Resource Virtualization and Management Using WSRF/WSN and WSDM

Latha Srinivasan
Hewlett-Packard

GlobusWorld 2005
February 9, 2005

Goals

- Overview of the concepts behind management and virtualization and how WSRF, WSN and WSDM fit
- Examples to illustrate how to apply WSRF, WSN and WSDM
- Places to get additional information

Agenda

- Management and Virtualization
- WSRF as a virtualization engine
- Managing a virtual resource
- WSDM – MUWS & MOWS
- Management tool integration
- HP-Globus open source contributions to Apache
- Q & A

Management

- What needs to be managed:
  - Hardware & software resources:
    - Computer, disk, network, database, operating system, etc.
  - Logical (sometimes transient) resources:
    - Print job, executing application, etc.
- Resources must:
  - Operate correctly
  - Meet availability & performance criteria
- Management:
  - Monitor status & performance
  - Respond to internal & environmental changes
  - Initiate routine operations: configure, start, stop, tune, …
  - Maintain the resource & its environment

Manageability

- Definition: The ability of a resource to be managed
- Manageability interfaces support common operations:
  - Control: start, stop, etc.
  - Monitoring: status & performance
- Manageability standards:
  - Specify standard interfaces for common operations
  - Manage resources in a uniform, interoperable fashion
- Problem:
  - Existing interfaces are generally resource-specific
  - Difficult to manage disparate resources with inconsistent interfaces
- Solution:
  - Virtualization!

