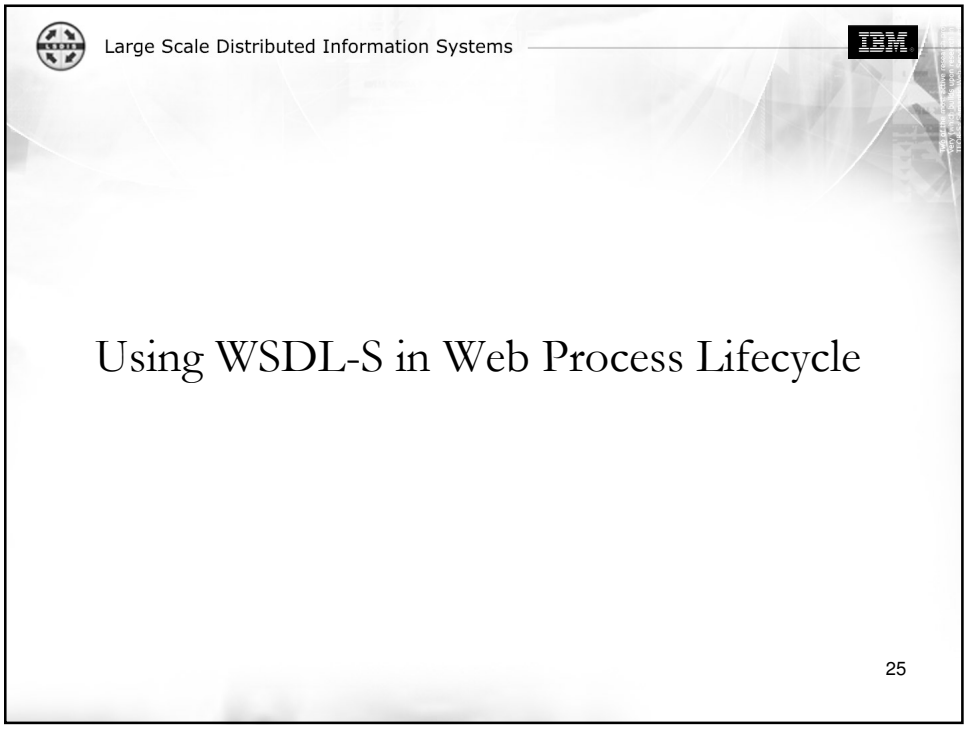

11
Large Scale Distributed Information Systems
IBM
WSDL-S evolution
Extension
Adaptation
Action Attribute for Functional Annotation
Can use XML, OWL or UML types
Pre and Post Conditions
Interface
Binding
Endpoint
Service
Property
Operation
Types
Fault
Infault
Outfault
Input
Output
Constraint
Pre
Post
xsd:Type
xsd: type
owl: type
xml: type
24

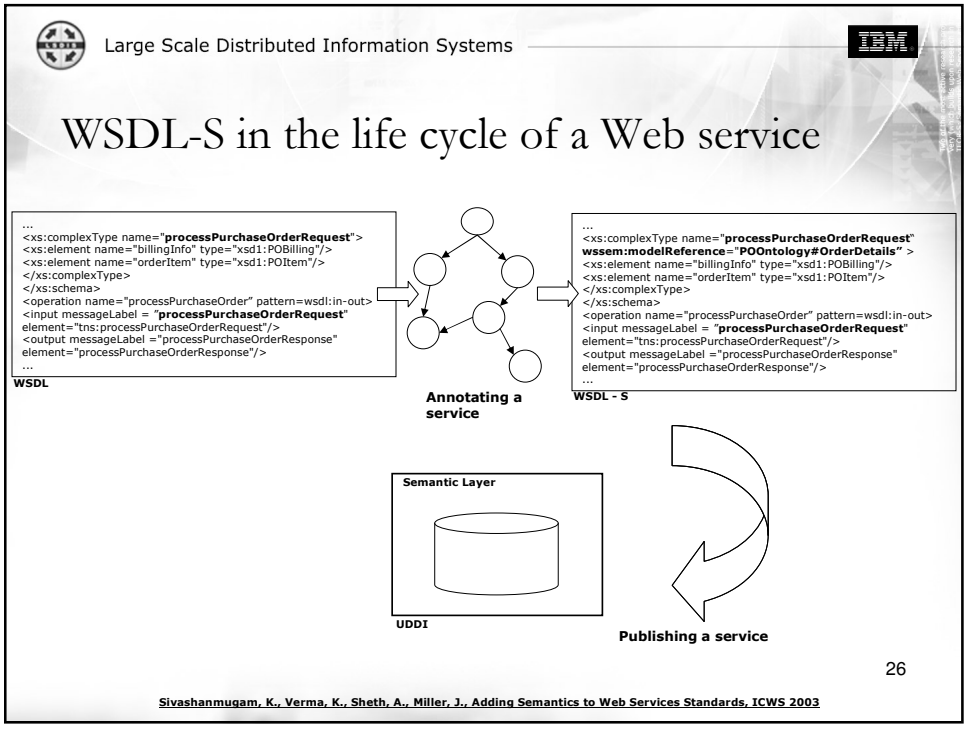
Slide 23

I6 <Rama> I recommend using XSLT, which is a bit more readable and less overwhelming than XQuery functions.
</Rama>
IBM_USER, 5/27/2005

Slide 24

I1 <Joel>In the interest of time, this could be dropped, but we need to be sure to mention that the modelReferences are independent of the ontology representation language
IBM_USER, 5/27/2005

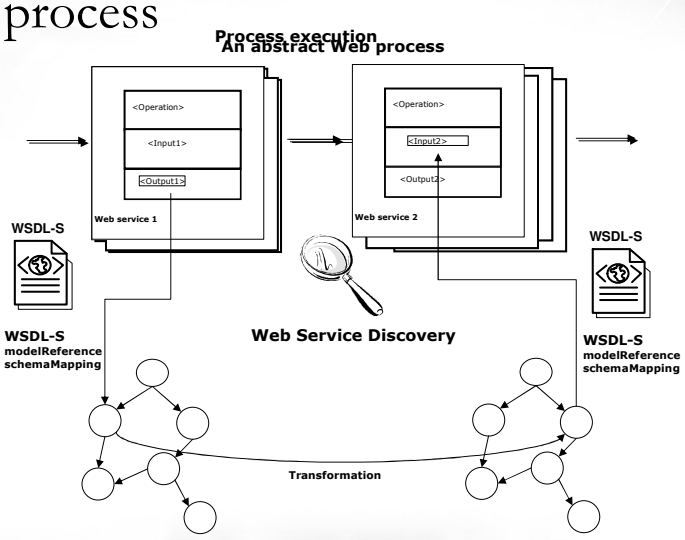

Large Scale Distributed Information Systems
IBM
Using WSDL-S in Web Process Lifecycle
25


Large Scale Distributed Information Systems
IBM
WSDL-S in the life cycle of a Web service
...
<xs:complexType name="processPurchaseOrderRequest">
<xs:element name="billingInfo" type="xsd1:POBilling"/>
<xs:element name="orderItem" type="xsd1:POItem"/>
</xs:complexType>
</xs:schema>
<operation name="processPurchaseOrder" pattern=wsdl:in-out>
<input messageLabel = "processPurchaseOrderRequest"
element="tns:processPurchaseOrderRequest"/>
<output messageLabel ="processPurchaseOrderResponse"
element="processPurchaseOrderResponse"/>
...
WSDL
Annotating a service
...
<xs:complexType name="processPurchaseOrderRequest"
wssem:modelReference="POOntology#OrderDetails" >
<xs:element name="billingInfo" type="xsd1:POBilling"/>
<xs:element name="orderItem" type="xsd1:POItem"/>
</xs:complexType>
</xs:schema>
<operation name="processPurchaseOrder" pattern=wsdl:in-out>
<input messageLabel = "processPurchaseOrderRequest"
element="tns:processPurchaseOrderRequest"/>
<output messageLabel ="processPurchaseOrderResponse"
element="processPurchaseOrderResponse"/>
...
WSDL - S
Semantic Layer
UDDI
Publishing a service
26
Sivashanmugam, K., Verma, K., Sheth, A., Miller, J., Adding Semantics to Web Services Standards, ICWS 2003

Large Scale Distributed Information Systems

IBM

WSDL-S in the life cycle of a Web process



27

Sivashanmugam, K., Verma, K., Sheth, A., Miller, J., Adding Semantics to Web Services Standards, ICWS 2003

