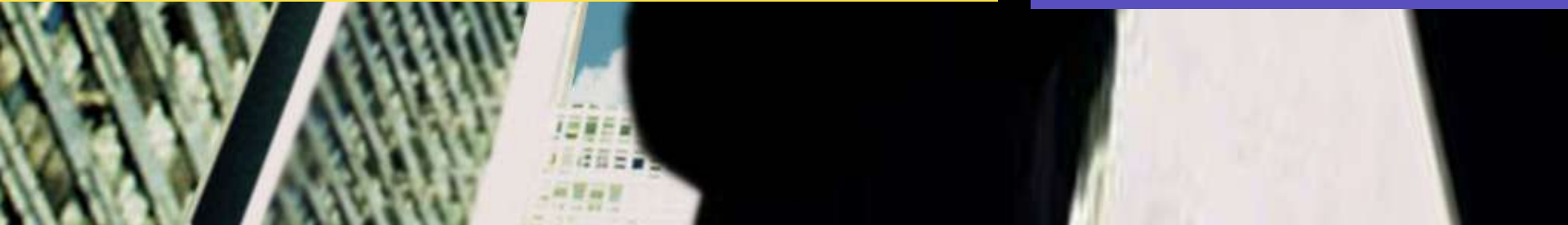




**ebXML Registry 3.0:
An Overview**



Agenda

- ▶ What is *ebXML Registry*?
- ▶ Major Use Cases
- ▶ Key Benefits
- ▶ Key Features
- ▶ ebXML Registry in Action
- ▶ Future Directions
- ▶ Summary

What is ebXML Registry?

- ♦ A system that enables secure, federated information management
 - Provides services for sharing content and metadata between entities in a federated environment
 - Provides a stable store where content is made persistent
- ♦ Defined by OASIS ebXML Registry 3.0 standard
- ♦ Is a registry as well as a repository

Registry vs. Repository

♦ Repository

- Contains any type of electronic content such as: XML Schema, XML Instance, WSDL, GIF image...
- A content instance in the Repository **MUST** have an associated metadata instance in the *Registry*

♦ Registry

- Contains metadata instances
- Metadata instance describes a content instance or another metadata instance
- Metadata instances enable discovery of content

What ebXML Registry Is Not

- ▶ Not just for ebXML artifacts
- ▶ Not just a web services registry
- ▶ Not just for design-time use
- ▶ Not dependent on any other ebXML specification

The Specifications

- ♦ ebXML Registry Information Model
 - Defines what metadata and content can be stored in the registry
- ♦ ebXML Registry Services and Protocols
 - Defines the services and service interfaces provided by the registry
 - Defines the API for accessing the registry
- ♦ ISO 15000 Standard, Part 3 and 4

Use Cases

- ◆ Web services registry
- ◆ Web content management
- ◆ Controlled vocabulary registry
- ◆ Business process registry
- ◆ eForms registry
- ◆ Standards registry
- ◆ Domain-specific use cases such as
 - Electronic Medical Records
 - Geological Information Systems

ebXML Registry Profiles

- ▶ Enable interoperability in particular domains
- ▶ Define restrictions and extensions of registry usage
- ▶ Current examples include
 - Web Services Profile
 - WSRP (Remote Portlets) Profile
 - Open GIS Profile
 - HL7 Profile
 - IHE XDR Profile

Web Services Registry Use

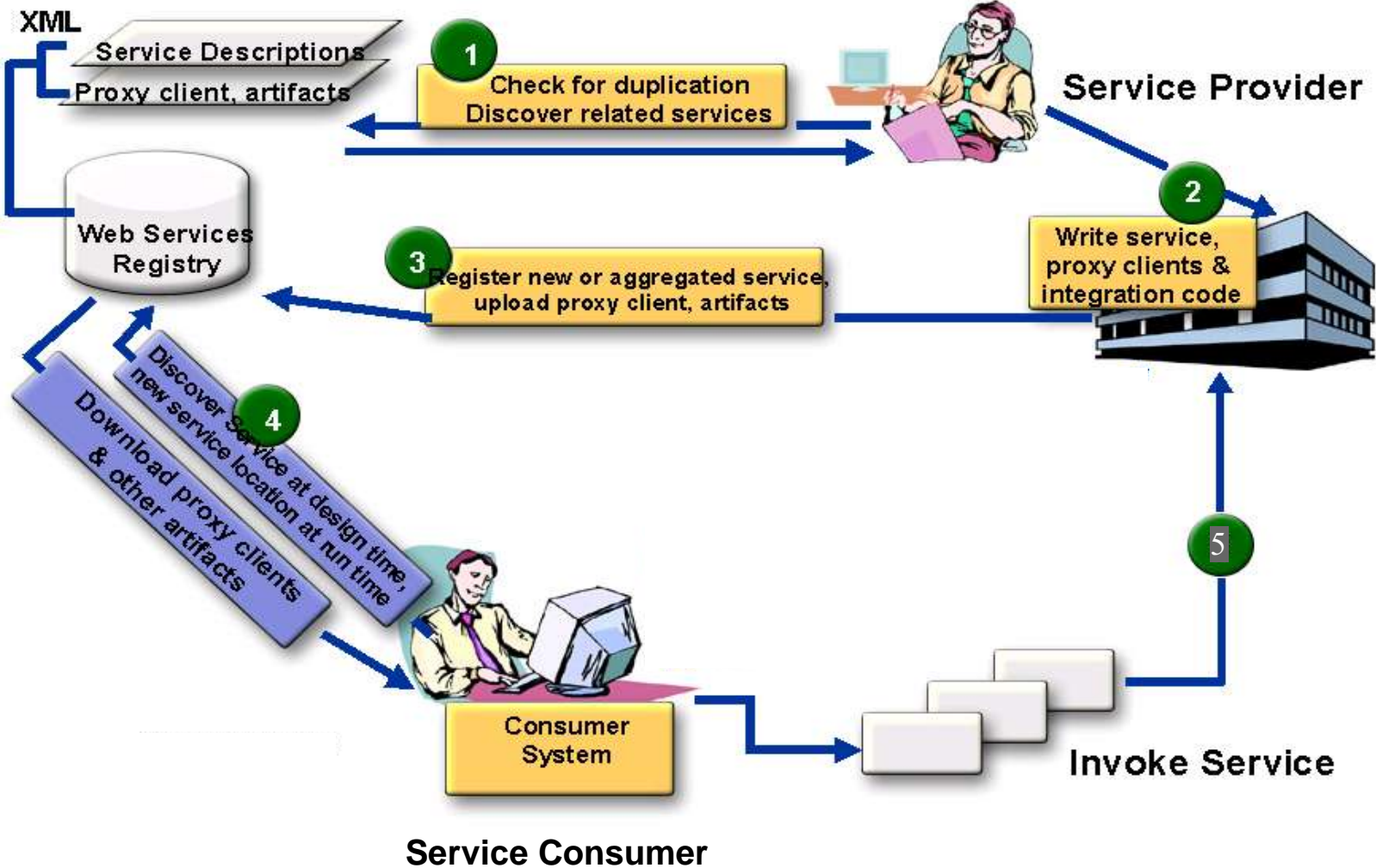
♦ High-level business function

Centralized manageability and accessibility of enterprise web services metadata and artifacts for developers, deployers, administrators, and users