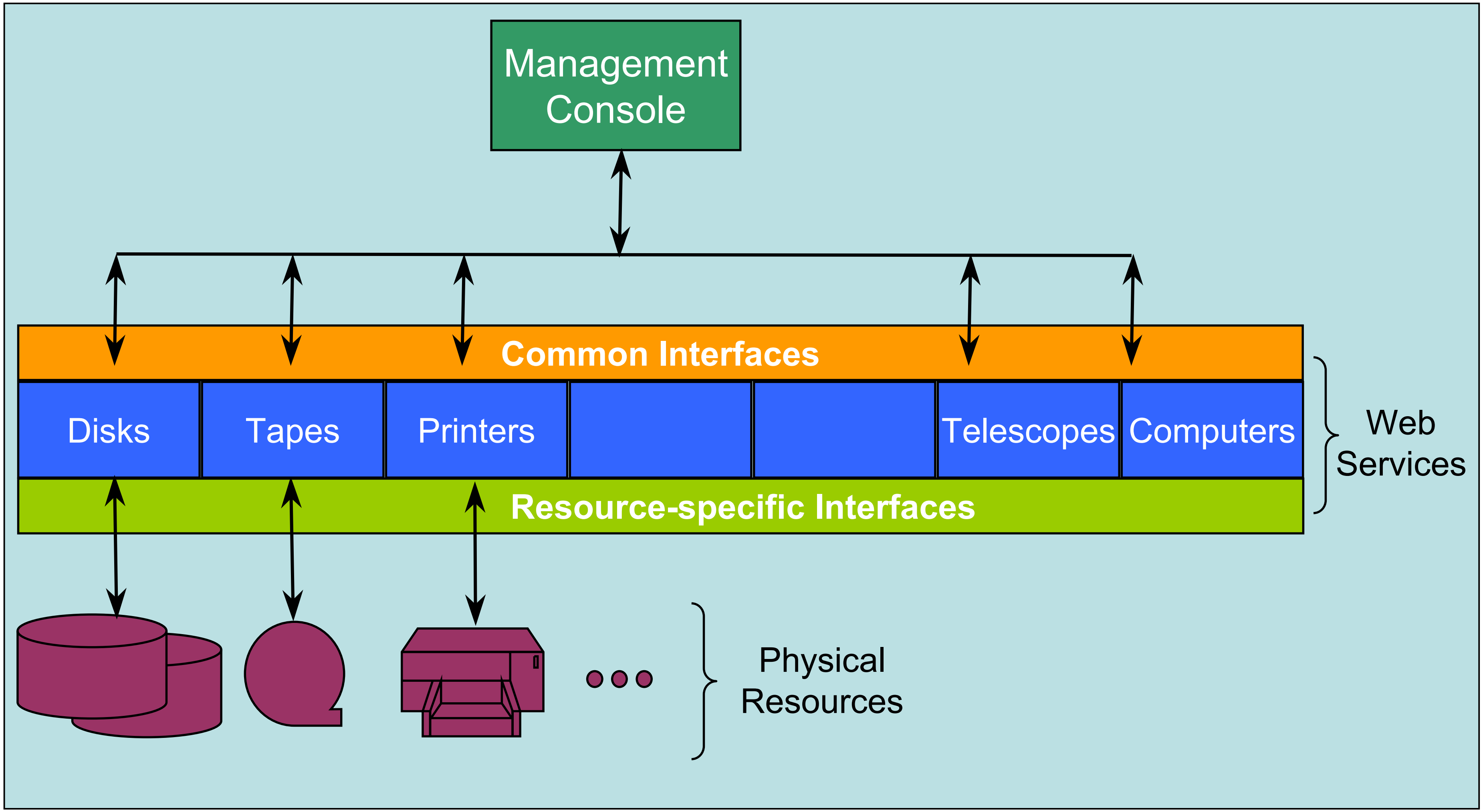
Virtualization

- Definition:
  - Make a common set of abstract *interfaces* available for a set of similar *resources*
  - …allowing them to be viewed and/or manipulated in a common way
  - …hiding differences in their properties and operations

Source: OGSA Glossary v1.0

- Key benefits:
  - Common, well-published, interoperable interfaces that can be used in Grid, Utility Computing, etc.
  - Easier integration of management interfaces across virtualized resources
- Web services technology is one enabler of virtualization

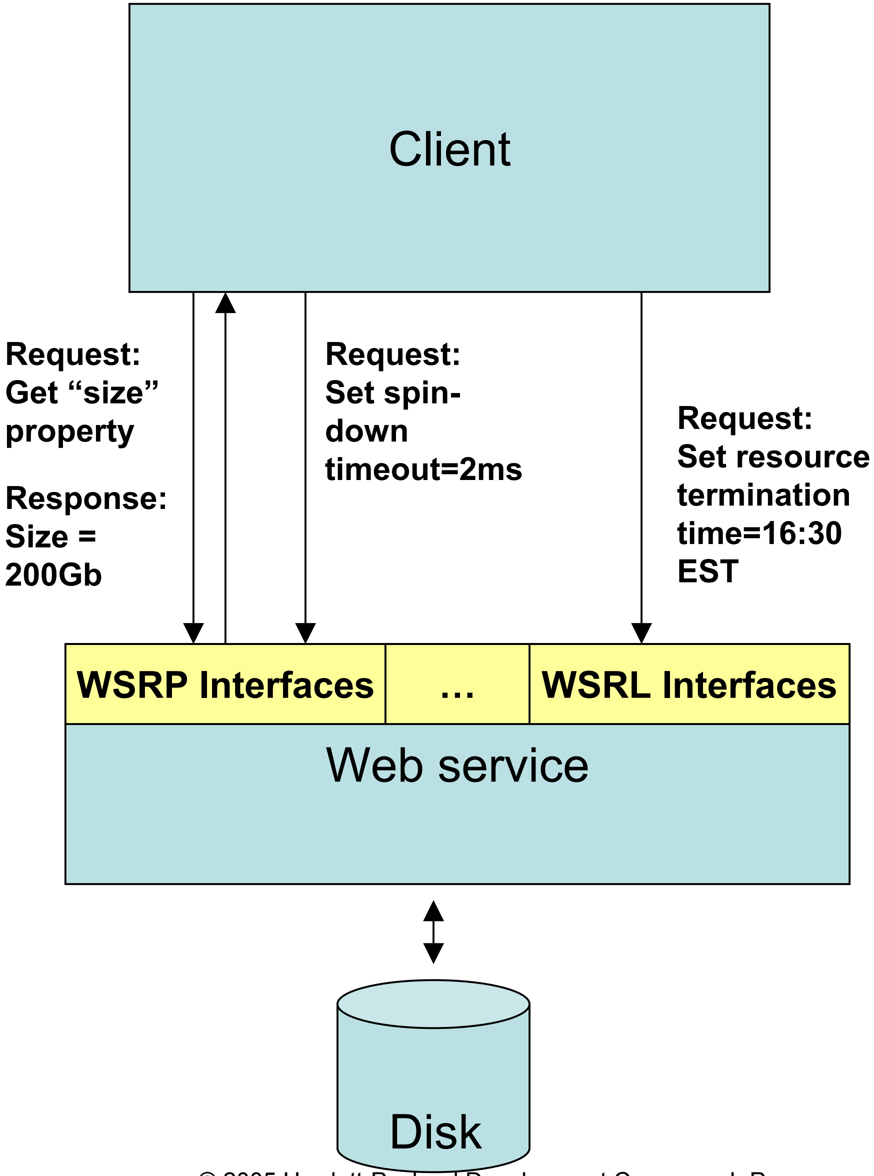
Resource Virtualization



WS-Resource Framework (WSRF)