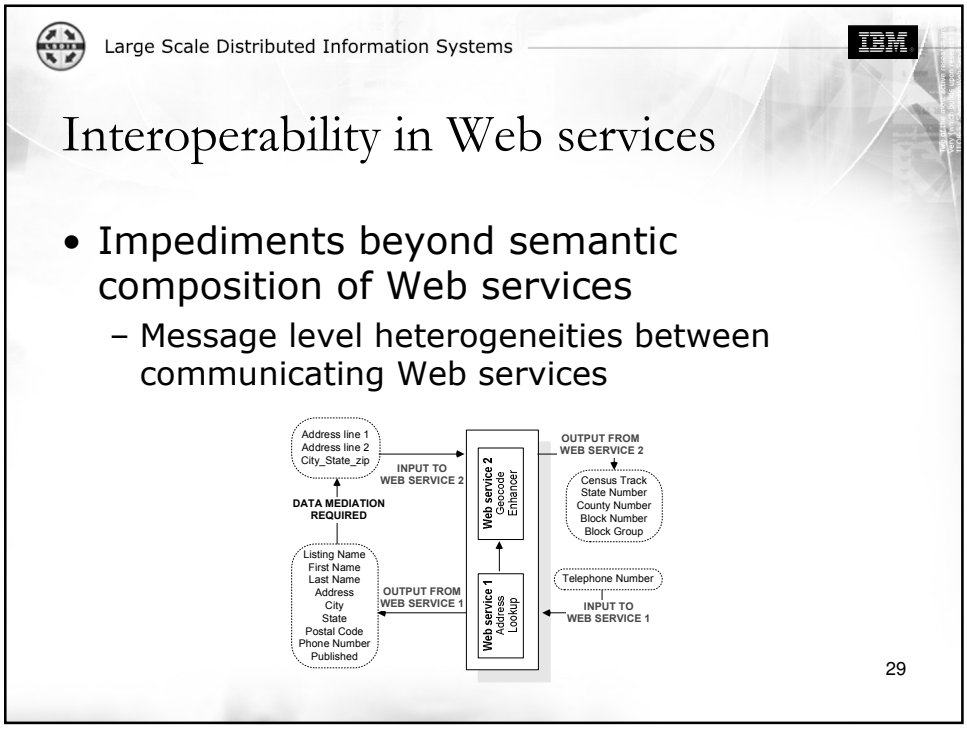
Large Scale Distributed Information Systems

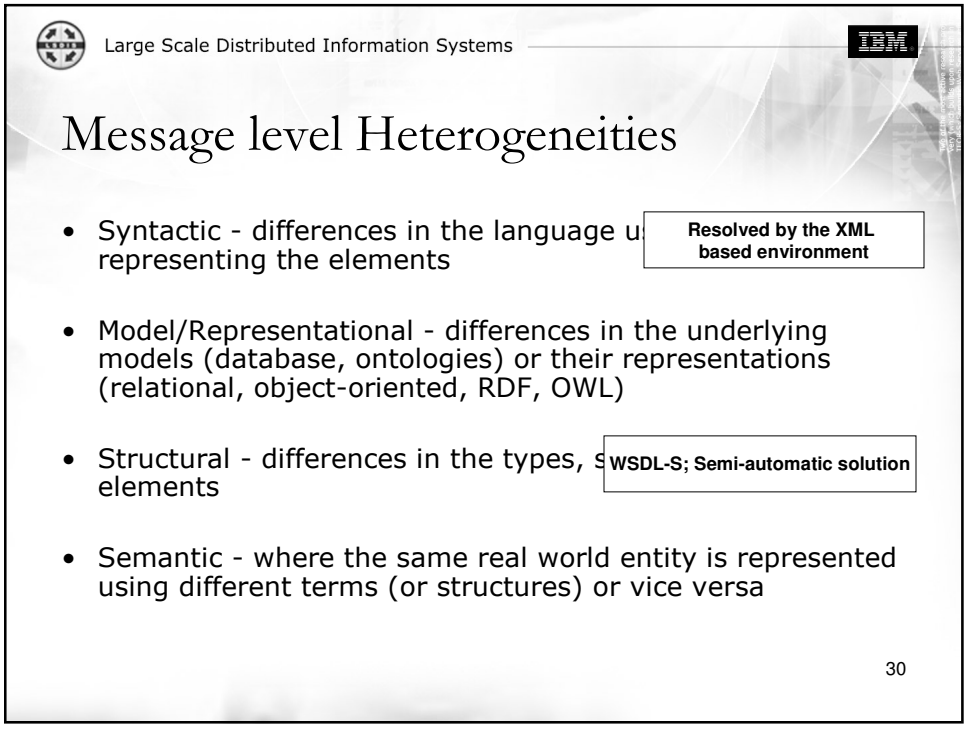
IBM

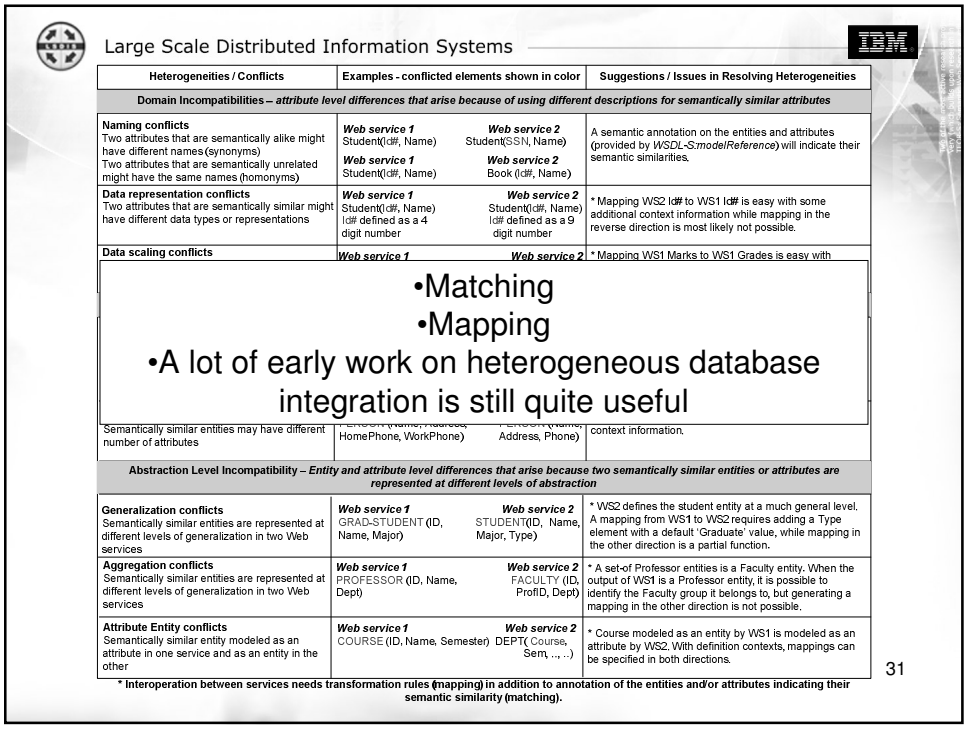
WSDL-S for advanced lifecycle support

- Data Mediation support
  - with Axis2 implementation
- Dynamic Process Binding
  - Process template
- Adaptation

28

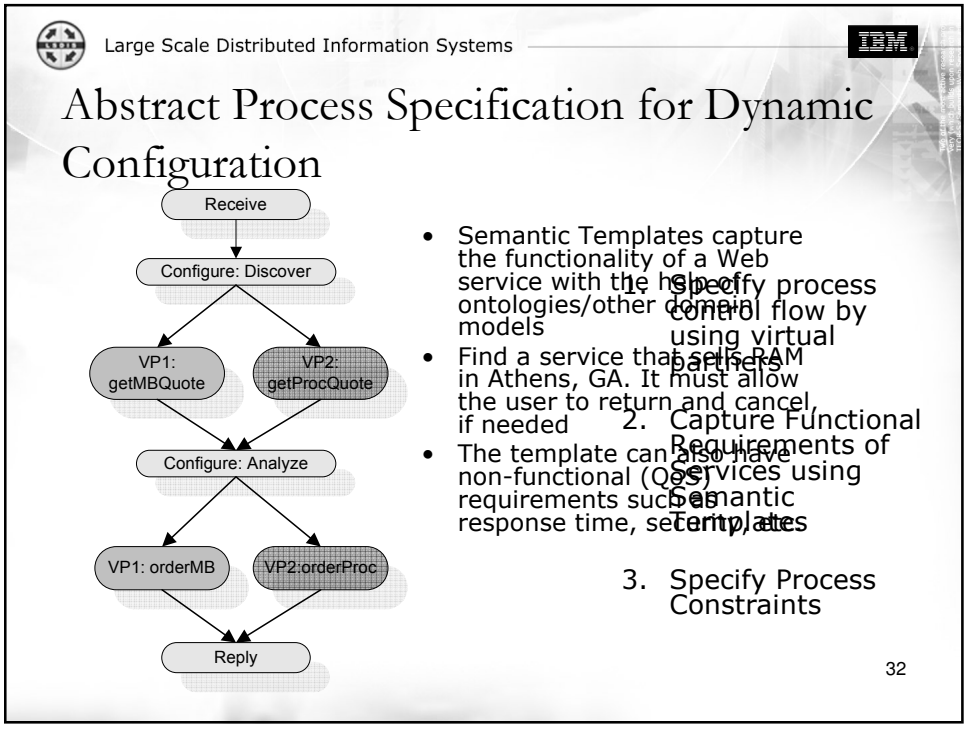

Large Scale Distributed Information Systems
IBM
Interoperability in Web services
• Impediments beyond semantic composition of Web services
– Message level heterogeneities between communicating Web services
Address line 1
Address line 2
City_State_zip
INPUT TO WEB SERVICE 2
DATA MEDIATION REQUIRED
Listing Name
First Name
Last Name
Address
City
State
Postal Code
Phone Number
Published
OUTPUT FROM WEB SERVICE 1
Web service 2
Geocode Enhancer
Web service 1
Address Lookup
OUTPUT FROM WEB SERVICE 2
Census Track
State Number
County Number
Block Number
Block Group
Telephone Number
INPUT TO WEB SERVICE 1
29

