



## Overview of WS-Management, CIM-XML, WS-CIM

Rick Landau  
System Management Architecture & Standards  
2006/04/04 v0.4



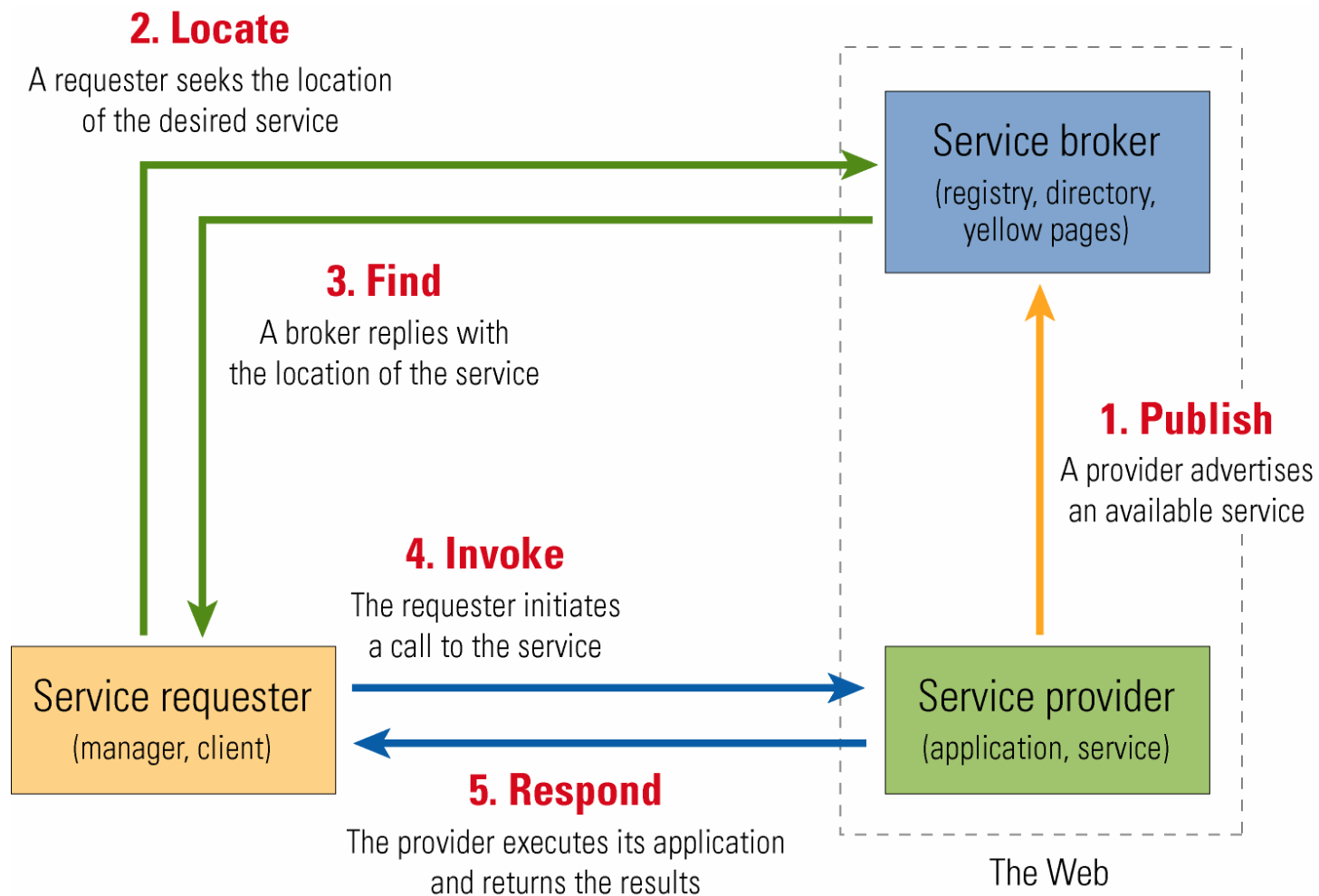
## Three Things Today

- WS-Management web service (WS) protocol
- CIM-XML WS protocol
- WS-CIM DMTF working group, specs

# What is a Web Service

- Many early-generation examples in the market
- Realistic definition
  - Is addressed at some URI
  - Uses XML to represent data
  - Includes an XML schema to describe a valid data structure
  - Uses SOAP to structure messages
  - Uses WSDL to describe interfaces, if necessary