- Describe common ways to represent, access and modify resources
- Specifications under development in the OASIS WSRF Technical Committee
- Evolution of GGF’s OGSI, since deprecated; convergence of Grid and Web Services
- Core concept is the WS-Resource:
  - Representation of state in a Web services context
  - Provides access mechanisms for resource state
  - Manage the lifetime of state
  - Aggregate “stateful” resources
  - Describe faults related to resource manipulation

WSRF: an illustration



Client

Request:
Get “size”
property

Response:
Size =
200Gb

Request:
Set spin-down
timeout=2ms

Request:
Set resource
termination
time=16:30
EST

WSRP Interfaces

…

WSRL Interfaces

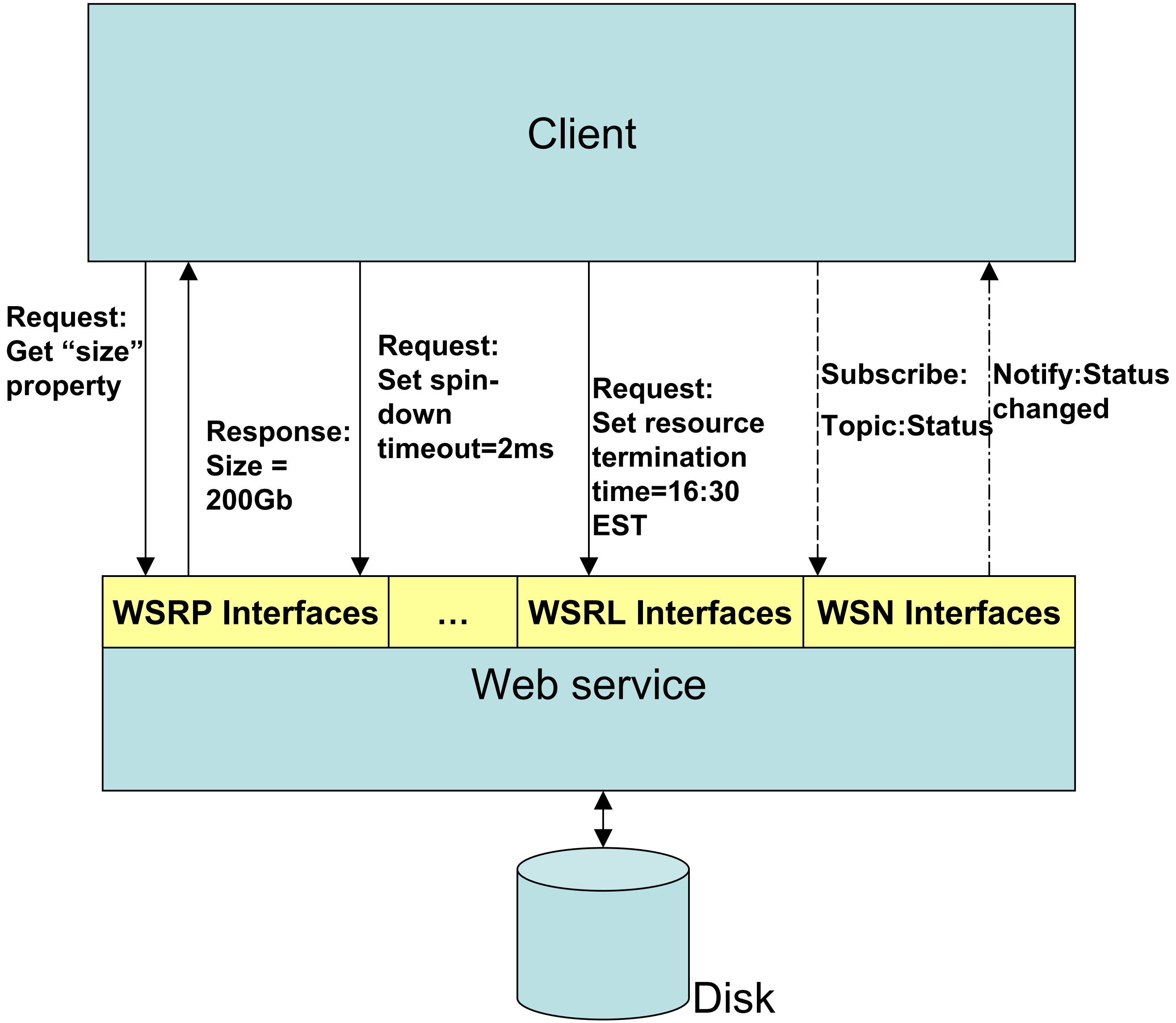
Web service

Disk

WS-Notification (WSN)