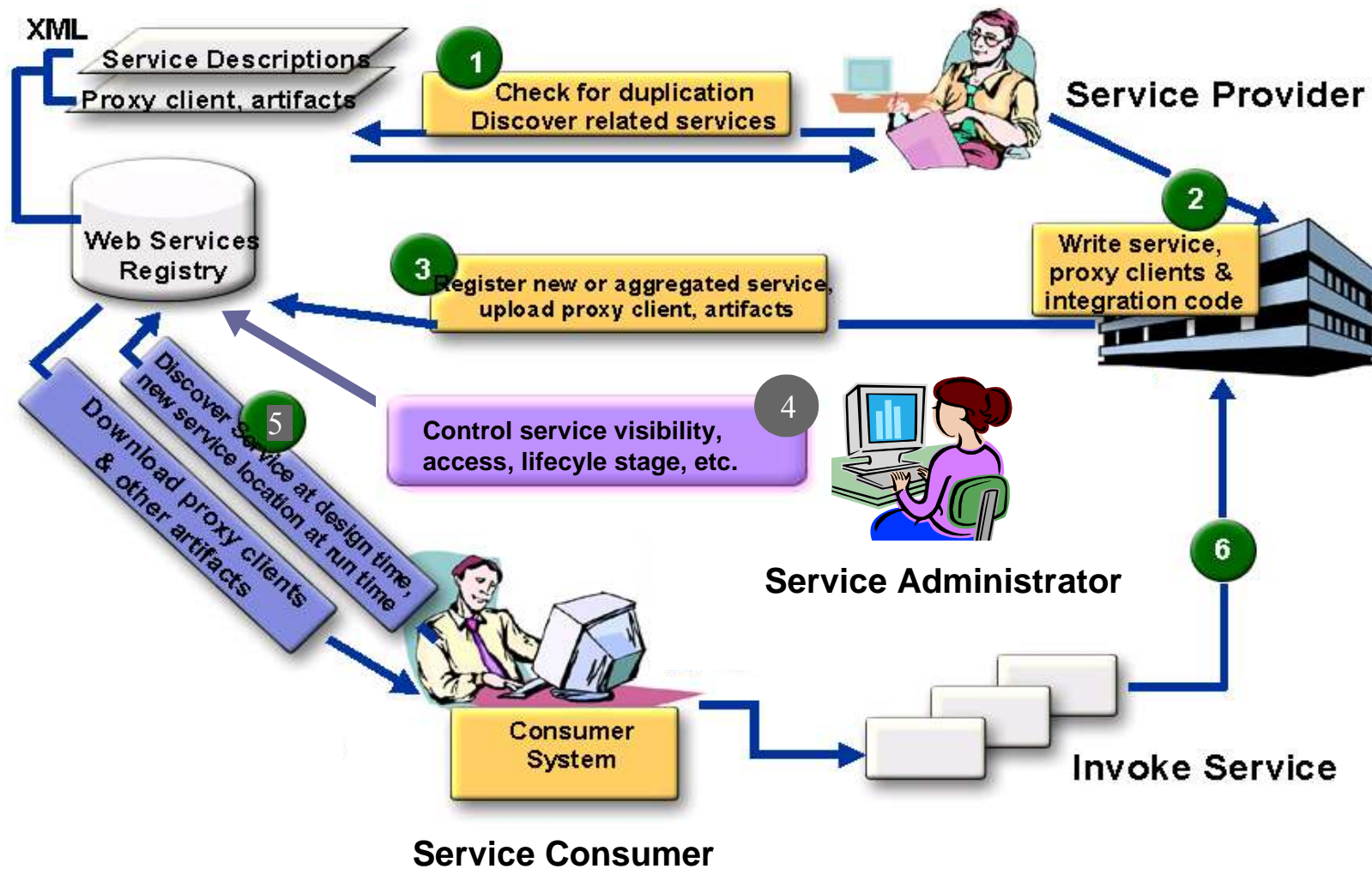
♦ Primary use cases

- Deploy/publish and federated discovery of web services metadata and artifacts
 - Service metadata and artifacts = WSDL's, schemas, BPEL and WSRP descriptions, XSLT transforms, etc.
- Web services governance and lifecycle management
 - Approval, visibility, use, deprecation, and deletion based on maturity level, user roles, organizational policies, etc.

Web Services Use Case



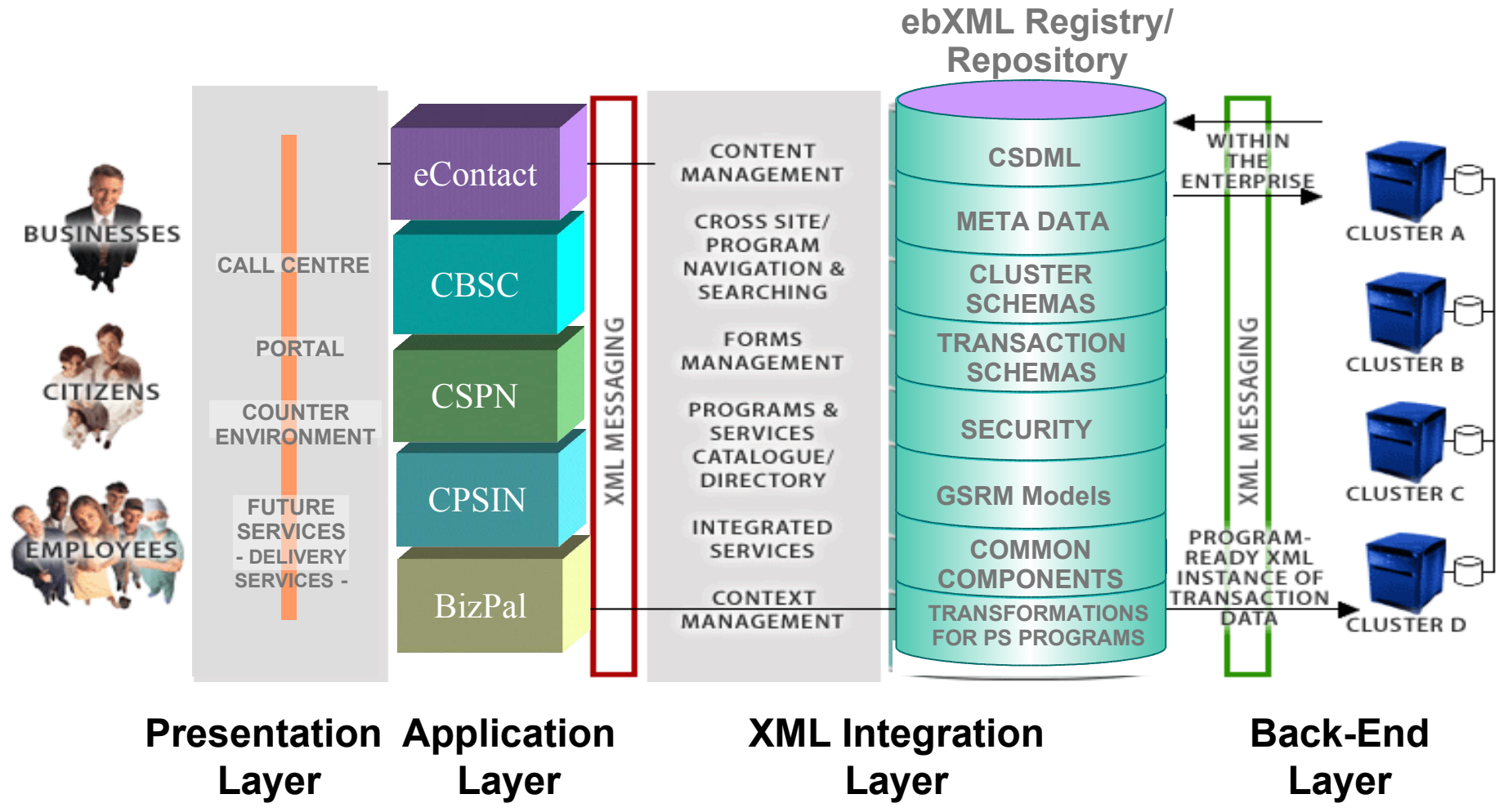
Web Services Use Case with Governance



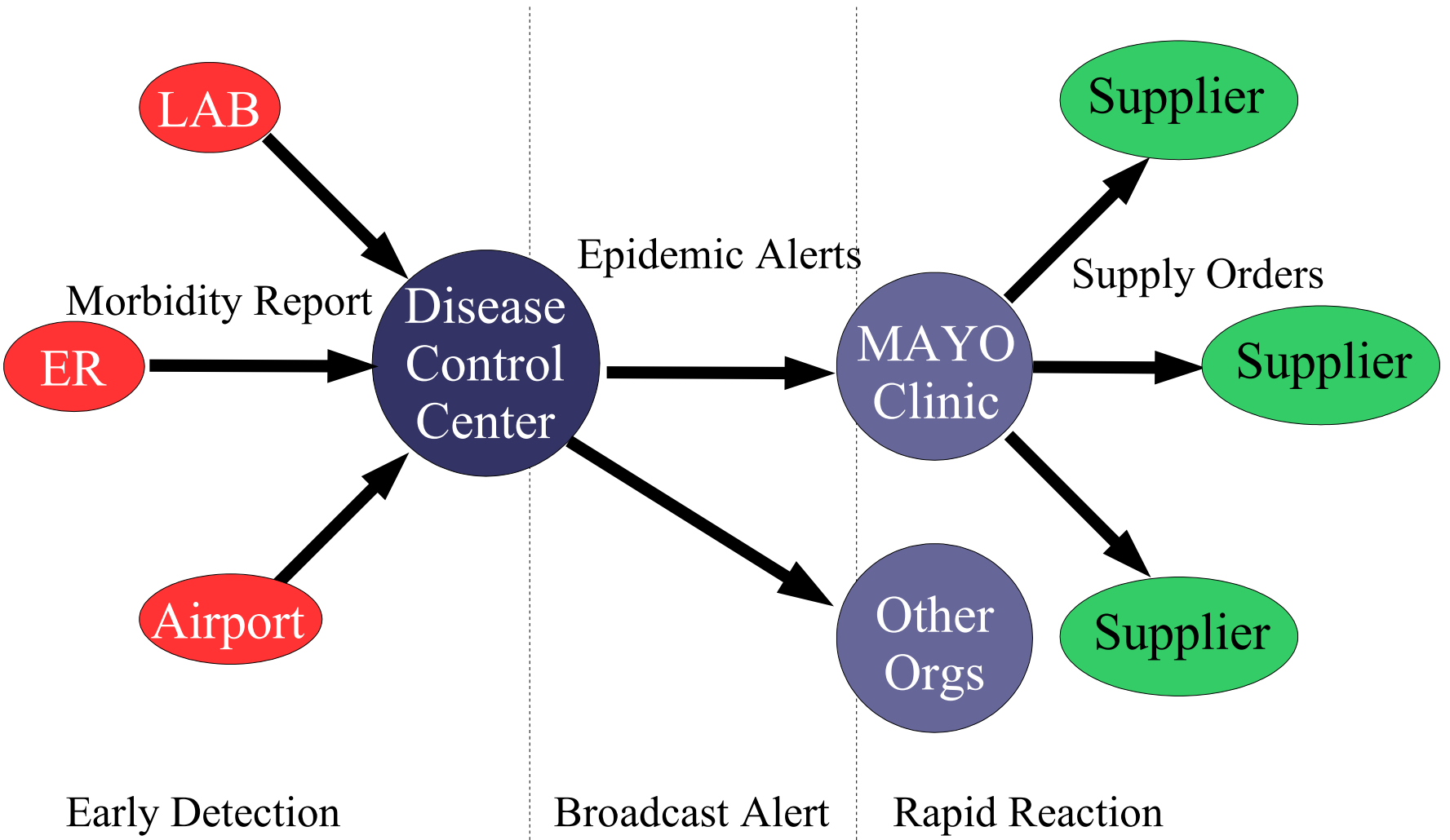
Government Information Management Use

- ▶ Governmental bodies are prominent among ebXML Registry adopters for electronic information management and dissemination
- ▶ Examples include
 - Registry-based environment for assembling XML Schemas and electronic forms (Finland)
 - <http://xml.coverpages.org/ni2004-10-29-b.html>
 - Global reg/rep of eBusiness reference data (UN/CEFACT)
 - <http://www.disa.org/cefact-groups/icg>
 - Standardized clinical document registries (US NIST)
 - <http://ebxmlforum.blogspot.com/2005/05/of-xds-ihe-nist-and-ebxml-registry.html>