# Interaction with a Web Service

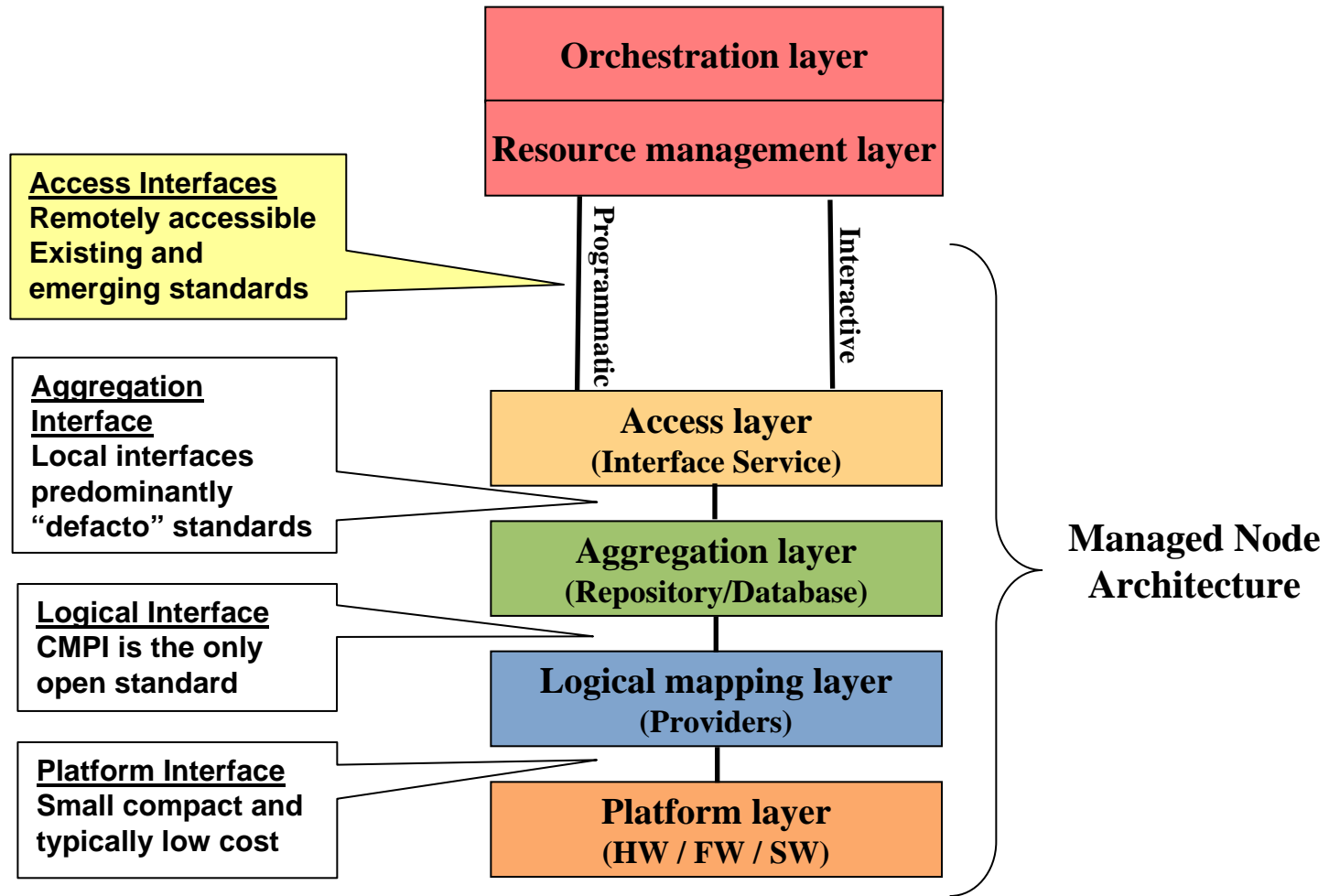


## Why Care

- Easier to implement
  - Plain text, XML
- More secure
  - Plain text header for filtering, firewalls
  - Opaque contents may be encrypted
  - Security protocols may be layered onto base
- Mix and match protocol components to meet needs
  - Pieced together from "composable" protocols
    - Optional add-in features, separate specs, orthogonal
  - If you need more function, just add headers and messages
  - E.g., atomic transactions

Orchestration layer
Resource management layer
Access layer
Aggregation layer
Logical mapping layer
Platform layer

# Where It Fits in Dell Unified Manageability Architecture



# WS-Management & CIM-XML Protocols

- Both access protocols to repository of management data
- WS-Management uses new web service protocol structure
- CIM-XML uses older protocols (where "old" = 3-5 years)

# WS-Management

- WS protocol (suite)
- Originally from Microsoft, Intel, CA, BMC, Dell, et al.
- Being worked on in DMTF WG, in a subgroup of WIP
  - WIP = WBEM Infrastructure & Protocols WG of DMTF
- Not CIM-based per se
  - Could use any data model
  - Everyone wants to use CIM
  - WS-CIM WG is creating the CIM schema to be used by WS-Man and other WS protocols
- Specs: go to [microsoft.com](http://microsoft.com), search for "ws-management"



# WS Composed Protocols

- WS-Addressing (endpoint references)
- SOAP (envelope=header+body structure)
- WS-Transfer (get, put)
- WS-Enumeration (enumerate, pull)
- WS-Eventing (subscribe, unsubscribe)
- WS-Security (authenticate)
- Etc.