- Common mechanisms to represent Web Services Events
- Specifications under development in the OASIS WSN Technical Committee
- Successor to the events portion of OGSI
- Utilizes the WS-Resource concepts to:
  - Create, manage subscriptions
  - Define basic notification message format
  - Define “topics” for event notification

WSN: an illustration



WSRF & WSN: Enablers of Virtualization

- WSRF provides a set of foundational specifications to build higher level services:
  - Standard interface definition and service invocation constructs provided by XML and WSDL
  - Standard, interoperable constructs to model real life entities (physical and logical) as WS-Resources
  - Formal methods to access and modify properties modeled in WS-Resources
  - Standard ways to manage lifetime of WS-Resources
- WSN builds upon the WSRF specifications to provide mechanisms for notifications, subscriptions and topics.

WSRF & WSN Example: Disk

- Identify attributes of the disk to be captured
  - E.g,: Number of Blocks, Blocksize, Filesystems
- Create XML schema definitions for the attributes
  - E.g:
    ```
    <element name =“size” type=“xsd:int”>
      <element name =“Manufacturer”
    type=“xsd:string”>
      …
    ```
  - Disk properties that we want to expose are in the <schema> section of the WSDL file
- WSDL PortType element refers to the resource we want to virtualize:
  - E.g:
    ```
    <portType name=“DiskPortType”
    wsrf-rp:ResourceProperties=“tns:DiskProperties”>
    ```
  - portType has an attribute that references the Resource Properties document

WSRF & WSN Example (cont’d)

- ResourceProperties example:

```
<element name="DiskProperties">
  <complexType>
    <sequence>
      <element ref="tns:Manufacturer" minOccurs="0" />
      <element ref="tns:size" />
      <element ref="tns:NumberOfBlocks" />
      <element ref="tns:FileSystem" minOccurs="1"
        maxOccurs="2" />
      <element ref="tns:TopicSpace" minOccurs="1"
        maxOccurs="unbounded" />
    </sequence>
  </complexType>
</element>
```

WSRF & WSN Example (cont'd)

- Access, update and query disk properties:
  - GetResourceProperties, GetMultipleResourceProperties
  - SetResourceProperties
  - QueryResourceProperties

  Note: The actual "backend" operations of modifying, accessing and querying disk properties must be implemented by the user.
- Custom operations for disk:
  - Assign to a storage array, unassign,…
- Set up notifications:
  - Subscribe, Manage, Cancel
- Topics:
  - Resource Property additions, removals, modifications
  - Others