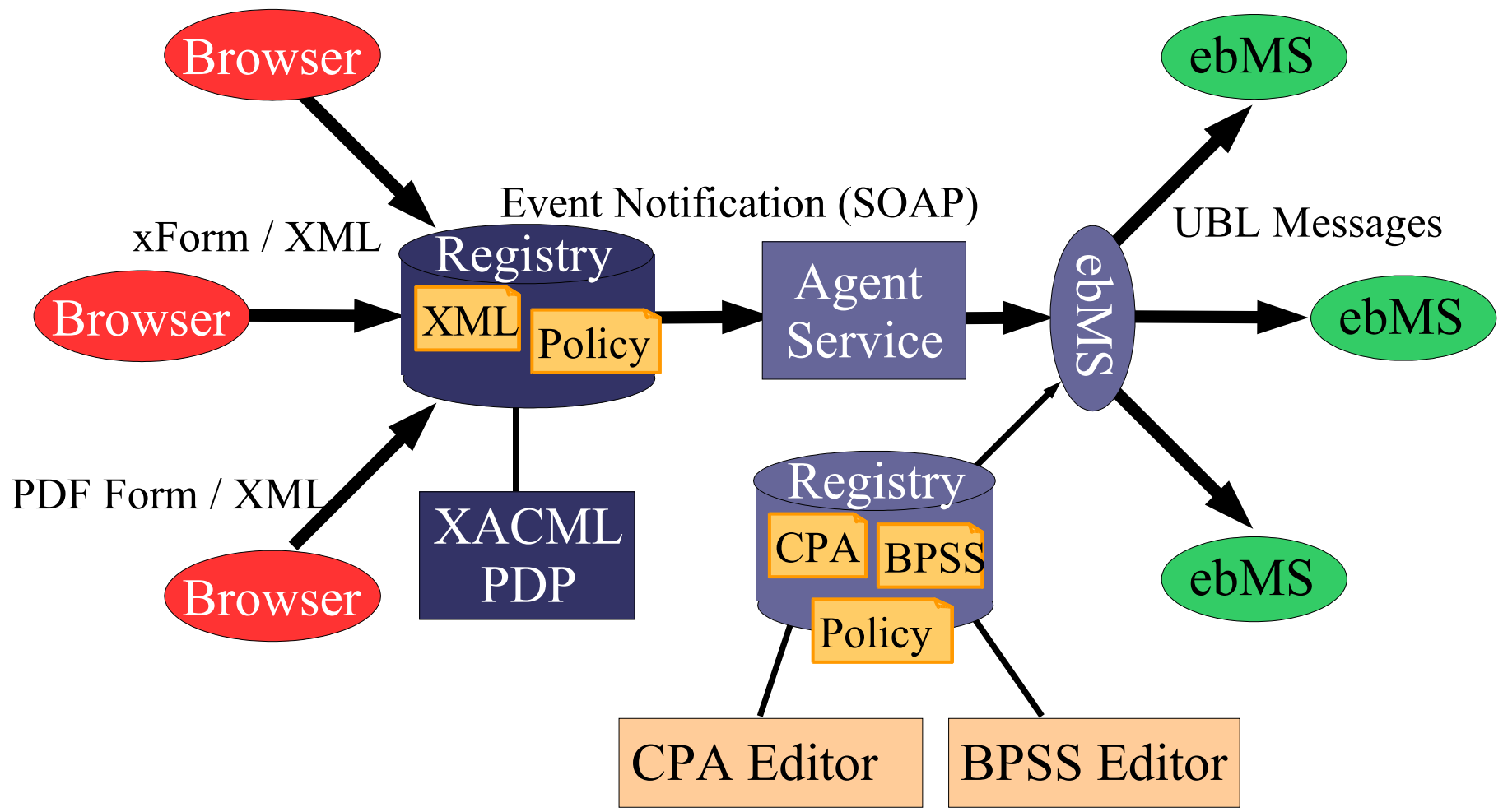
Government of Canada: Pan Canadian Registry Pilot



Epidemic Management Example: Process Scenario



Epidemic Management Example: Technology



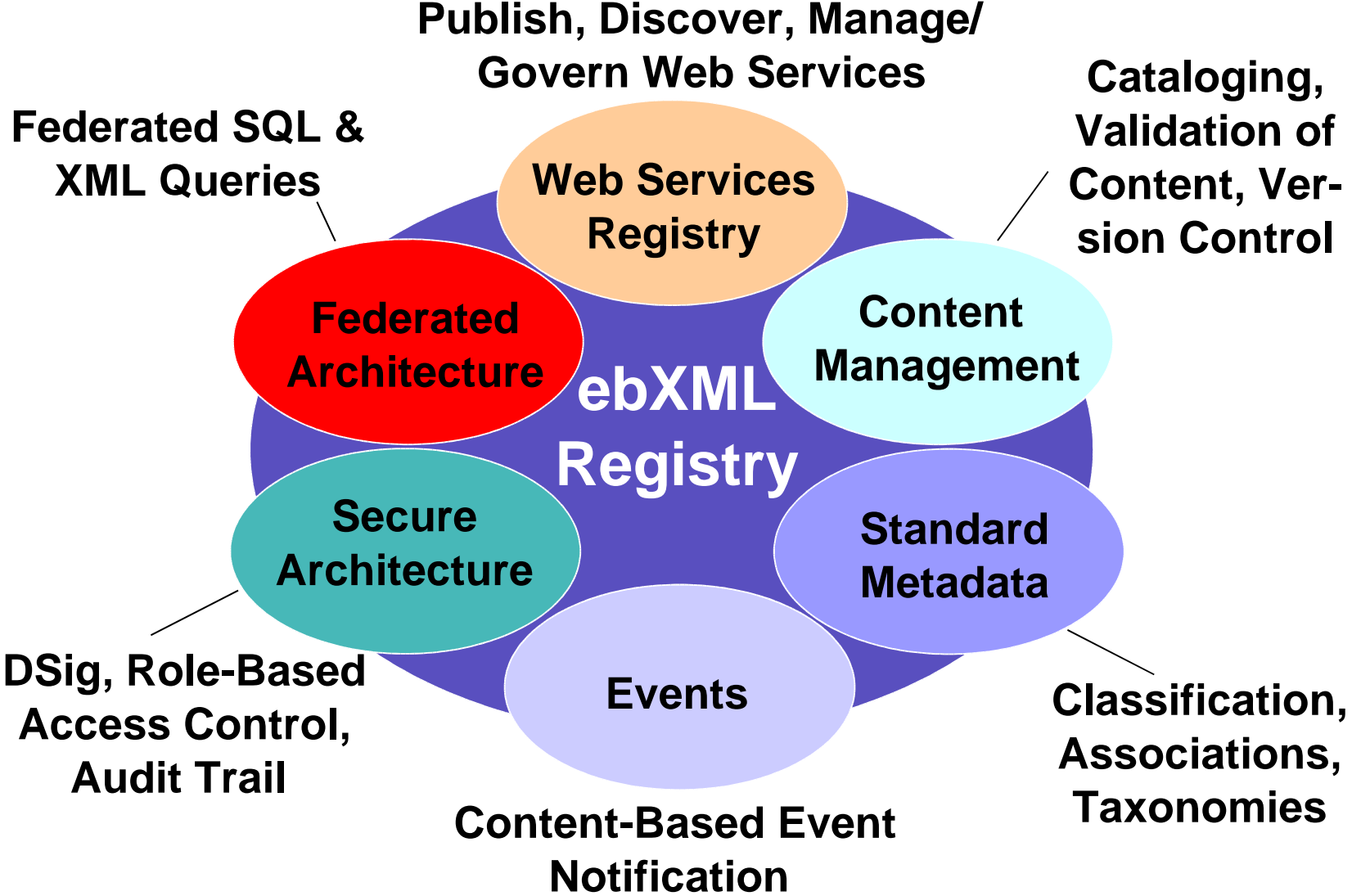
Key Benefits of ebXML Registry

- ◆ Provides standard way to manage information assets
- ◆ Manages user-defined organization of and relationships among content and metadata
- ◆ Enforces user-defined standards for content
- ◆ Includes capabilities for managing and governance of information asset lifecycles

Key Benefits (2)