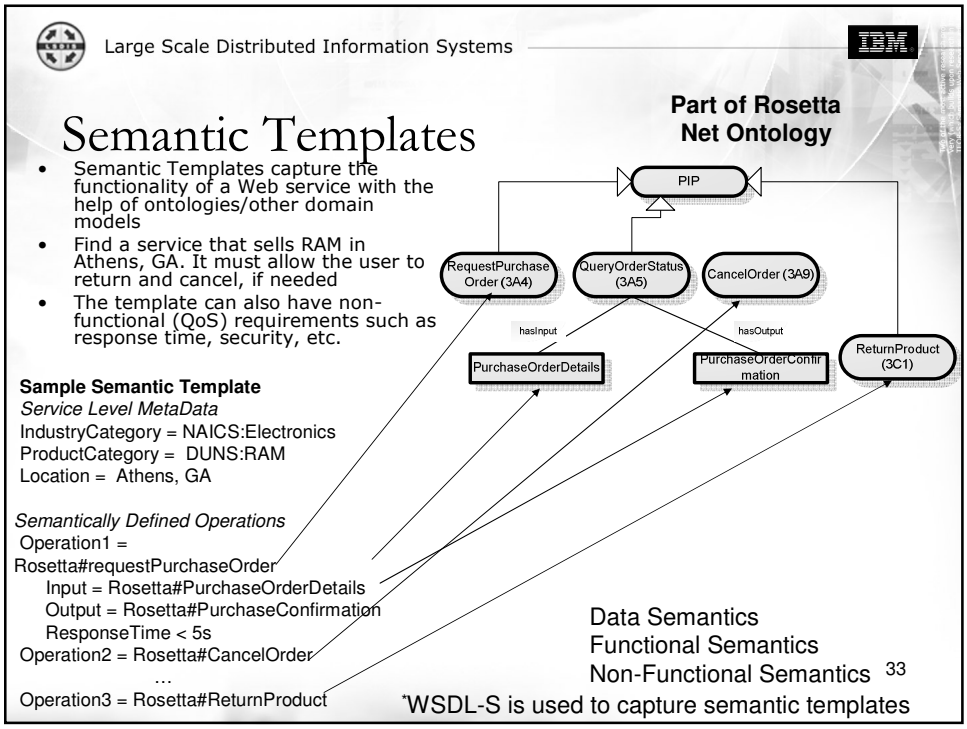

Large Scale Distributed Information Systems
IBM
Message level Heterogeneities
• Syntactic - differences in the language u representing the elements
Resolved by the XML based environment
• Model/Representational - differences in the underlying models (database, ontologies) or their representations (relational, object-oriented, RDF, OWL)
• Structural - differences in the types, s elements
WSDL-S; Semi-automatic solution
• Semantic - where the same real world entity is represented using different terms (or structures) or vice versa
30

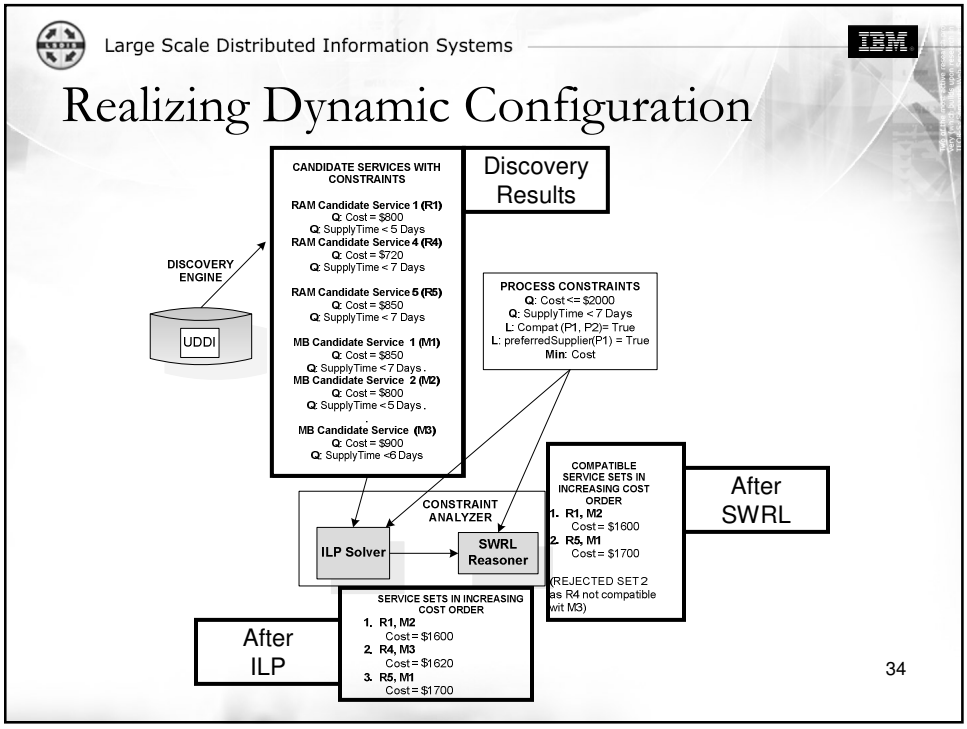


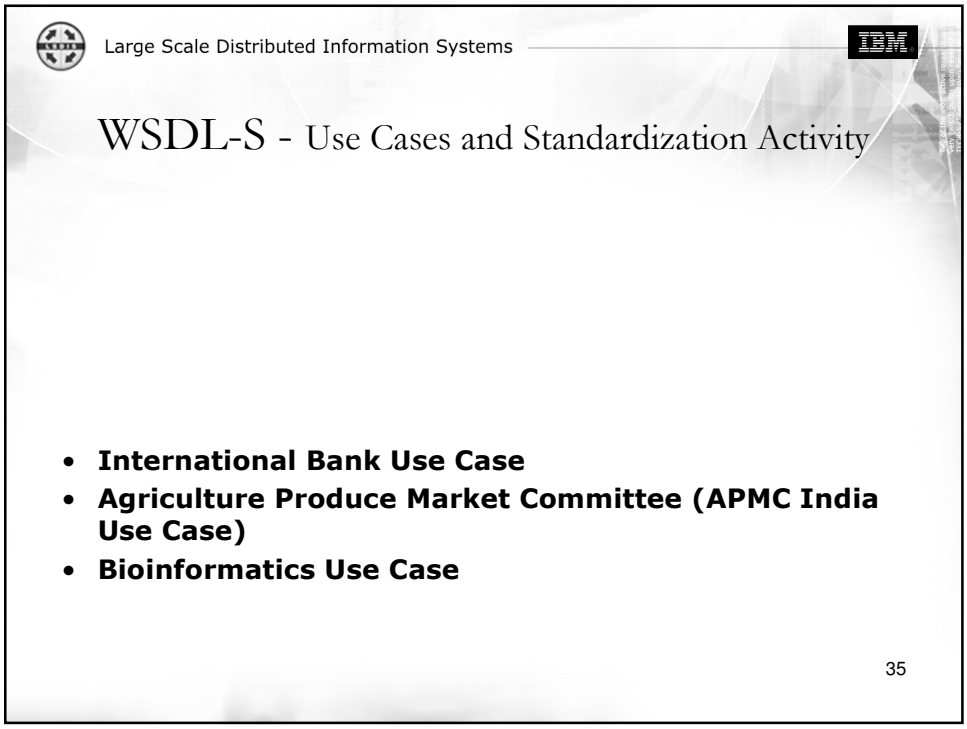
Heterogeneities / Conflicts	Examples - conflicted elements shown in color	Suggestions / Issues in Resolving Heterogeneities
Domain Incompatibilities – *attribute level differences that arise because of using different descriptions for semantically similar attributes*		
Naming conflicts Two attributes that are semantically alike might have different names (synonyms) Two attributes that are semantically unrelated might have the same names (homonyms)	***Web service 1*** Student(Id#, Name) ***Web service 2*** Student(SSN, Name) ***Web service 1*** Student(Id#, Name) ***Web service 2*** Book (Id#, Name)	A semantic annotation on the entities and attributes (provided by *WSDL-S:modelReference*) will indicate their semantic similarities.
Data representation conflicts Two attributes that are semantically similar might have different data types or representations	***Web service 1*** Student(Id#, Name) Id# defined as a 4 digit number ***Web service 2*** Student(Id#, Name) Id# defined as a 9 digit number	* Mapping WS2 Id# to WS1 Id# is easy with some additional context information while mapping in the reverse direction is most likely not possible.
Data scaling conflicts	***Web service 1*** ***Web service 2***	* Mapping WS1 Marks to WS1 Grades is easy with
Semantically similar entities may have different number of attributes	HomePhone, WorkPhone) Address, Phone)	context information.
Abstraction Level Incompatibility – *Entity and attribute level differences that arise because two semantically similar entities or attributes are represented at different levels of abstraction*		
Generalization conflicts Semantically similar entities are represented at different levels of generalization in two Web services	***Web service 1*** GRAD-STUDENT (ID, Name, Major) ***Web service 2*** STUDENT(ID, Name, Major, Type)	* WS2 defines the student entity at a much general level. A mapping from WS1 to WS2 requires adding a Type element with a default 'Graduate' value, while mapping in the other direction is a partial function.
Aggregation conflicts Semantically similar entities are represented at different levels of generalization in two Web services	***Web service 1*** PROFESSOR (ID, Name, Dept) ***Web service 2*** FACULTY (ID, ProfID, Dept)	* A set-of Professor entities is a Faculty entity. When the output of WS1 is a Professor entity, it is possible to identify the Faculty group it belongs to, but generating a mapping in the other direction is not possible.
Attribute Entity conflicts Semantically similar entity modeled as an attribute in one service and as an entity in the other	***Web service 1*** COURSE (ID, Name, Semester) ***Web service 2*** DEPT(Course, Sem, .., ..)	* Course modeled as an entity by WS1 is modeled as an attribute by WS2. With definition contexts, mappings can be specified in both directions.