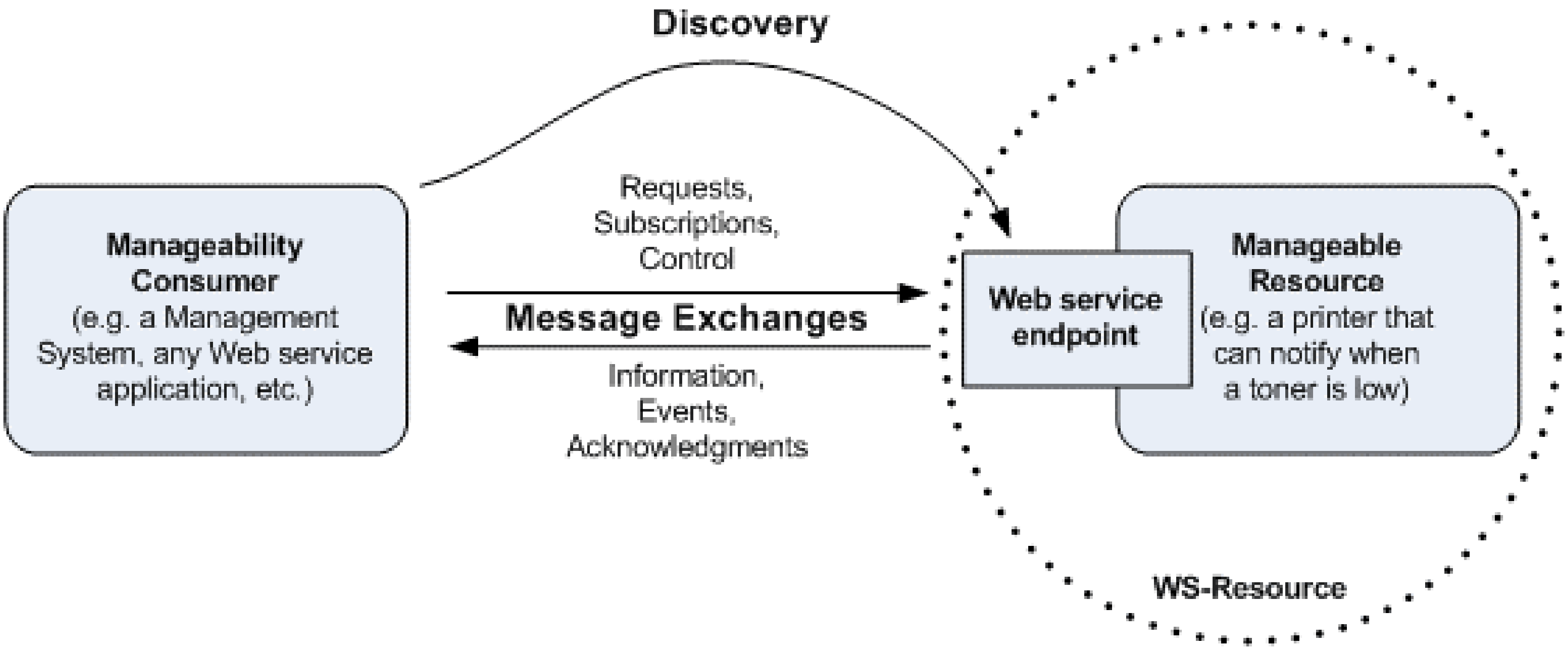
Managing a Virtualized Resource

- Define manageability interfaces to support common operations:
  - Get & Set Operational Status:
    - for monitoring & control operations
  - Get & Set Metrics:
    - for performance operations
  - Get & Set Relationships:
    - to aid discovery
- Define manageability attributes & metadata
  - Identifier, security policies
- Define management events
  - Relationship added, Resource created

Web Services Distributed Management

- Web Services Distributed Management (WSDM):
  - Specifications addressing Web Services management for distributed resources
  - Developed by OASIS WSDM Technical Committee (TC)
- HP contributed Web Services Management Framework to the TC; other contributions by were also made
- Two specifications:
  - Management <u>using</u> Web Services (MUWS)
  - Management <u>of</u> Web Services (MOWS)
- Version 0.5:
  - Released April 2004
- Version 1.0:
  - a major update
  - Committee draft released December 2004

WSDM


Discovery
Manageability Consumer
(e.g. a Management System, any Web service application, etc.)
Requests, Subscriptions, Control
Message Exchanges
Information, Events, Acknowledgments
Web service endpoint
Manageable Resource
(e.g. a printer that can notify when a toner is low)
WS-Resource

Source: WSDM MUWS 1.0 Specification

WSDM-MUWS

- Management Using Web Services
- Foundational specification of WSDM (i.e. used by MOWS)
- Defines common capabilities for managing arbitrary resources
- Management features are packaged as Management Capabilities (set of Operations/Properties/Metadata/Events) that are composable
- Topics for events related to management capabilities
- Identity capability (required):
  - ResourceId:
    - Globally-unique ID for managed resource
    - Unique for all time
    - Persistence throughout the lifetime of manageability endpoint
    - Guidelines for determining equality of endpoints

WSDM-MUWS (cont'd)

- Manageability Characteristics Capability (optional)
  - List of properties describing which management capabilities are supported
  - `<ManageabilityCapability>`
    ```
    <ManageabilityCapability>
      xsd:anyURI
    </ManageabilityCapability>*
    ```
- Correlateable Properties Capability (optional)
  - Used to determine whether two WS-Resources represent the same managed resource
  - Set of "match" expressions on resource properties (boolean, XPath 1.0, XPath 2.0)

WSDM-MUWS (cont’d)

- Configuration Capability (optional)
  - Any resource property exposing a value that, when changed, changes the operational behavior of the resource
- Description Capability (optional)
  - Provides caption, textual description and version information for a resource