- ◆ Provides flexible mechanisms for content discovery
- ◆ Manages secure access to information assets
- ◆ Facilitates event-based delivery of information to appropriate personnel or systems
- ◆ Enables integration of information assets across organizational boundaries

Key ebXML Registry Features at a Glance



New Features in ebXML Registry 3.0

| <i>Feature</i> | <i>Description</i> |
|----------------------------------|--|
| HTTP Protocol Binding | <ul style="list-style-type: none">▶ Web Browser client to access to registry using HTTP 1.1 protocol▶ Simple content retrieval |
| Registry Managed Version Control | <ul style="list-style-type: none">▶ Robust version control mechanisms based on the DeltaV/WebDAV protocol |
| Query Enhancements | <ul style="list-style-type: none">▶ Iterative query support▶ Parameterized stored queries▶ Improved Filter Query syntax |
| Content Management Services | <ul style="list-style-type: none">▶ Content validation▶ Content cataloging▶ Content-based discovery |
| Cooperating Registries Support | <ul style="list-style-type: none">▶ Distributed content/metadata▶ Federated queries▶ Replicated content/metadata▶ Object relocation |
| Event Notification | <ul style="list-style-type: none">▶ Publish/subscribe capabilities |

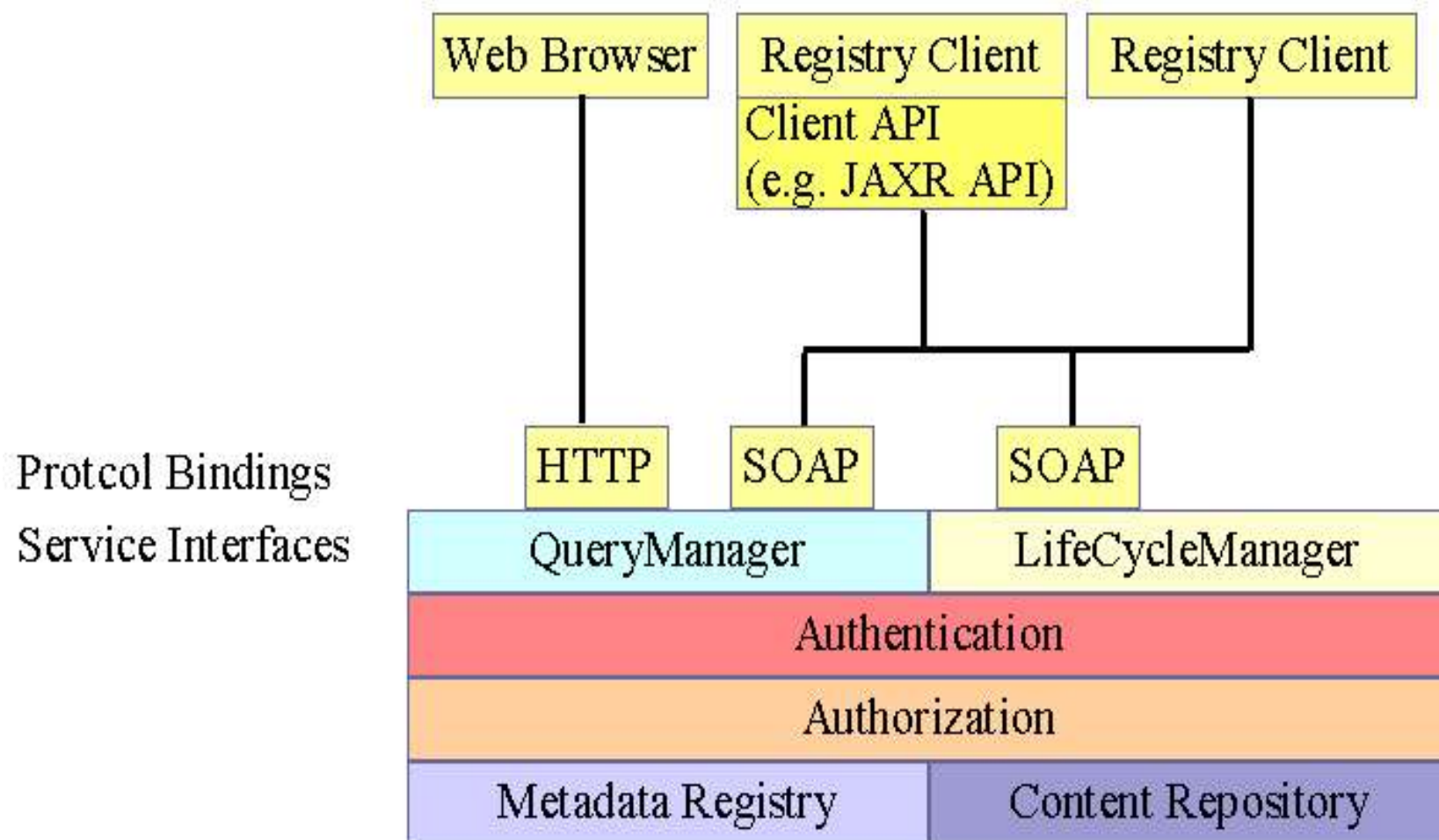
New Features in ebXML Registry 3.0 (2)

| <i>Feature</i> | <i>Description</i> |
|------------------------|---|
| Security Enhancements | <ul style="list-style-type: none">▶ XACML-Based Access Control Model▶ SAML-Based Federated Identity Management |
| Improved Extensibility | <ul style="list-style-type: none">▶ Easier to define new types of requests and responses |
| Improved Identifiers | <ul style="list-style-type: none">▶ Human-Friendly URN-based Identifiers |

Registry Conformance Levels & Profiles

- ◆ Registry Lite: requires core functionality but makes advanced features optional
- ◆ Registry Full: requires advanced features too:
 - Versioning, SQL Query, Stored Query, Iterative Query, Content validation and cataloging, Federation features, Custom Access Control Policies, SAML SSO
- ◆ Registry Profiles
 - Define an application specific standard for using ebXML Registry
 - Extends / restricts ebXML Registry functionality
 - Enables interoperability within specific domains

ebXML Registry Version 3.0: Simplified View of Architecture



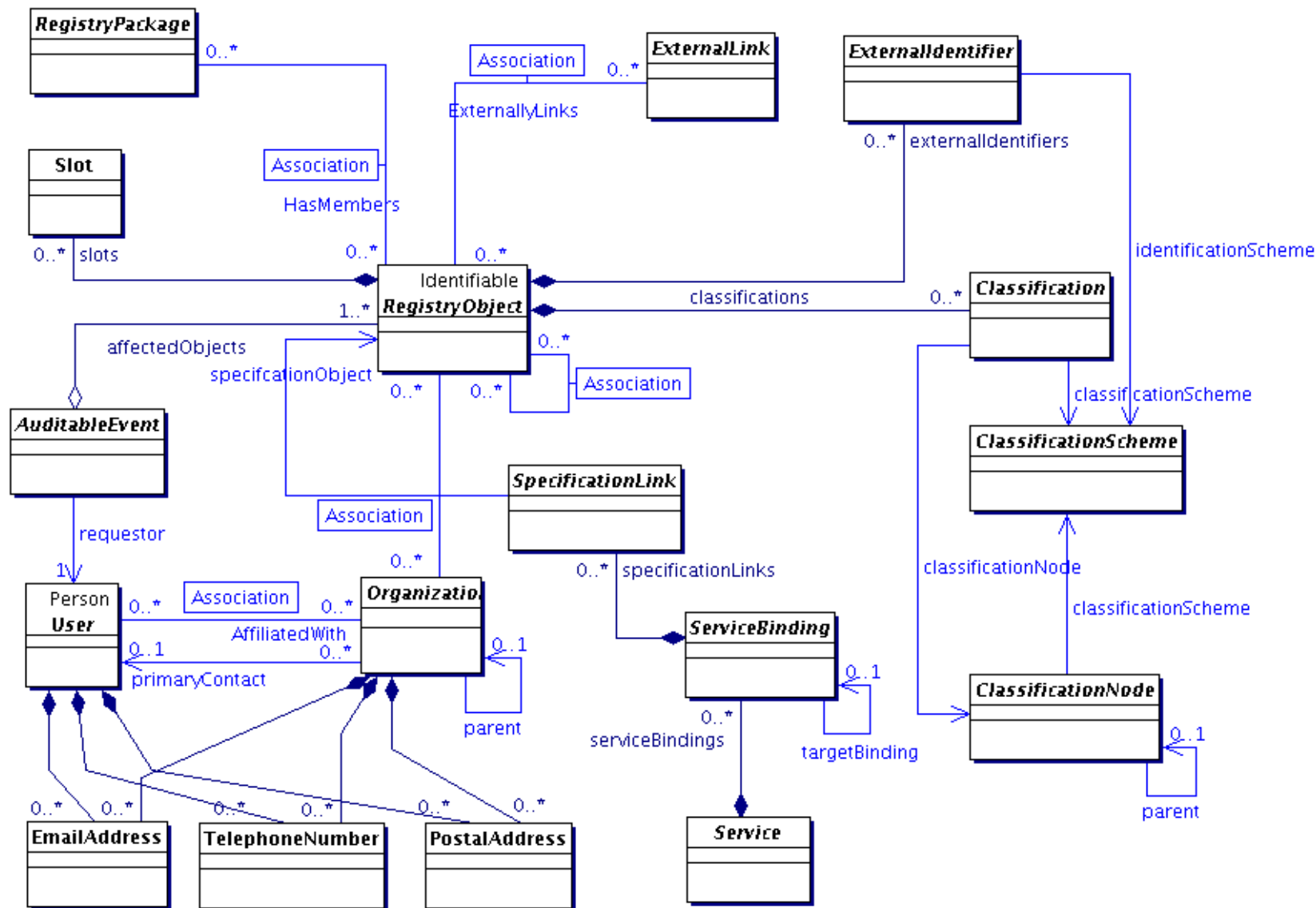
Registry Information Model (ebRIM)