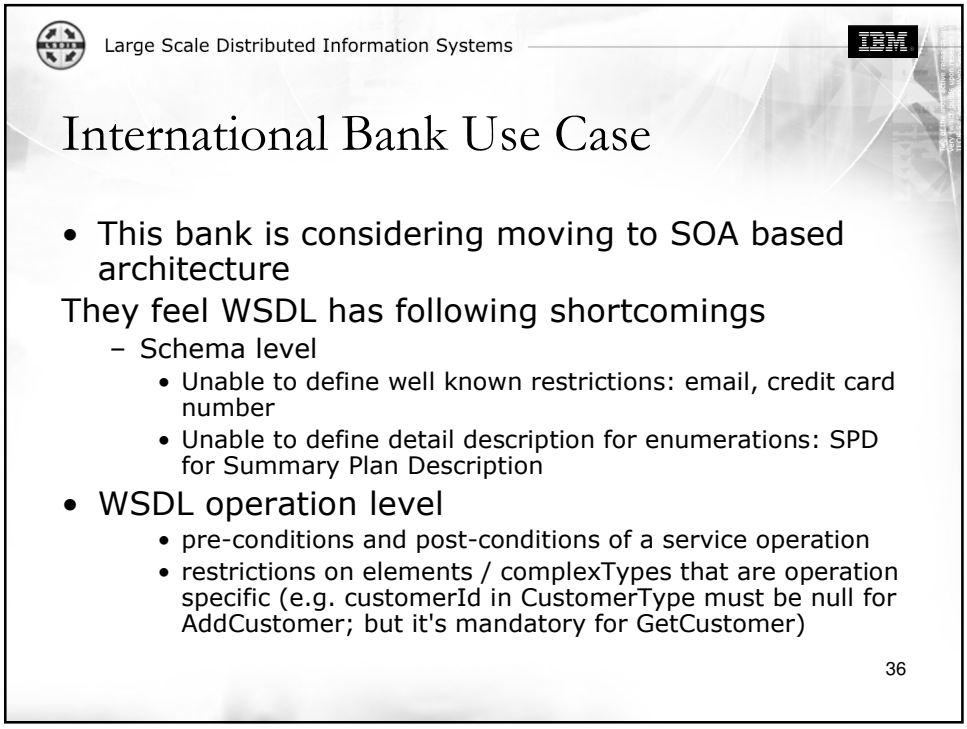
*** Interoperation between services needs transformation rules (mapping) in addition to annotation of the entities and/or attributes indicating their semantic similarity (matching).**




Large Scale Distributed Information Systems
IBM
Semantic Templates
Semantic Templates capture the functionality of a Web service with the help of ontologies/other domain models
Find a service that sells RAM in Athens, GA. It must allow the user to return and cancel, if needed
The template can also have non-functional (QoS) requirements such as response time, security, etc.
Sample Semantic Template
Service Level MetaData
IndustryCategory = NAICS:Electronics
ProductCategory = DUNS:RAM
Location = Athens, GA
Semantically Defined Operations
Operation1 = Rosetta#requestPurchaseOrder
Input = Rosetta#PurchaseOrderDetails
Output = Rosetta#PurchaseConfirmation
ResponseTime < 5s
Operation2 = Rosetta#CancelOrder
...
Operation3 = Rosetta#ReturnProduct
Part of Rosetta Net Ontology
PIP
RequestPurchaseOrder (3A4)
QueryOrderStatus (3A5)
CancelOrder (3A9)
hasInput
hasOutput
PurchaseOrderDetails
PurchaseOrderConfirmation
ReturnProduct (3C1)
Data Semantics
Functional Semantics
Non-Functional Semantics
33
*WSDL-S is used to capture semantic templates


Large Scale Distributed Information Systems
IBM
Realizing Dynamic Configuration
CANDIDATE SERVICES WITH CONSTRAINTS
RAM Candidate Service 1 (R1)
Q: Cost = $800
Q: SupplyTime < 5 Days
RAM Candidate Service 4 (R4)
Q: Cost = $720
Q: SupplyTime < 7 Days
RAM Candidate Service 5 (R5)
Q: Cost = $850
Q: SupplyTime < 7 Days
MB Candidate Service 1 (M1)
Q: Cost = $850
Q: SupplyTime < 7 Days .
MB Candidate Service 2 (M2)
Q: Cost = $800
Q: SupplyTime < 5 Days .
MB Candidate Service (M3)
Q: Cost = $900
Q: SupplyTime <6 Days
Discovery Results
DISCOVERY ENGINE
UDDI
PROCESS CONSTRAINTS
Q: Cost <= $2000
Q: SupplyTime < 7 Days
L: Compat (P1, P2)= True
L: preferredSupplier(P1) = True
Min: Cost
CONSTRAINT ANALYZER
ILP Solver
SWRL Reasoner
COMPATIBLE SERVICE SETS IN INCREASING COST ORDER
1. R1, M2
Cost = $1600
2. R5, M1
Cost = $1700
(REJECTED SET 2 as R4 not compatible wit M3)
After SWRL
SERVICE SETS IN INCREASING COST ORDER
1. R1, M2
Cost = $1600
2. R4, M3
Cost = $1620
3. R5, M1
Cost = $1700
After ILP
34


Large Scale Distributed Information Systems
IBM
WSDL-S - Use Cases and Standardization Activity
• International Bank Use Case
• Agriculture Produce Market Committee (APMC India Use Case)
• Bioinformatics Use Case
35

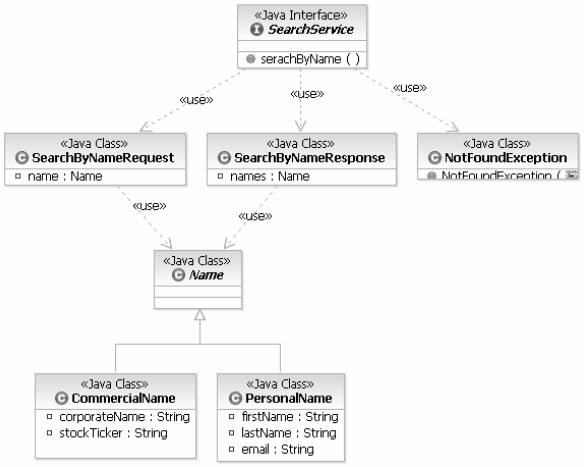

Large Scale Distributed Information Systems
IBM
International Bank Use Case
• This bank is considering moving to SOA based architecture
They feel WSDL has following shortcomings
– Schema level
• Unable to define well known restrictions: email, credit card number
• Unable to define detail description for enumerations: SPD for Summary Plan Description
• WSDL operation level
• pre-conditions and post-conditions of a service operation
• restrictions on elements / complexTypes that are operation specific (e.g. customerId in CustomerType must be null for AddCustomer; but it's mandatory for GetCustomer)
36

Large Scale Distributed Information Systems

IBM

Use Case Details

- A search service is defined to search by either personal name or commercial name.
- The search engine would return at least one element of names, or a SOAP fault



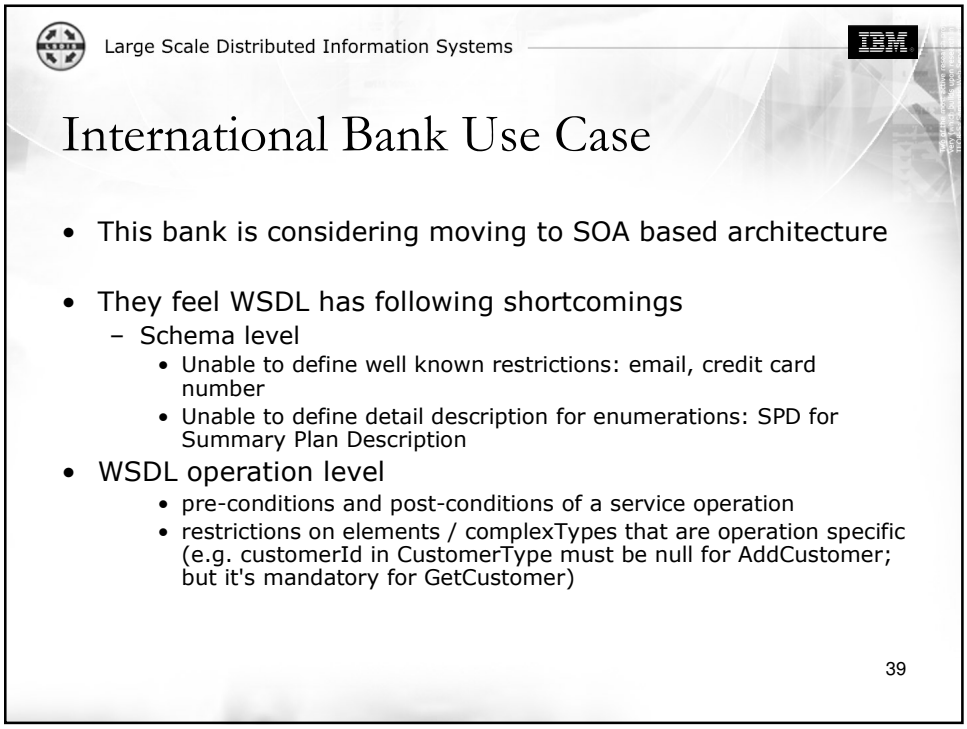
Large Scale Distributed Information Systems