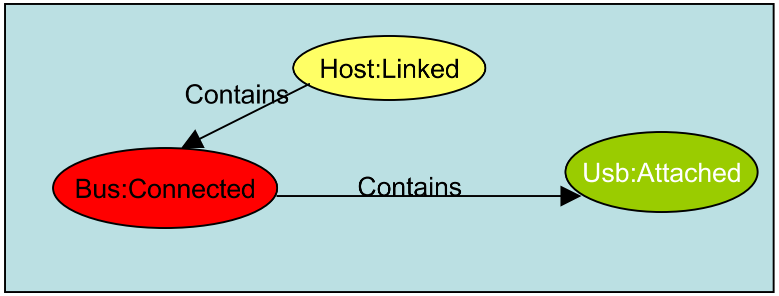
WSDM-MUWS (cont’d)

- Operational Status Capability (optional)
  - Representation of availability of a resource:
    - Available
    - PartiallyAvailable
    - Unavailable
    - Unknown
- Metrics Capability (optional)
  - Defines data types for different kinds of metrics
  - Predefined Integer and Duration types
  - CurrentTime resource property

WSDM-MUWS Relationships

- Describes association between resources; optional
- Type property conveys semantics
- Can have properties
  - Relationship Type, taxonomy can be represented:

```
<Type>
  <usb:Attached>
    <bus:Connected>
      <generally:Linked/>
    <bus:Connected>
  </usb:Attached>
</Type>
```

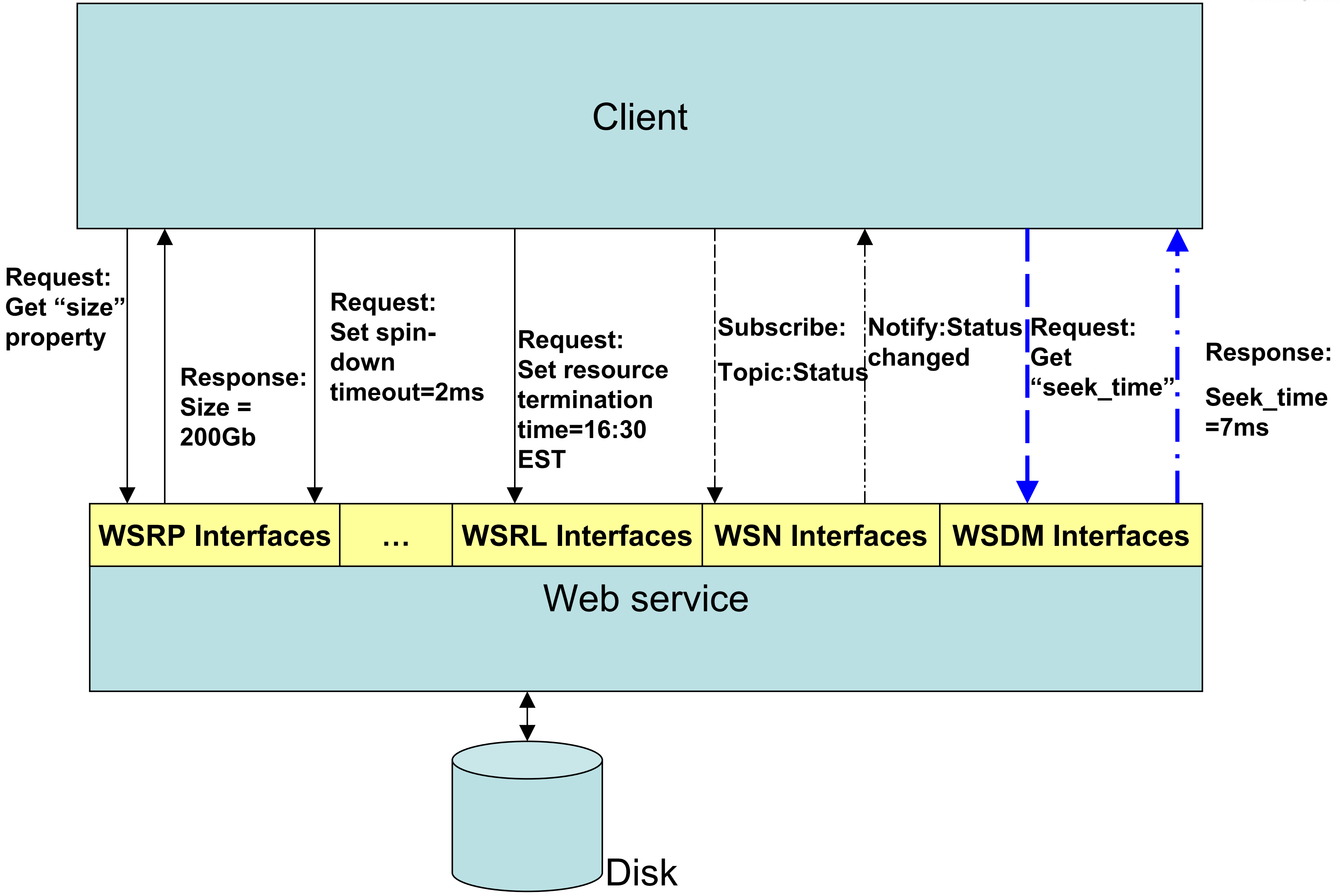


- Type is xsd:any with verbal restrictions in the spec; no specific types defined
- Exposed as array of Relationship resource properties
- Relationships' lifetimes can be managed using WS-ResourceLifetime