- ◆ Specification that defines the *standard* types of metadata and content that can be stored
- ◆ Terminology:
 - RepositoryItem => content instances
 - *RegistryObject* => metadata instances
 - Objects => RepositoryItem OR RegistryObject

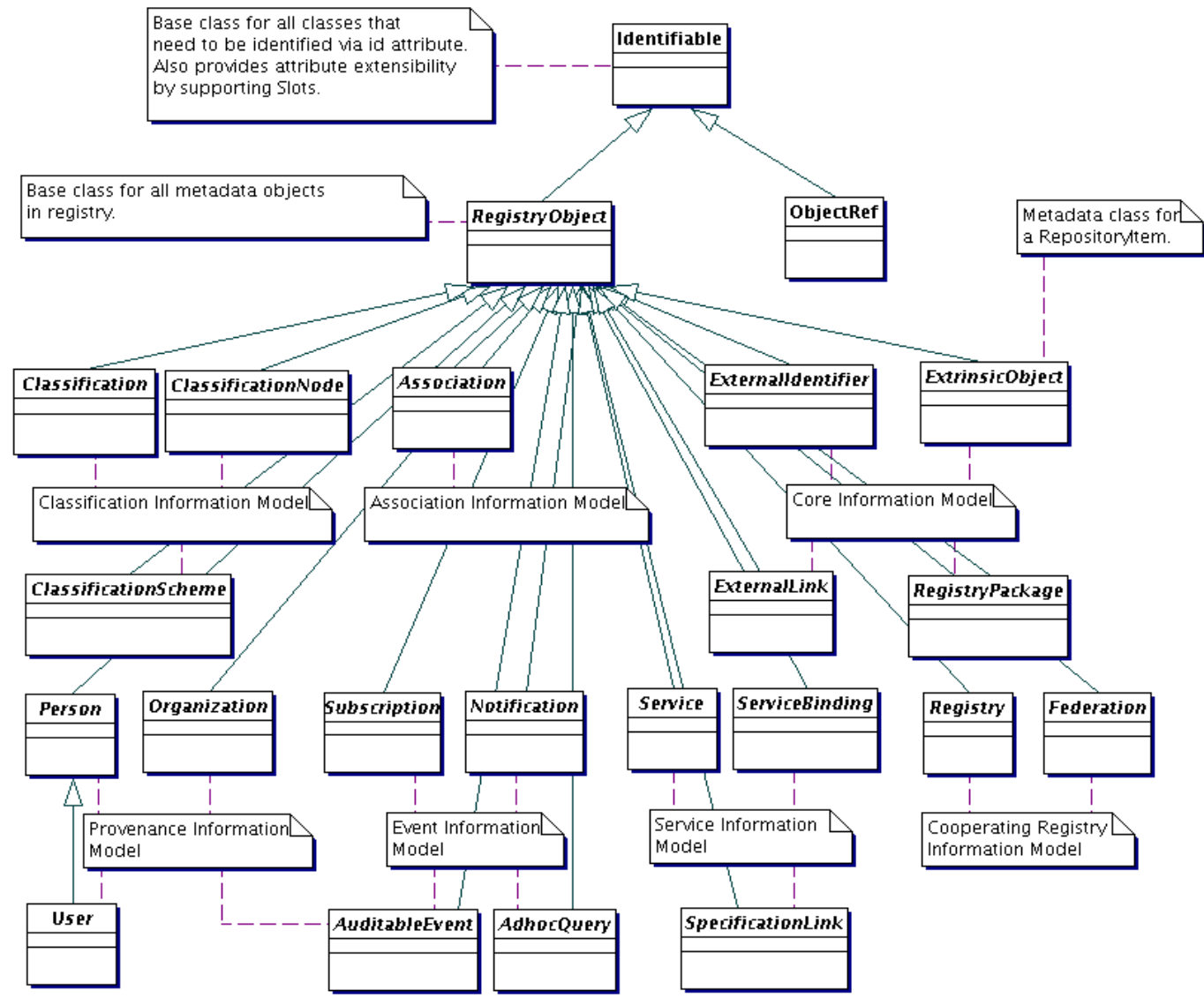
Registry Information Model (2)

- ◆ Uses of RegistryObjects
 - Classification of any type of object
 - Taxonomy hosting, browsing and validation
 - Defining relationships between any two objects
 - Defining identifiers for objects
 - Linking objects to external content
 - Organizing objects using a File/Folder metaphor
 - Defining organization and people information
 - Defining domain specific attribute for objects
 - Defining web service descriptions
 - Enabling discovery of objects

Registry Information Model (3)



Registry Information Model (4)



Registry Services and Protocols (ebRS)

- ◆ Specification that defines the
 - Services an ebXML Registry provides
 - Protocols used by Registry clients to interact with these services
- ◆ Services and protocols include
 - Lifecycle management, query management, content management, event notification, federation management

Registry Services and Protocols (2)

- ◆ Defines *abstract* definition of protocols in UML
- ◆ Defines *normative bindings* of abstract protocols to HTTP and SOAP
- ◆ SOAP binding defined as a web service by WSDL description
- ◆ Request / Response as XML messages
- ◆ Request / Response messages are extensible
 - Enable domain specific profiles to define extensions to registry protocols

Standard Metadata

- ▶ Defined by Registry Information Model
 - Classification of any type of object
 - Taxonomy hosting, browsing and validation
 - Association between any two objects
 - Links to external content
 - File / folder like organization of content
- ▶ User-defined attribute extensibility

Content Integrity and Quality

- ◆ All submissions must be digitally signed by a registered submitter
- ◆ Digital signatures of all submissions must be verified by the registry
- ◆ All registry responses must be signed
- ◆ Content validation enforces semantic correctness using business rules

Validation and Cataloging

- ▶ Validation example: Photo repository
 - A photo must be at least 8 cm x 13 cm
- ▶ Cataloging example: Photo repository
 - Automatically categorize photo as black-and-white or color
- ▶ Application specific, not pre-defined
- ▶ Performed by user-defined web service

Lifecycle Management (LCM)

- ◆ Enables Submit, Update, Approve, Deprecate, Undeprecate, Remove actions on objects
- ◆ Requires authentication and authorization
- ◆ All LCM actions logged in an audit trail
- ◆ Supports automatic versioning of objects
- ◆ Generic: predicate based – not subject based
- ◆ Extensible: application specific slots may be assigned to request / response messages

Query Management (1)

- ◆ Application specific, not pre-defined
 - Ad hoc queries
 - SQL 92 and XML Filter Query syntax
- ◆ Content-based queries
 - Find me all images that are color images
 - Find me all images that are greater than 10 cm x 15 cm
- ◆ Parameterized, stored queries

Query Management (2)

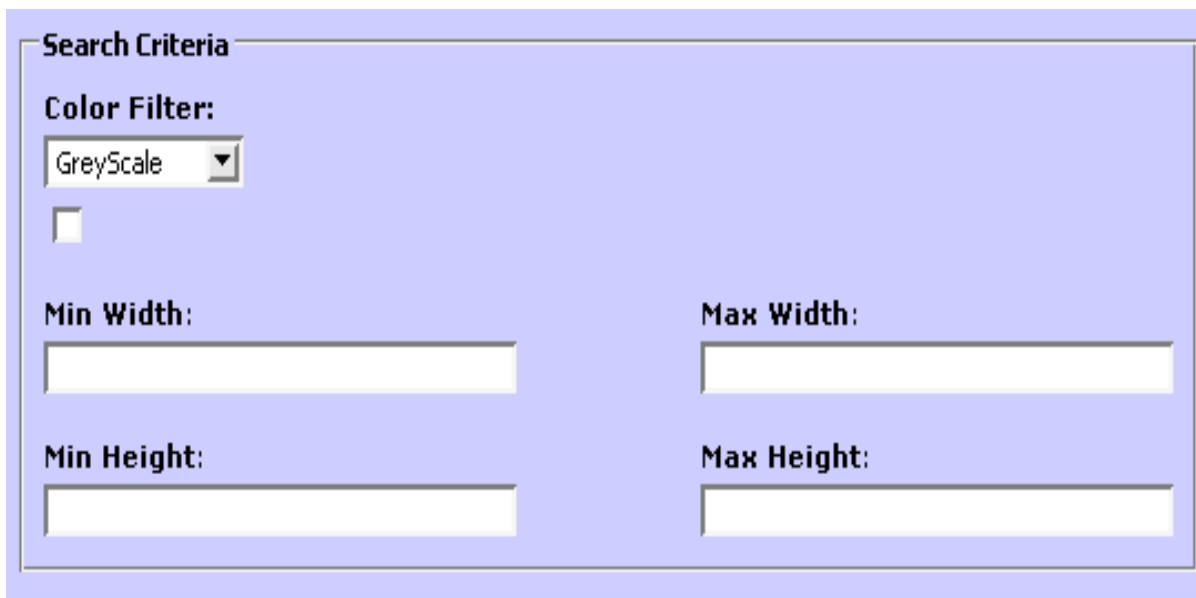
- ◆ AdhocQueryRequest contains:
 - Standard SQL-92 query – Normative SQL Schema
 - XML Filter Query
 - Parameterized query invocation
 - Iterative query parameters: startIndex, maxResults
- ◆ AdhocQueryResponse contains:
 - objects matched by query
 - Iterative query parameters: startIndex, totalResultsCount