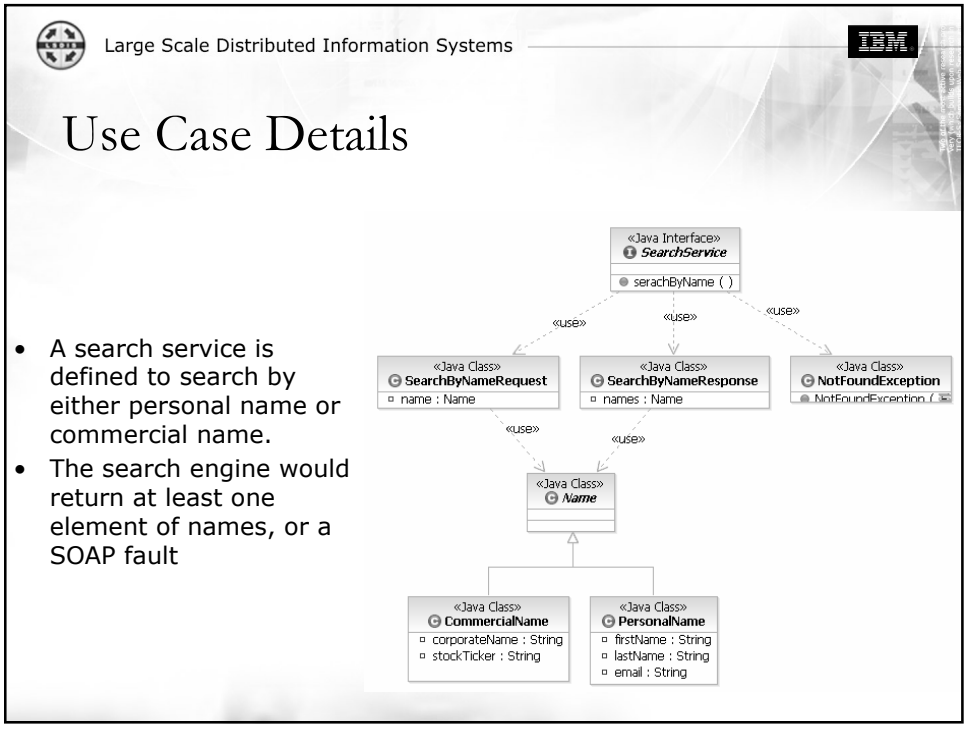
IBM

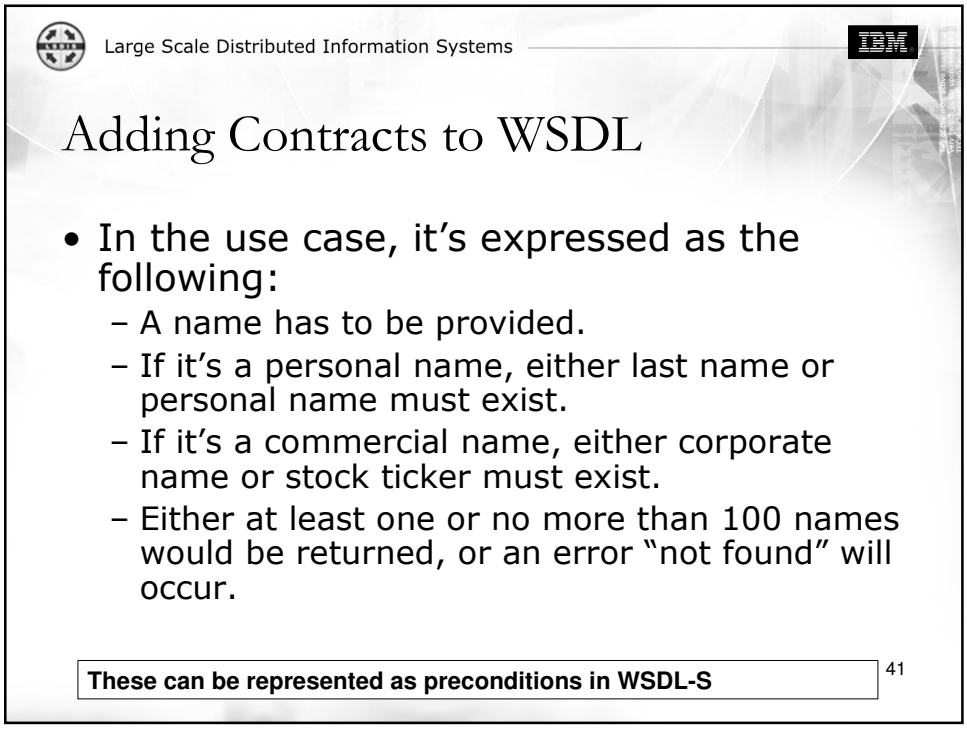
Adding Contracts to WSDL

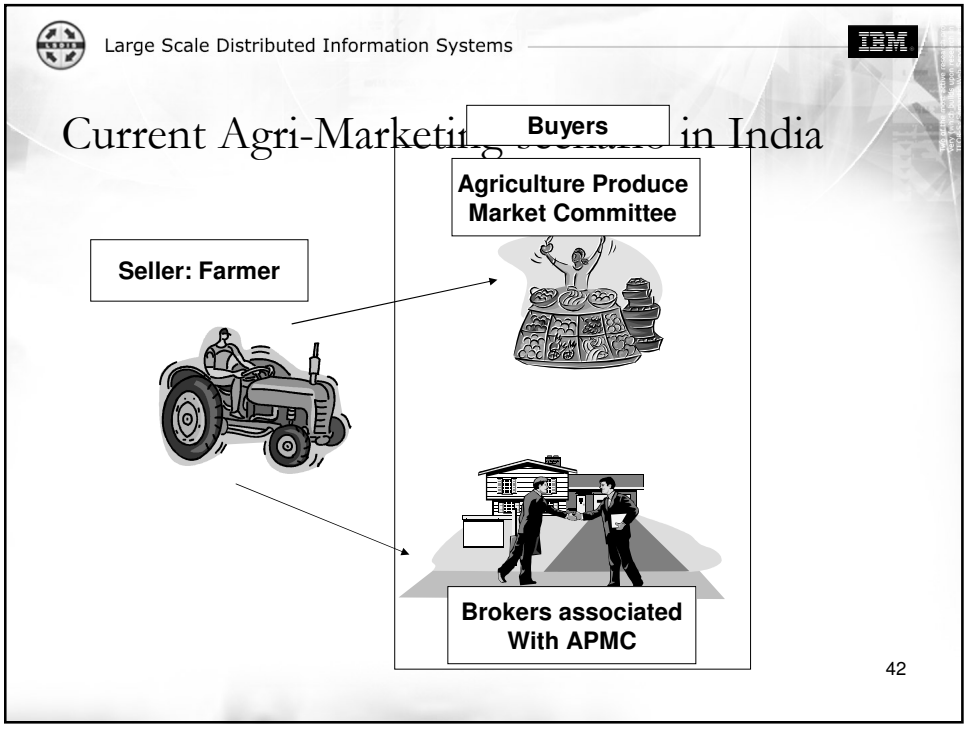
- In the use case, it's expressed as the following:
  - A name has to be provided.
  - If it's a personal name, either last name or personal name must exist.
  - If it's a commercial name, either corporate name or stock ticker must exist.
  - Either at least one or no more than 100 names would be returned, or an error "not found" will occur.

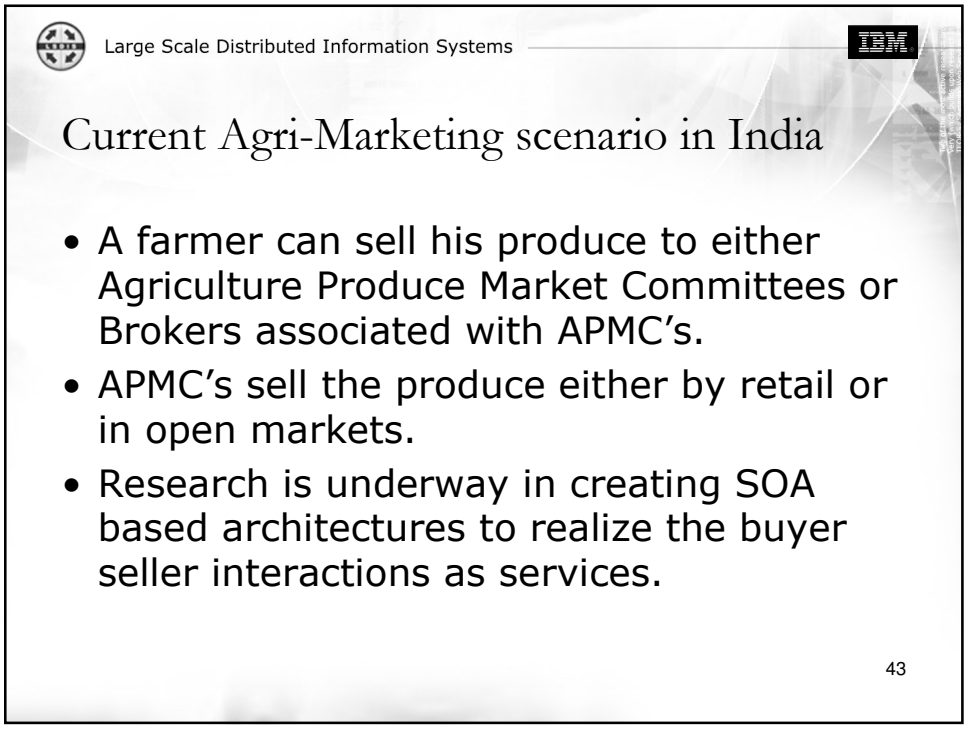
Currently, we are working with the bank to realize this case using WSDL-S