WSDM-MUWS Advertisement

- Defines four Notification topics, optional
  - ManageabilityEndpointCreation
  - ManageableResourceCreation:
    - Special case of ManageabilityEndpointCreation
  - ManageabilityEndpointDestruction
  - ManageableResourceDestruction:
    - Special case of ManageabilityEndpointDestruction
- Example
  - WSEE (Web Services Execution Environment) sends notification for creation of a Web service

WSDM-MUWS: an illustration



WSDM-MOWS

- Builds upon WSDM-MUWS to manage Web service endpoints:
  - An application of MUWS
- **Manageability** endpoint and **management** endpoint:
  - Both Web service endpoints
  - Could be same or different
  - Either approach is transparent to consumers
- Typical management operations:
  - Query status of Web service endpoint (up, down)
  - Number of queries processed
  - Describe relationships between endpoints
  - …

Example: Manageability integration

- Disk example with manageability
- Add MUWS properties:

```
<!-- MUWS Resource Identity properties -->
  <element ref="muws-xs:ResourceId" />
<!-- MUWS Resource Description properties -->
  <element ref="muws-xs:Name" minOccurs="0" />
  <element ref="muws-xs:Version" minOccurs="0" />
<!-- MUWS Resource State properties -->
  <element ref="muws-xs:OperationalStatus" />
```

Example: Manageability integration (cont’d)

Add MUWS operations:

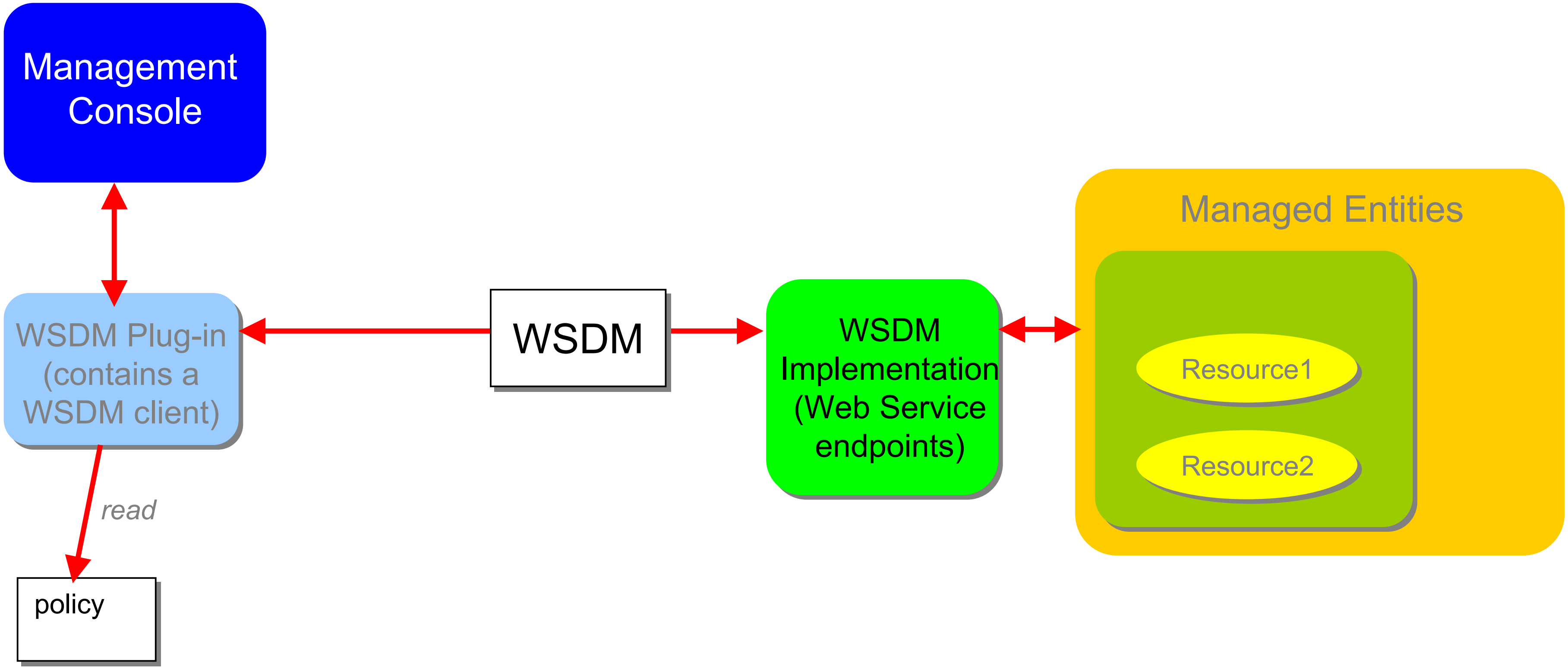
```
<!-- muws-wsdl:ResourceState operations -->

• <operation name="Start">
     <input name="StartRequest" message="muws-wsdl:StartRequest" />
     <output name="StartResponse" message="muws-wsdl:StartResponse" />
  </operation>

• <operation name="Stop">
     <input name="StopRequest" message="muws-wsdl:StopRequest" />
     <output name="StopResponse" message="muws-wsdl:StopResponse" />
  </operation> <!-- muws-wsdl:Metrics operation   -->
```