Sample Parameterized Stored SQL Query

```
SELECT * from ExtrinsicObject p, Slot w, Slot h,  
Classification c WHERE  
(p.objectType = $objectType) AND  
(w.parent = p, w.name = "width" AND  
w.value >= $minw AND w.value <= $maxw) AND  
(h.parent = p, h.name = "height" AND  
h.value >= $minh AND h.value <= $maxh) AND  
(c.parent = p, c.classificationNode = $colorFilter)
```

Stored Queries and Content Discovery

- ▶ Application specific, not pre-defined
- ▶ Hides query complexity from user
- ▶ Displayed as a simple web form



Search Criteria

Color Filter:
GreyScale ▾

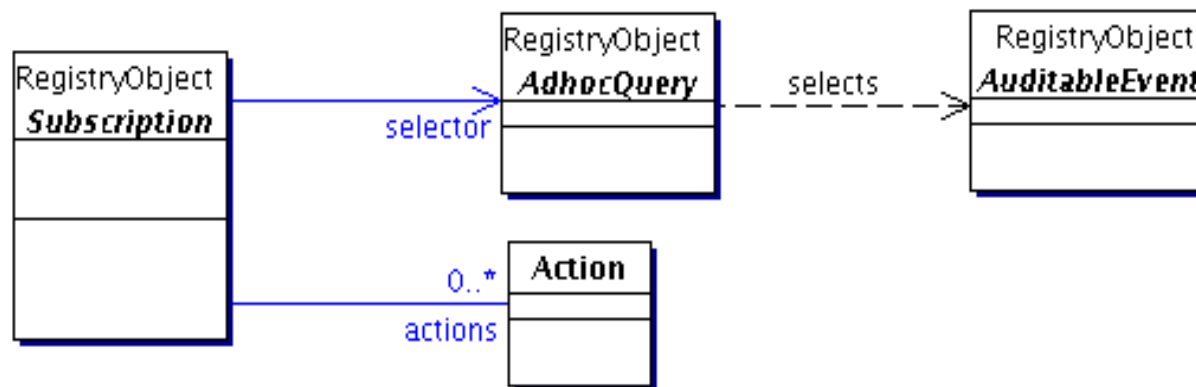
Min Width:

Max Width:

Min Height:

Max Height:

Content-based Event Notification



- ▶ A subscription created by a user specifies:
 - A selector query that selects events of interest
 - Actions that deliver notification of events to interested parties
- ▶ A Notification sent by registry contains:
 - RegistryObjects (or ObjectRefs) that matched selector

Subscription Example: Epidemic Alert

```
<rim:Subscription
  id="urn:canada:cidpc:subscription:EpidemicAlert"
  objectType=
    "urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Subscription"

  selector="urn:canada:cidpc:query:EpidemicAlertQuery">

  <!-- Next endPoint is an email address -->
  <rim:NotifyAction endPoint="mailto:farrukh.najmi@sun.com"
    notificationOption=
      "urn:oasis:names:tc:ebxml-regrep:NotificationOptionType:Objects"/>

  <!-- Next endPoint is a service via reference to its ServiceBinding object -->
  <rim:NotifyAction
    endPoint="urn:canada:cidpc:serviceBinding:EpidemicAlertListenerBinding"
    notificationOption=
      "urn:oasis:names:tc:ebxml-regrep:NotificationOptionType:Objects"/>
</rim:Subscription>
```

Selector Query Example

//Match all changes to objects of type EpidemicAlert

//that are classified by geography of Alberta

SELECT * from ExtrinsicObject p,

Classification c WHERE

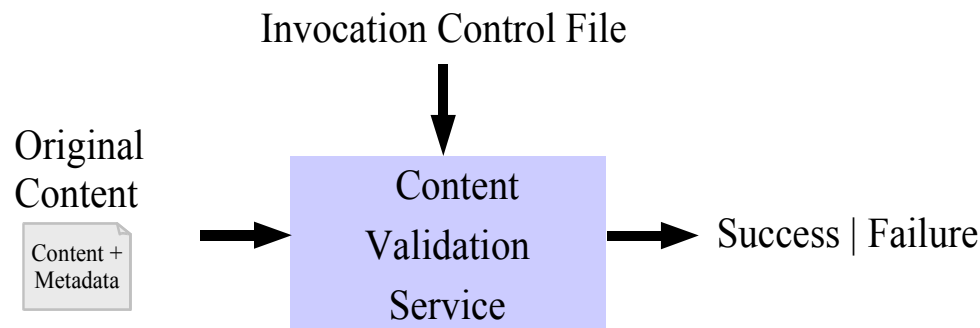
(p.objectType = "urn:canada:cidpc:objectTypes:EpidemicAlert")

(c.parent = p, c.classificationNode = <Alberta>)

Content Management Services

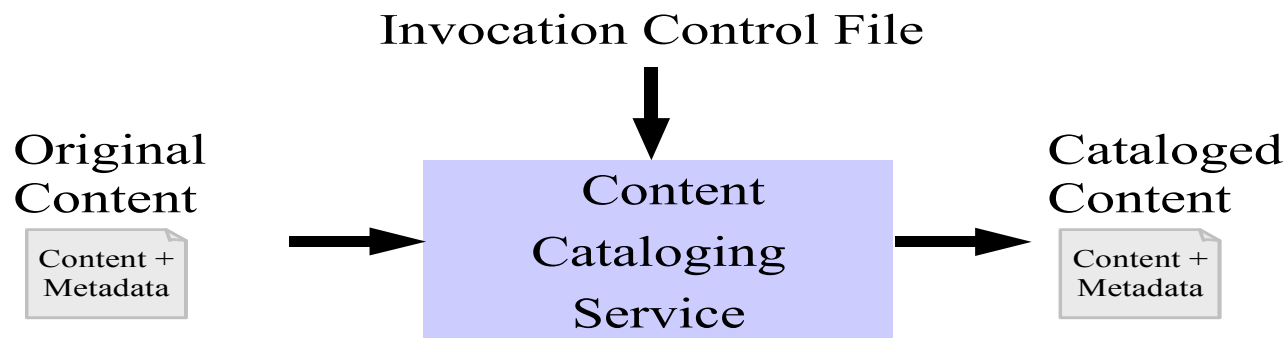
- ◆ Service endpoint described by WSDL
- ◆ Requires programming the Service
 - Developed using a WS developer kit like JWSDP
- ◆ Requires publishing the Service description in the registry
 - Need UI tool to make this easier
- ◆ Automatically invoked by registry when content is published that matches the Service
- ◆ Content validation and cataloging services defined by ebRS specification

Content Management: Validation



- ◆ Input is the submitted objects
- ◆ Invocation control files is content specific
- ◆ Service determines if input is valid
- ◆ Input rejected if validation results in Failure

Content Management: Cataloging



- ◆ Input is the submitted objects
- ◆ Invocation control files is content specific
 - e.g. XSLT for XML Cataloging Service
- ◆ Outputs new objects to be stored in registry

Secure Architecture

- ◆ Digital signature-based authentication for specified functions (e.g., publishing)
- ◆ Role Based Access Control (using XACML)
- ◆ Event archiving - complete audit trail
- ◆ SAML 2.0-based Single Sign-On (SSO)

Security: Authentication

- ▶ Signed SOAP Messages for request/response
 - WSS Soap Message Security 1.0 compliant
 - User's public key included in message
 - User's public key **MUST** be known to registry
 - Registry performs authentication based on public key
- ▶ HTTP/S transport security
 - Secures communication to registry over HTTP
 - SSL Client cert authentication done by container
 - Require CA issuer to be added to container's trust store

SOAP Message Security: Simple Example

```
<soap:Envelope>
  <soap:Header>
    <wsse:Security>
      <wsse:BinarySecurityToken> ...public key...</wsse:BinarySecurityToken>
      <ds:Signature>
        <ds:SignedInfo>...</ds:SignedInfo>
        <ds:SignatureValue>PipXJ2Sfc+LTDnq4pM5JclYt9gg=</ds:SignatureValue>
        <ds:KeyInfo>
          <wsse:SecurityTokenReference .../>
        </ds:KeyInfo>
      </ds:Signature>
    </wsse:Security>
  </soap:Header>
  <soap:Body wsu:Id="TheBody">
    <lcm:SubmitObjectsRequest .../>
  </soap:Body>
</soap:Envelope>
```

Security: Role Based Access Control

- ◆ XACML 1.0 based access control policies
- ◆ Default Access Control Policy (ACP)
 - Any one can read object
 - Owner can modify or delete object
 - RegistryAdministrator role can modify or delete object
- ◆ Custom ACP may be assigned to an object to restrict or relax access to specific identities, roles and groups
- ◆ ACP creation is manual
 - Planned ACP Editor will make it easier in future