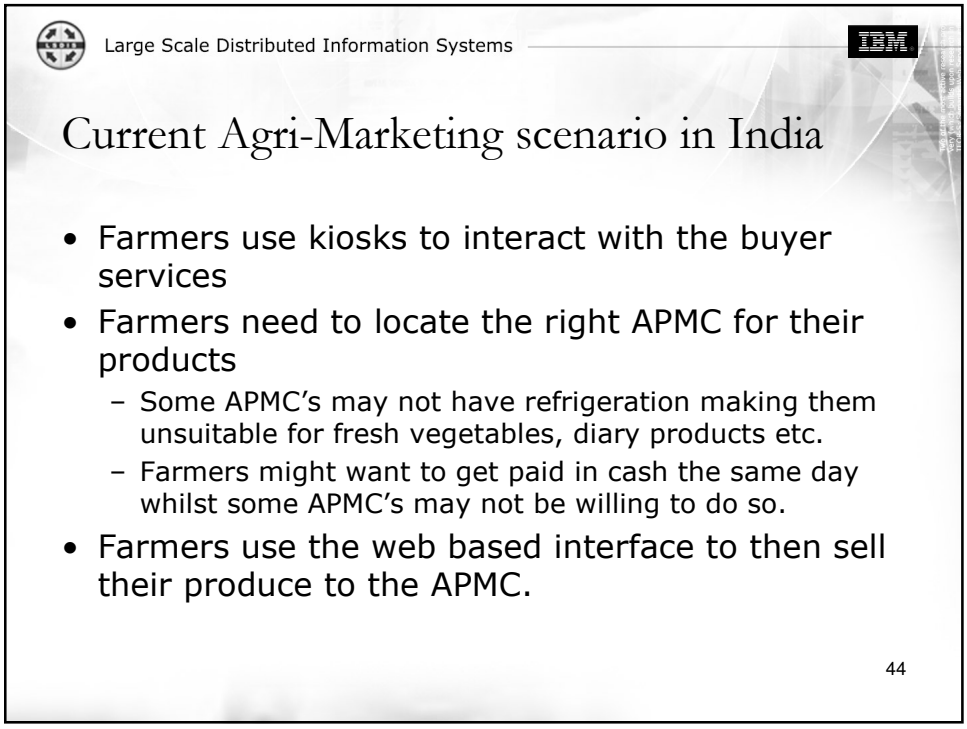

Large Scale Distributed Information Systems
IBM
International Bank Use Case
• This bank is considering moving to SOA based architecture
• They feel WSDL has following shortcomings
– Schema level
• Unable to define well known restrictions: email, credit card number
• Unable to define detail description for enumerations: SPD for Summary Plan Description
• WSDL operation level
• pre-conditions and post-conditions of a service operation
• restrictions on elements / complexTypes that are operation specific (e.g. customerId in CustomerType must be null for AddCustomer; but it's mandatory for GetCustomer)
39

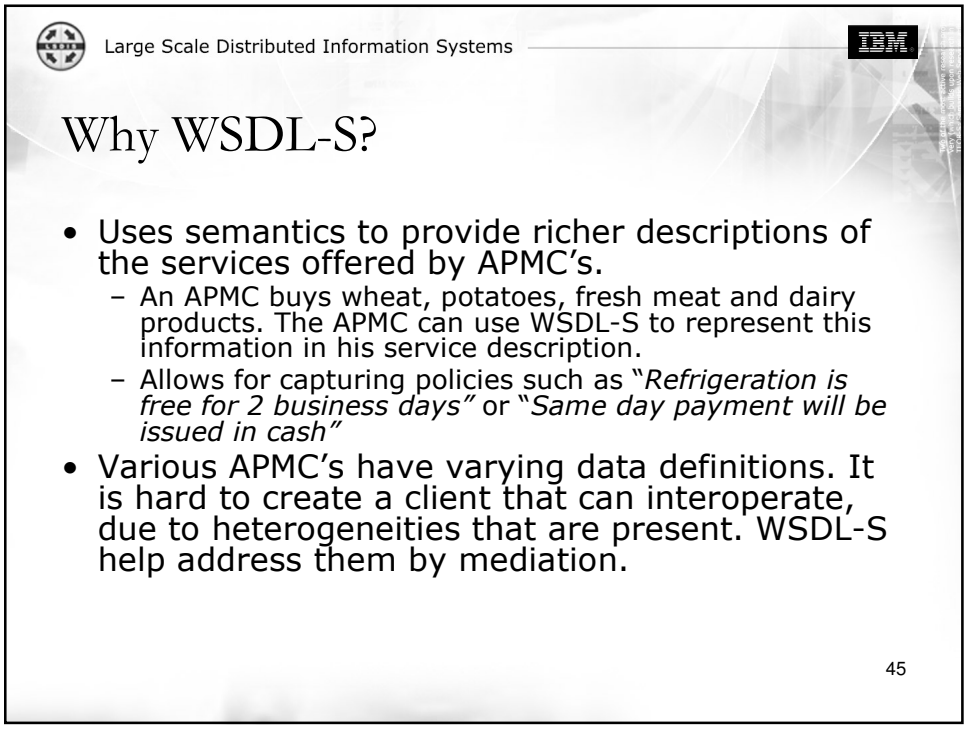

Large Scale Distributed Information Systems
IBM
Use Case Details
• A search service is defined to search by either personal name or commercial name.
• The search engine would return at least one element of names, or a SOAP fault
«Java Interface»
SearchService
serachByName ( )
«use»
«use»
«use»
«Java Class»
SearchByNameRequest
name : Name
«Java Class»
SearchByNameResponse
names : Name
«Java Class»
NotFoundException
«use»
«use»
«Java Class»
Name
«Java Class»
CommercialName
corporateName : String
stockTicker : String
«Java Class»
PersonalName
firstName : String
lastName : String
email : String

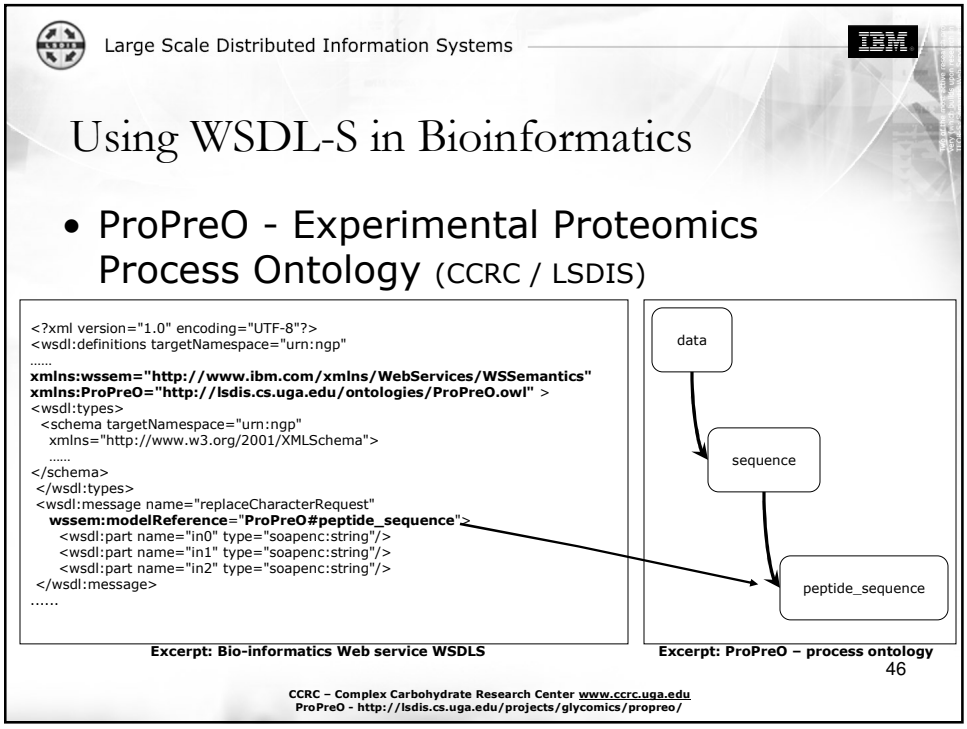

Large Scale Distributed Information Systems
IBM
Adding Contracts to WSDL
• In the use case, it's expressed as the following:
– A name has to be provided.
– If it's a personal name, either last name or personal name must exist.
– If it's a commercial name, either corporate name or stock ticker must exist.
– Either at least one or no more than 100 names would be returned, or an error "not found" will occur.
These can be represented as preconditions in WSDL-S
41