# WS-Man Operations

- Protocol operation specifies request and response
- Target: resource instance
  - Create, Delete
  - Get, Put (properties of an instance)
- Target: collection of instances
  - Enumerate (get handle), Release (handle)
  - Pull (instances)
- Target: event streams
  - Subscribe, GetStatus, Renew, Unsubscribe
  - Pull (collections of events)
    - Normal mode is push from service to client
  - Ack
  - Etc.

# WS-Man Example Message

```
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wsman="http://schemas.xmlsoap.org/ws/2005/02/management">
  <env:Header>
    <wsa:To>http://<IP address of server>:<port>/wsman?ResourceURI=(http://schemas.dmtf.org/wsman/2005/02/cimv2.9/CIM\_ComputerSystem)
    </wsa:To>
    <wsa:ReplyTo>
      <wsa:Address env:mustUnderstand="true">
        http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous
      </wsa:Address>
    </wsa:ReplyTo>
    <wsa:Action env:mustUnderstand="true">
      http://schemas.xmlsoap.org/ws/2004/09/transfer/Get
    </wsa:Action>
    <wsman:MaxEnvelopeSize env:mustUnderstand="true">
      51200
    </wsman:MaxEnvelopeSize>
    <wsa:MessageID>uuid:20CBF75C-008E-4C68-AA14-F3EB9D94E617</wsa:MessageID>
    <wsman:SelectorSet>
      <wsman:Selector Name="CreationClassName">
        ComputerSystem
      </wsman:Selector>
      <wsman:Selector Name="Name">
        IPMI%20Controller%2032
      </wsman:Selector>
    </wsman:SelectorSet>
    <wsman:OperationTimeout>PT60.000S</wsman:OperationTimeout>
  </env:Header>
  <env:Body>
  </env:Body>
</env:Envelope>
```

# WS-Man Catalog Contents

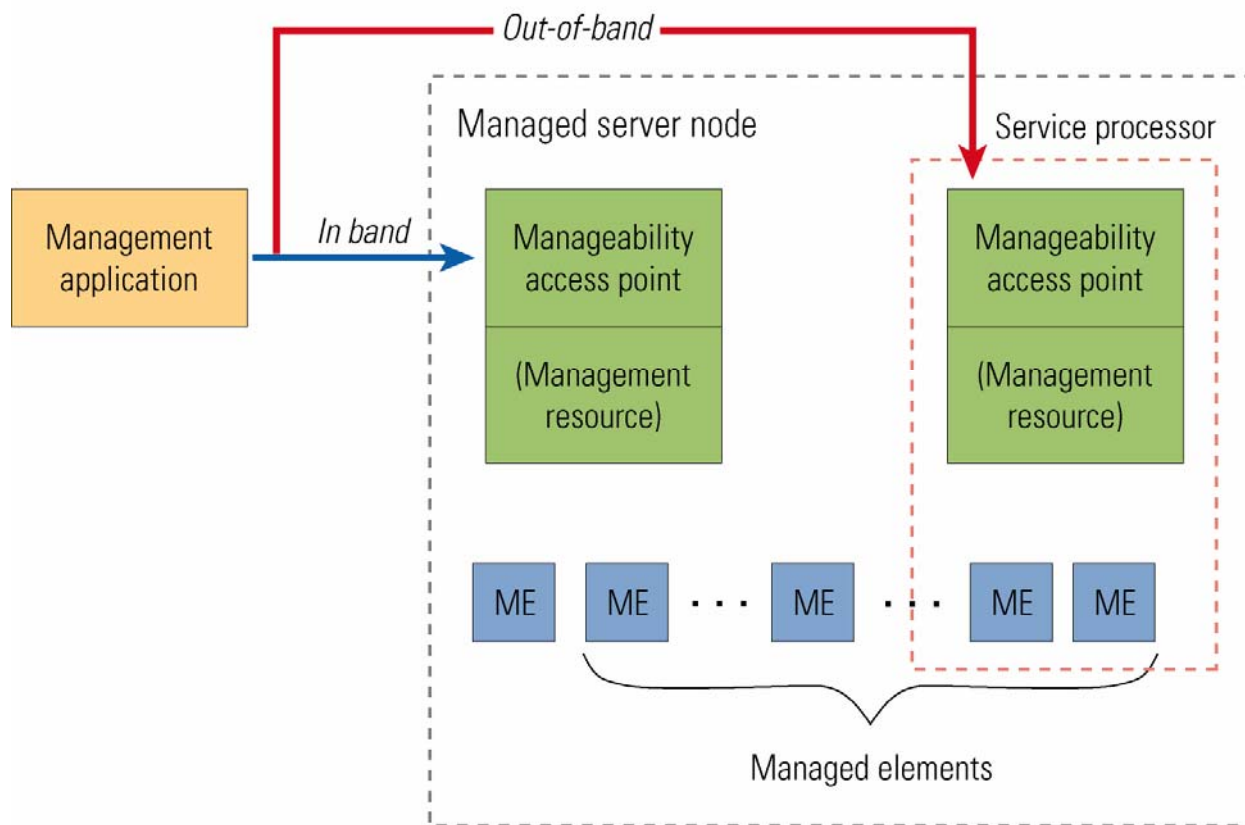
- Data only, accessed thru WS-Man agent
- Catalog lists resource class names
  - E.g., disk, memory, printer
  - Use these names to talk to agents about resource instances
- Catalog lists attributes of each resource class
  - ResourceURI
  - DisplayName, Keywords, Notes, Vendor
  - Relationships
  - Compliance (spec version)
  - Action(s), Operation(s)
  - SelectorSetRef(s), OptionSetRef(s), SchemaRef
  - FilterDialect for Enumerate & Subscribe, DeliveryMode