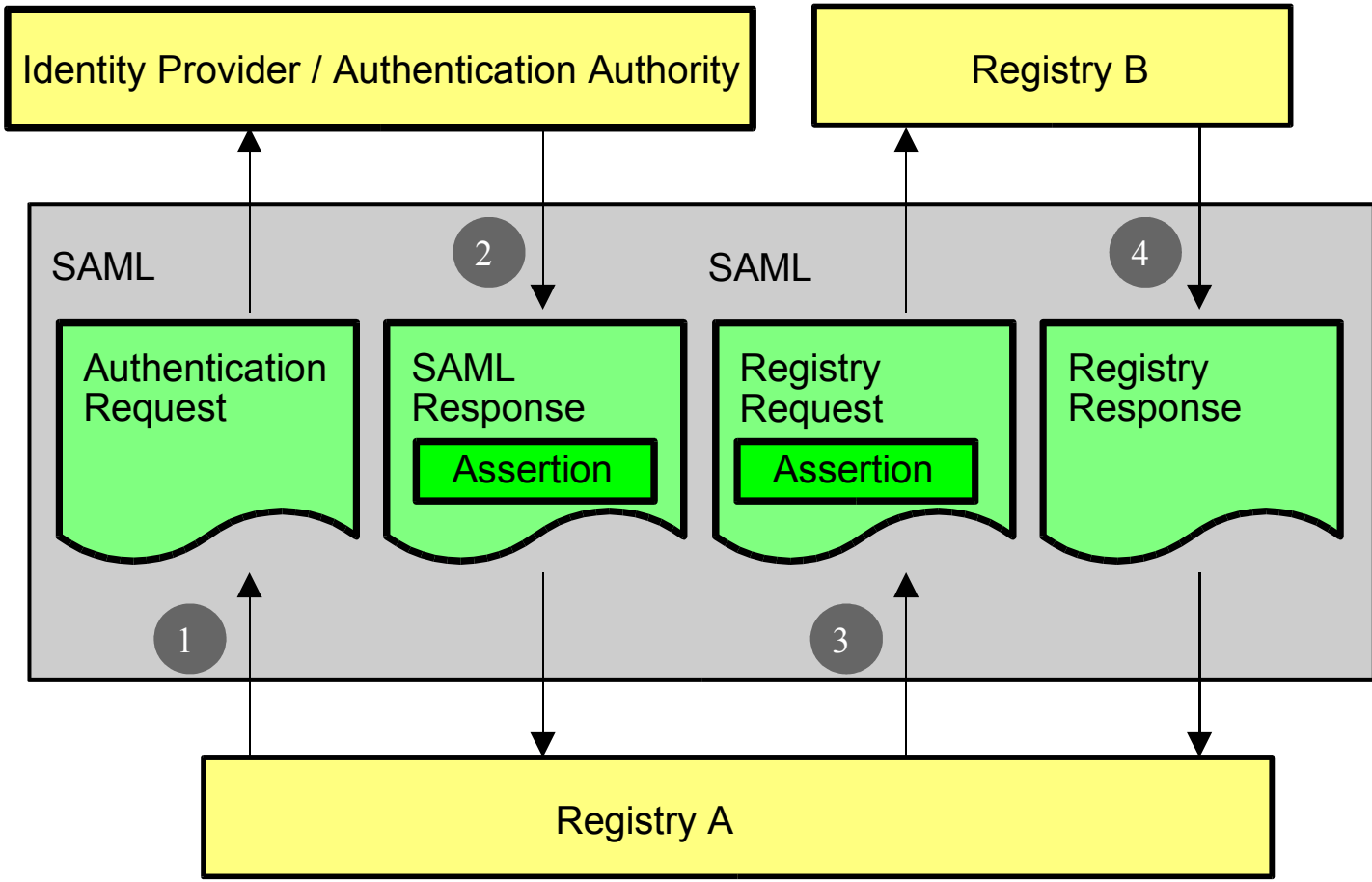
Security: Role Based Access Control (2)

- ◆ Extensible Roles and Groups
 - Roles defined by SubjectRole scheme
 - Groups defined by SubjectGroup scheme
 - Roles/Groups assigned to users by classifying their User object with role or group ClassificationNode
 - Only RegistryAdmin role can assign roles

Security: SAML Single Sign-On (SSO)

- ◆ Enables registry to work with any existing SAML 2 based enterprise user database
 - Eliminates need for duplicated user information
- ◆ Registry no longer stores User instances
- ◆ Users register with enterprise user database and not registry
- ◆ Planned but not implemented in freebXML Registry project

Security: SAML Single Sign-On (SSO)



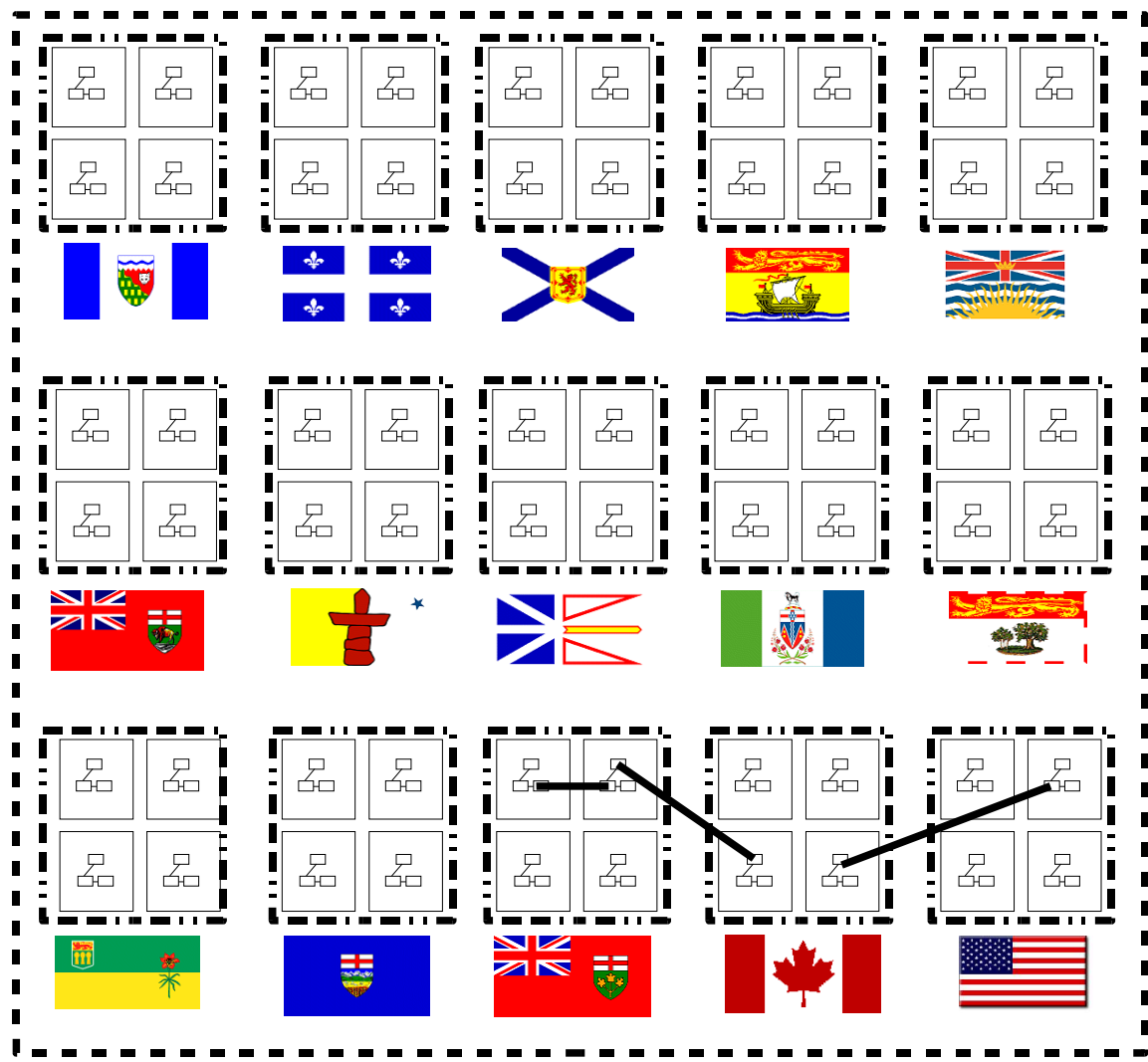
Role Based Access Control

- ◆ Determines WHO can do WHAT with WHICH registry resource
- ◆ Resource may be any metadata or content
- ◆ Custom Access Control Policies use XACML 1.0 syntax

Content Based Event Notification

- ◆ Subscriber subscribes using Selector Query
- ◆ Subscriber receives event notifications using
 - SOAP interface: NotificationListener
 - Email
- ◆ Notification granularity is configurable