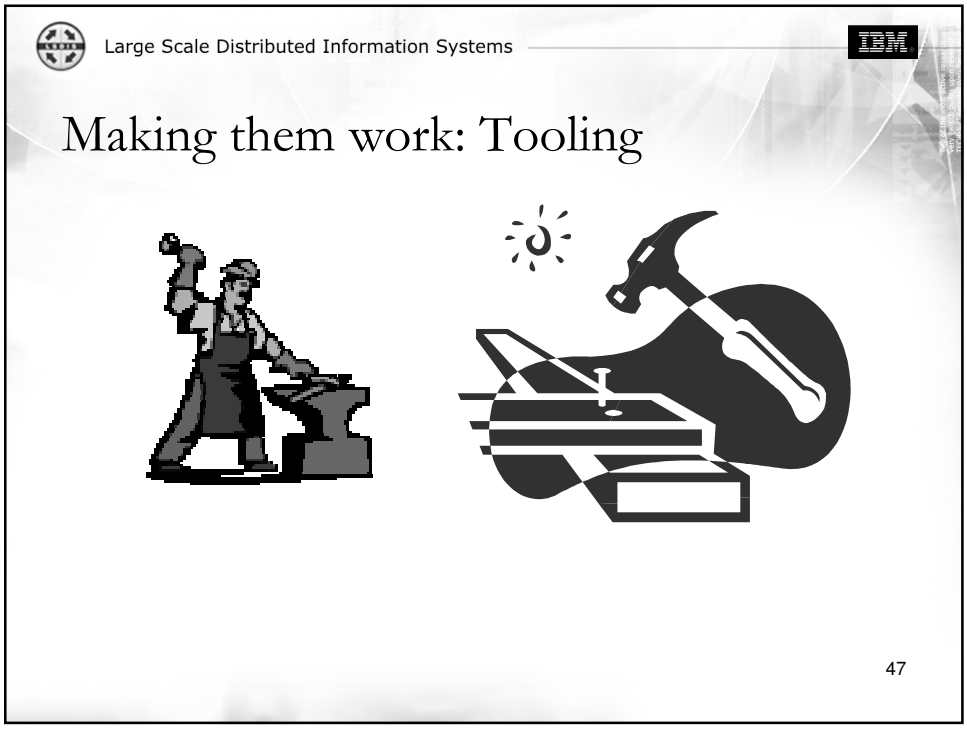

Large Scale Distributed Information Systems
IBM
Current Agri-Marketing Scenario in India
Buyers
Agriculture Produce Market Committee
Seller: Farmer
Brokers associated With APMC
42

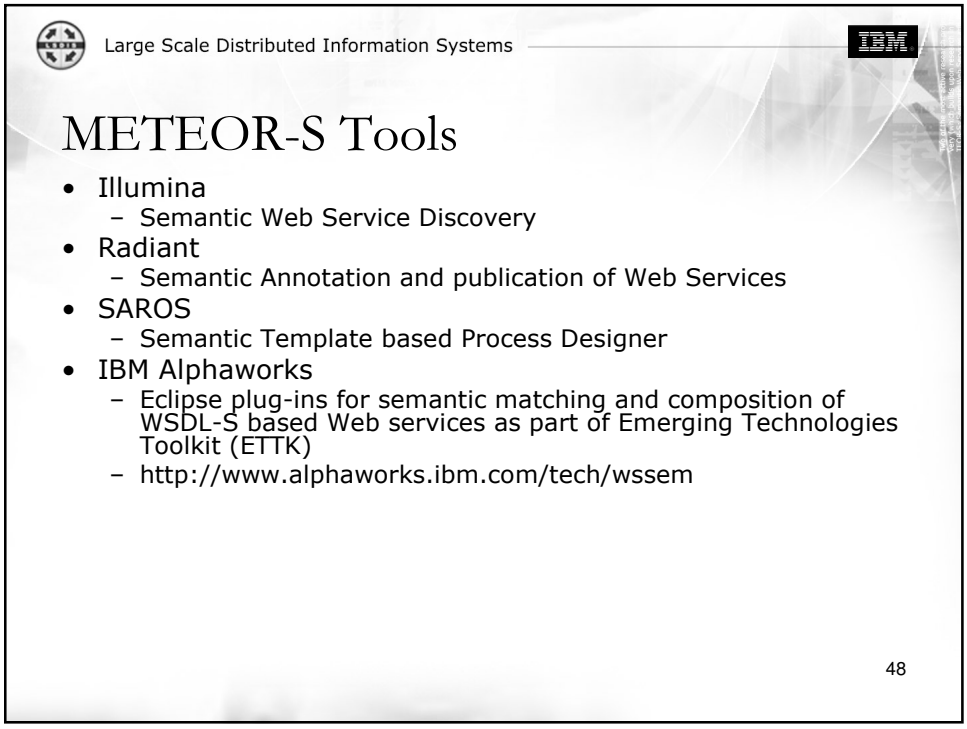

Large Scale Distributed Information Systems
IBM
Current Agri-Marketing scenario in India
• A farmer can sell his produce to either Agriculture Produce Market Committees or Brokers associated with APMC's.
• APMC's sell the produce either by retail or in open markets.
• Research is underway in creating SOA based architectures to realize the buyer seller interactions as services.
43

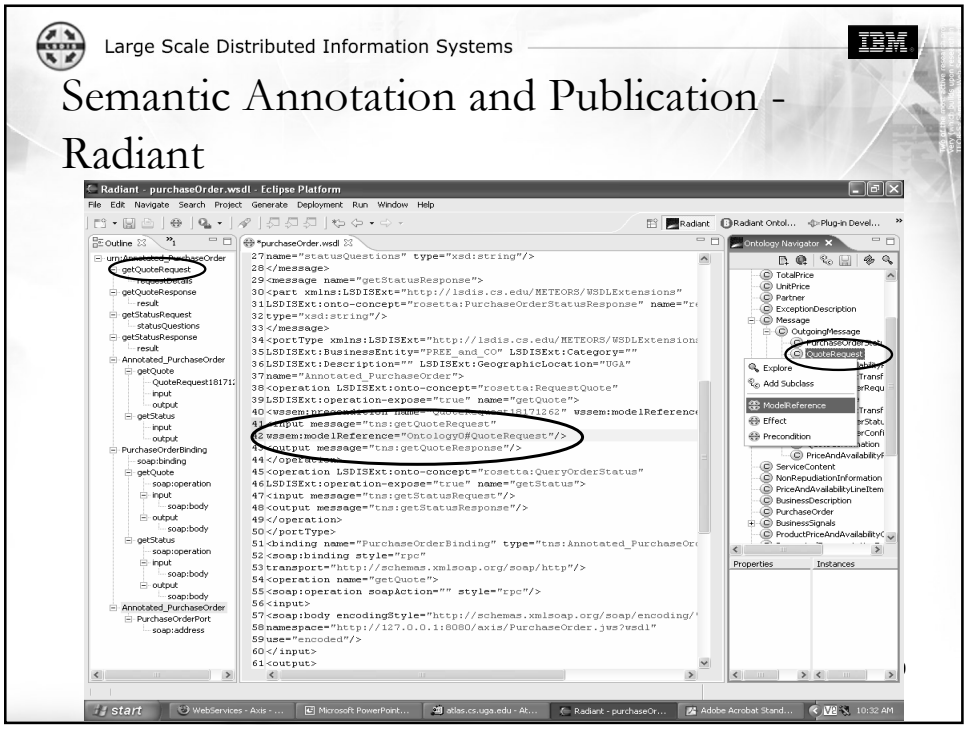

Large Scale Distributed Information Systems
IBM
Current Agri-Marketing scenario in India
• Farmers use kiosks to interact with the buyer services
• Farmers need to locate the right APMC for their products
– Some APMC's may not have refrigeration making them unsuitable for fresh vegetables, diary products etc.
– Farmers might want to get paid in cash the same day whilst some APMC's may not be willing to do so.
• Farmers use the web based interface to then sell their produce to the APMC.
44

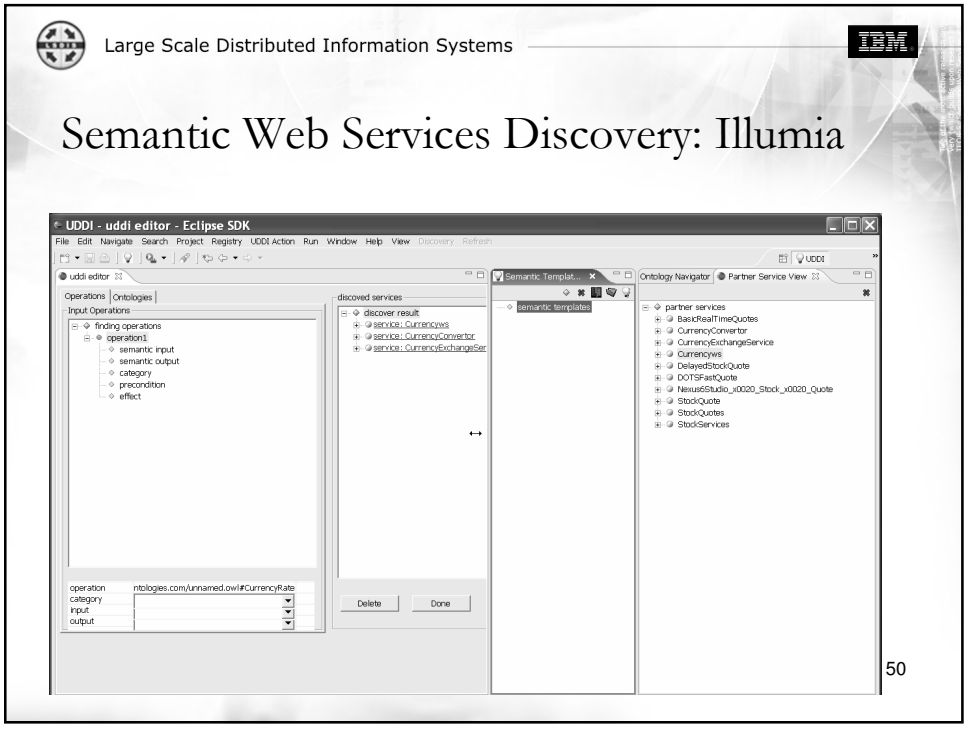

Large Scale Distributed Information Systems
IBM
Why WSDL-S?
• Uses semantics to provide richer descriptions of the services offered by APMC's.
– An APMC buys wheat, potatoes, fresh meat and dairy products. The APMC can use WSDL-S to represent this information in his service description.
– Allows for capturing policies such as "Refrigeration is free for 2 business days" or "Same day payment will be issued in cash"
• Various APMC's have varying data definitions. It is hard to create a client that can interoperate, due to heterogeneities that are present. WSDL-S help address them by mediation.
45