# Process of Discovery

- Find a node (S.E.P.)
- Find WS-Man service(s) on that node (S.E.P.)
  - Sticky point recently, general problem
    - One system may have multiple agents, OR
    - Multiple nodes with agents need to be correlated
- Read catalog to get resource classes (ResourceURIs)
- Enumerate resource instances of resource classes (Selectors)
- Get or Put to a resource instance

# Discovery of Multiple MAPs

## Managed node with service processor (two MAPs)



## Where WS-Man is Implemented

- Windows
- Linux (need an open source service)
- Sun Solaris

- Access protocol for CIMOM data
  - Protocol-independent operations and data representation, plus HTTP binding
  - Implemented
    - Windows WMI
    - OpenWBEM on Suse
  - Docs
    - DSP200 on dmtf.org, "CIM Operations over HTTP"
    - DSP201, "What is xmlCIM?"
- Function style
  - Operation and arguments
- To be replaced by WS protocols



# WBEM Operations

- Basic read and write
  - GetClass, EnumerateClasses, EnumerateClassNames
  - GetInstance, EnumerateInstances EnumerateInstanceNames
  - GetProperty, SetProperty
- Instance create and delete
  - CreateInstance, ModifyInstance, DeleteInstance
- Metadata
  - CreateClass, ModifyClass, DeleteClass
  - GetQualifier, SetQualifier, DeleteQualifier, EnumerateQualifiers
- Navigation
  - Associators, AssociatorNames
  - References, ReferenceNames
- Invoke method
- Etc.

# CIM-XML Example

```
M-POST /cimom HTTP/1.0
Content-Type: text/xml;charset=UTF-8
Accept: text/xml, application/xml
Man: http://www.dmtf.org/cim/mapping/http/v1.0;ns=48
48-CIMProtocolVersion: 1.0
48-CIMOperation: MethodCall
48-CIMMethod: GetClass
48-CIMObject: root%2Fcimv2
User-Agent: Javal.2.1
Host: edoc5-pc
Content-length: 445
<?xml version="1.0" encoding="UTF-8"?>
  <CIM DTDVERSION="2.0" CIMVERSION="2.0">
    <MESSAGE ID="2000:1:24:11:0:44:58:1" PROTOCOLVERSION="1.0">
      <SIMPLEREQ>
        <IMETHODCALL NAME="GetClass">
          <LOCALNAMESPACEPATH>
            <NAMESPACE NAME="root" />
            <NAMESPACE NAME="cimv2" />
          </LOCALNAMESPACEPATH>
          <IPARAMVALUE NAME="ClassName">
            <CLASSNAME NAME="cim_logicalport" />
          </IPARAMVALUE>
        </IMETHODCALL>
      </SIMPLEREQ>
    </MESSAGE>
  </CIM>
```

- Confusion in name
  - "WS-CIM" used to be thought of as a single combined protocol
    - Produced only requirements for two other protocols
  - Name WS-CIM reused for new data mapping effort
- Establishing XML schema for WS-Man and WSDM to use
  - Mapping datatypes and other CIM attributes to XML
  - Rules for algorithmic translation
- Example not available yet

Questions?