Note: Actual “start” & “stop” operations must be implemented by the developer

Management Tool Integration

- Goal: Utilize WSDM as the management channel
- Components needed:
  - Implementation of WSDM (and WSRF and WSN)
    - Expose resource manageability interfaces as WSDM-compliant Web services
    - Execute in a Web service container; receive and send SOAP messages
    - Provide base management interfaces and capabilities that can be extended by application-specific services, as needed
    - Provide bootstrapping/discovery strategies (typically, a known “root object”)
    - Lifecycle management of WSDM Web services
  - Management Tool
    - WSDM client; ability to send and receive SOAP messages
    - Support for policy regarding access control for operations
    - Understand discovery strategy adopted by WSDM Implementation

Management Tool integration


Management Console
WSDM Plug-in (contains a WSDM client)
read
policy
WSDM
WSDM Implementation (Web Service endpoints)
Managed Entities
Resource1
Resource2

Example: Management Tool Integration

- HP’s OpenView suite provides:
  - Functionality to create managed nodes on the server side and representations on the client side (console):
    - Reflects the hierarchy and dependencies of the various resources in a managed environment
  - Management channel that communicates using the WSDM protocol
  - Ability to create custom message groups for events that are generated
  - Alerts; specification of severity levels; color-coding for different severity levels
  - Generate customized reports for various aspects of the managed environment, such as performance data

Open-Source Contributions to Apache

- HP has made three open-source contributions to the Apache Software Foundation:
  - Two announced jointly with the Globus Alliance
- Contributions:
  - Implementation of WSRF (“Apollo”)
  - Implementation of WSN (“Hermes”)
  - Implementation of WSDM-MUWS (“Muse”)
- All three projects are in the “incubation” stage:
  - Committers from HP, Globus & others
  - Seeking widespread participation

Apache “Apollo” Project

- WSRF 1.2 implementation
- Project website:
  - http://incubator.apache.org/apollo
- Developers’ mailing list
  - apollo-dev@ws.apache.org

Apache “Hermes” Project

- WSN 1.2 implementation
- Project website:
  - http://incubator.apache.org/hermes
- Developers’ mailing list:
  - hermes-dev@ws.apache.org

Apache “Muse” Project

- MUWS 0.5:
  - Fully implemented
- MUWS 1.0:
  - Re-engineered to build on top of separate WSRF & WSN implementations
  - Targeted for March/April 2005
- Project website:
  - http://incubator.apache.org/muse
- Developers’ mailing list:
  - muse-dev@ws.apache.org

Tutorials, additional information

- Tutorials provided on Apollo, Hermes and Muse sites:
  - Documentation → Tutorial
  - Examples, templates of services provided
  - Trail Map walks through process of creating interfaces, generating stubs, service deployment and service invocation

References

- WSRF: http://www.oasis-open.org/apps/org/workgroup/wsrf/
- WSN: http://www.oasis-open.org/apps/org/workgroup/wsn/
- WSDM: http://www.oasis-open.org/apps/org/workgroup/wsdm/
- OGSI: http://forge.gridforum.org/projects/ogsi-wg
- GGF: http://www.ggf.org
- HP technologies: http://www.hp.com/go/technologies

Q&A

Questions?