Large Scale Distributed Information Systems
IBM
Using WSDL-S in Bioinformatics
• ProPreO - Experimental Proteomics Process Ontology (CCRC / LSDIS)
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="urn:ngp"
.....
xmlns:wssem="http://www.ibm.com/xmlns/WebServices/WSSemantics"
xmlns:ProPreO="http://lsdis.cs.uga.edu/ontologies/ProPreO.owl" >
<wsdl:types>
<schema targetNamespace="urn:ngp"
xmlns="http://www.w3.org/2001/XMLSchema">
.....
</schema>
</wsdl:types>
<wsdl:message name="replaceCharacterRequest"
wssem:modelReference="ProPreO#peptide_sequence">
<wsdl:part name="in0" type="soapenc:string"/>
<wsdl:part name="in1" type="soapenc:string"/>
<wsdl:part name="in2" type="soapenc:string"/>
</wsdl:message>
.....
data
sequence
peptide_sequence
Excerpt: Bio-informatics Web service WSDLS
Excerpt: ProPreO – process ontology
46
CCRC – Complex Carbohydrate Research Center www.ccrc.uga.edu
ProPreO - http://lsdis.cs.uga.edu/projects/glycomics/propreo/


Large Scale Distributed Information Systems
IBM
Making them work: Tooling
47


Large Scale Distributed Information Systems
IBM
METEOR-S Tools
• Illumina
– Semantic Web Service Discovery
• Radiant
– Semantic Annotation and publication of Web Services
• SAROS
– Semantic Template based Process Designer
• IBM Alphaworks
– Eclipse plug-ins for semantic matching and composition of WSDL-S based Web services as part of Emerging Technologies Toolkit (ETTK)
– http://www.alphaworks.ibm.com/tech/wssem
48


Large Scale Distributed Information Systems
IBM
Semantic Annotation and Publication - Radiant
Radiant - purchaseOrder.wsdl - Eclipse Platform
Ontology Navigator
ModelReference
Effect
Precondition


Large Scale Distributed Information Systems
IBM
Semantic Web Services Discovery: Illumia
UDDI - uddi editor - Eclipse SDK
Delete
Done
50

13

Large Scale Distributed Information Systems

WSDL-S collaborations

- METEOR-S collaboration with WSMO
  - Using WSDL-S for grounding Web services annotated with WSML ontologies

- Influencing OASIS / W3C

51

Kunal Verma, Adrian Mochan, Michal Zaremba, Amit Sheth, John Miller, Christoph Bussler, Linking Semantics Web service Efforts - Integrating WSMX and METEOR-S, Second International Workshop on Semantic and Dynamic Web Processes, July 2005

Large Scale Distributed Information Systems

W3C Charters Proposed

- Charter of the Semantic Annotations for WSDL Working Group
  - Primary objective to use WSDL's extensibility mechanism to add more information to data definitions in WSDL
  - Also, recognizes WSDL-S as an important input

52

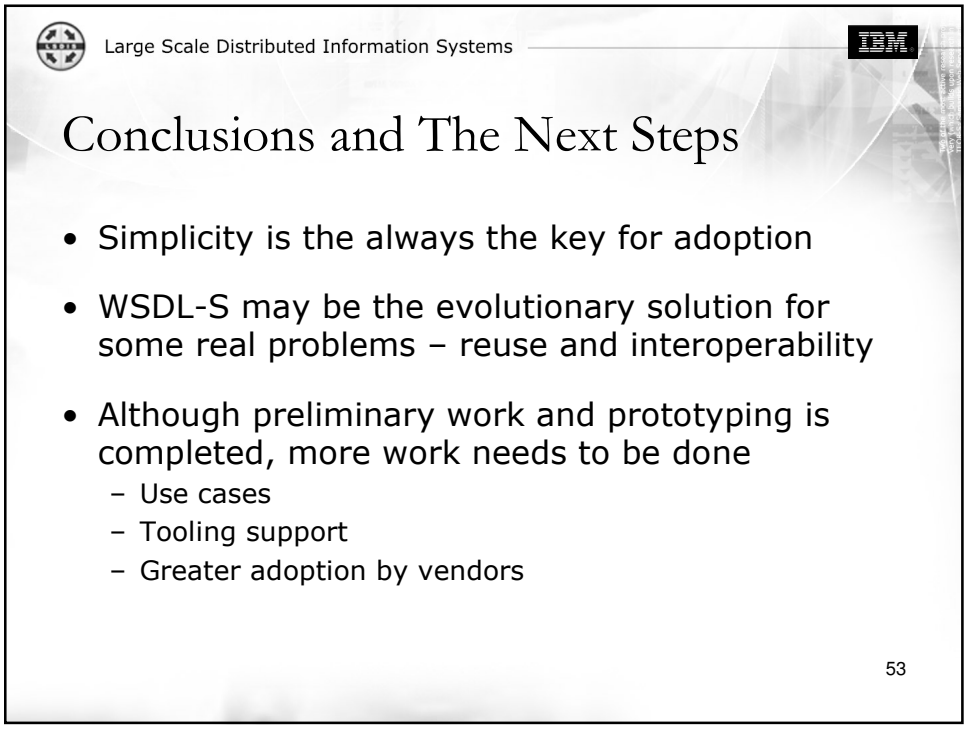
Slide 51

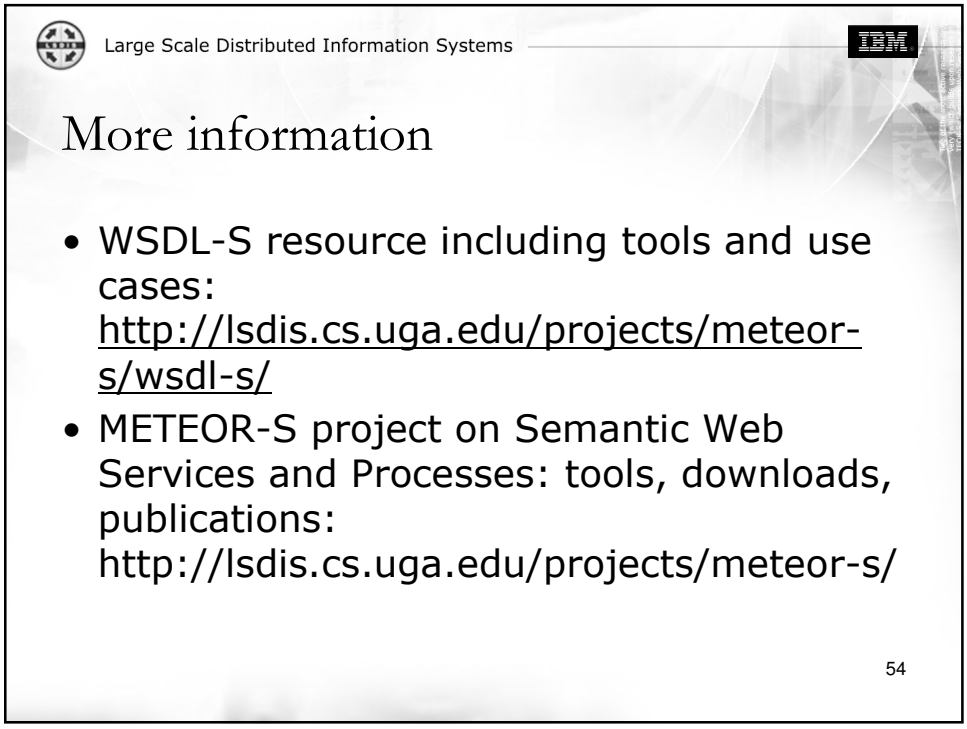
I3 <Joel>Points I would make:

If more revolutionary approaches are pursued, we must be sure that existing WSDL, XML Schemas for business documents, and the tools that exploit them can still be leveraged.

WSDL-S could be that bridge.

This is a key success factor.
IBM_USER, 5/27/2005


Large Scale Distributed Information Systems
IBM
Conclusions and The Next Steps
• Simplicity is the always the key for adoption
• WSDL-S may be the evolutionary solution for some real problems – reuse and interoperability
• Although preliminary work and prototyping is completed, more work needs to be done
– Use cases
– Tooling support
– Greater adoption by vendors
53


Large Scale Distributed Information Systems
IBM
More information
• WSDL-S resource including tools and use cases: http://lsdis.cs.uga.edu/projects/meteor-s/wsdl-s/
• METEOR-S project on Semantic Web Services and Processes: tools, downloads, publications: http://lsdis.cs.uga.edu/projects/meteor-s/
54