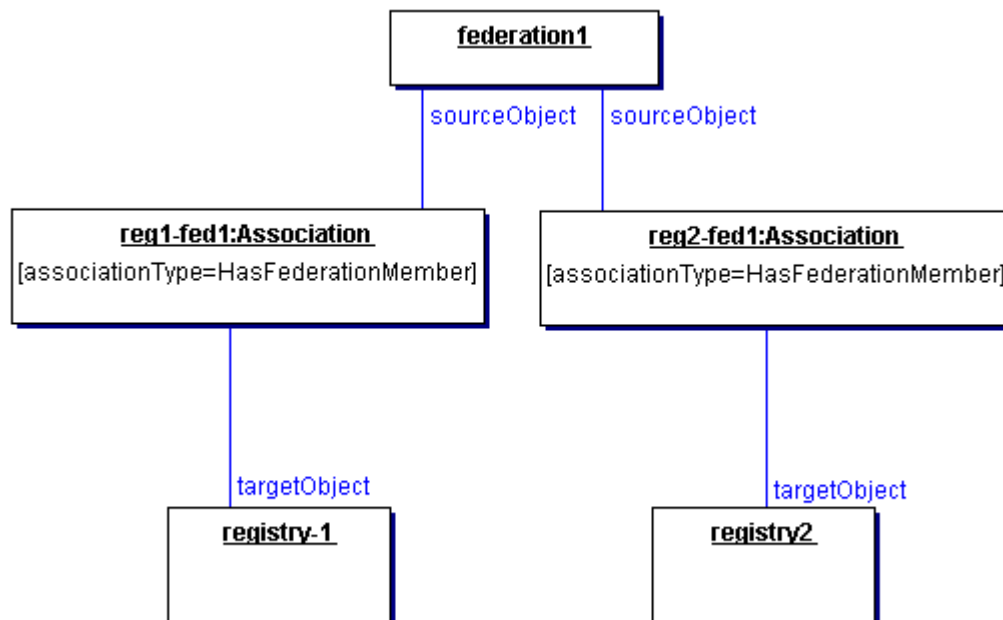
Federated Architecture



Federated Architecture

- ◆ Registry federation and federated query
- ◆ Inter-registry object references
- ◆ Object relocation
 - From one registry to another
 - From one user to another
- ◆ Object replication

Cooperating Registries: Federation Description

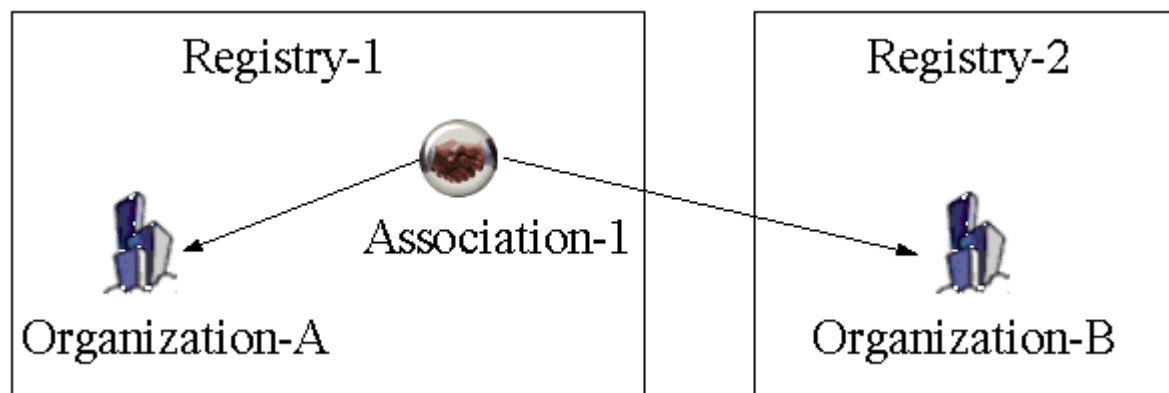


- ◆ Federation description published using LCM
- ◆ Kept synchronized using Event Notification
- ◆ Federation members SHOULD be part of a circle of trust and allow SSO across members

Cooperating Registries: Federated Queries

- ♦ Looks like a normal query but specifies:
federated="true"
- ♦ May be sent to any federation member
- ♦ Searches all federation members for matches
 - Return results from multiple registries
- ♦ Enforces loose consistency (partial results ok)
- ♦ May optionally specify target federation
 - In case registry is member of multiple federations

Cooperating Registries: Remote Object Referencing

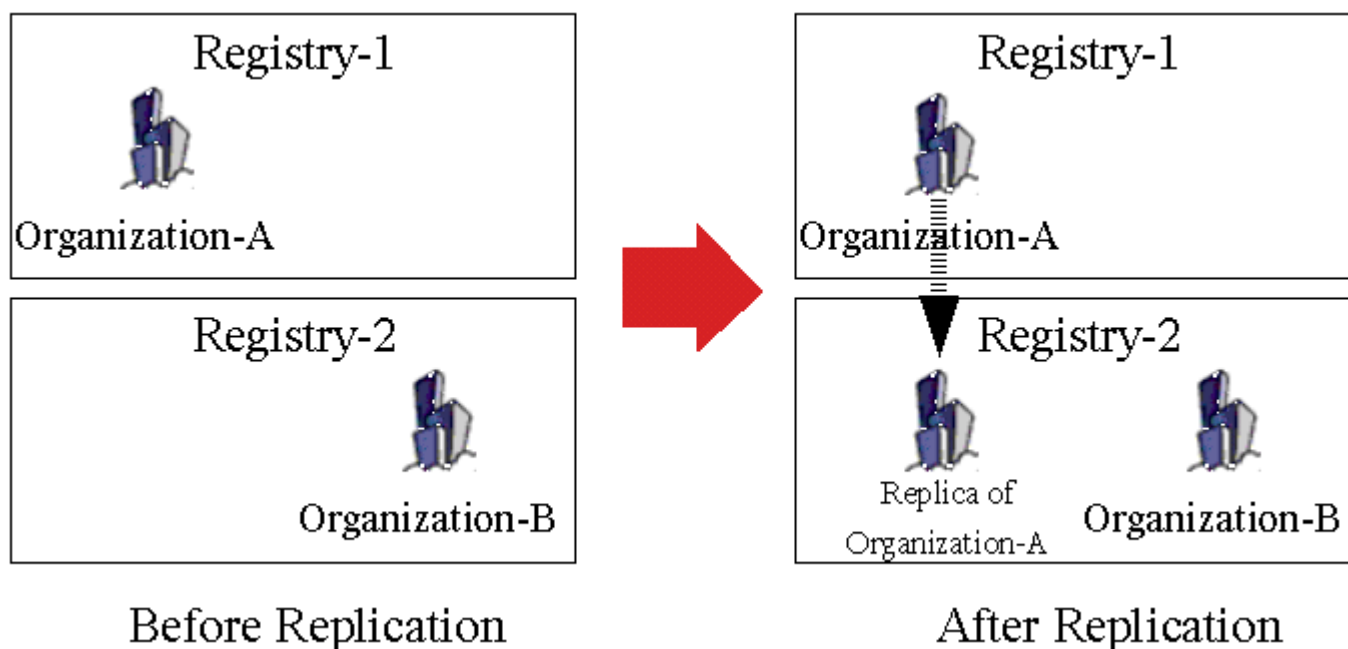


- Enables an object in one registry to reference an object in another registry

```
<Association id="urn:registry1:assoc1"
  sourceObject="urn:registry1:orgA"
  targetObject="urn:registry2:orgB"/>
```

```
<ObjectRef
  home="urn:registry2"
  id="urn:registry2:obj2"/>
```

Cooperating Registries: Replication



- ◆ Created explicitly by submitting a remote ObjectRef
- ◆ Created implicitly by registry when processing federated queries
- ◆ Replicas synced using event notification

Summary: Key Features and Benefits

- ◆ **Standards:** Provides standards-based way to manage information assets
- ◆ **Classification and affiliation:** Manages user-defined organization of and relationships among content and metadata
- ◆ **Validation:** Enforces conformance of content to user-defined standards
- ◆ **Lifecycles:** Governance capabilities for managing information asset lifecycles

Summary: Key Features and Benefits (2)

- ◆ **Query:** Provides flexible mechanisms for content discovery
- ◆ **Security:** Manages secure access to information assets
- ◆ **Event notificaton:** Facilitates event-based delivery of information to appropriate personnel or systems
- ◆ **Federation:** Enables integration of information assets across organizational boundaries

ebXML Registry in Action (1)

- ◆ Government adoption
 - Finland: XML Schemas and eForms
 - UN/CEFACT, EDIFrance: eBiz reference data
 - US NIST: Clinical documents
 - Taiwan: eGovernment Platform
 - Korea: Central Registry and Repository (REMCO)
 - Canada: Federated SOA Registry

ebXML Registry in action (cont.)

- ◆ Commercial implementations
 - Adobe: Forms registry product
 - Infravio: Web services registry product
 - Digital Artefacts: Registry product
 - ebXMLsoft: Registry product
 - KTNET: Registry service

Future Directions

- ◆ Interop testing
- ◆ Conformance specification
- ◆ Semantic content management
- ◆ More profiles- e.g. Dublin Core
- ◆ More protocol bindings – SMTP, RM, ebMS
- ◆ ...

Summary

- ◆ ebXML Registry is a standard for secure, federated information management
- ◆ One registry supporting both SOA and eBusiness use cases
- ◆ Extensibility enables domain-specific profiling of Registry capabilities
- ◆ It is being deployed in diverse applications by organizations worldwide

Related Links

- ▶ Cover Pages Article:
<http://xml.coverpages.org/ni2005-02-14-a.html>
- ▶ ebXML Registry Meta Links page:
<http://ebxmlrr.sourceforge.net/tmp/ebXMLRegistryLinks.html>



**ebXML Registry 3.0:
An Overview
Finish**