



# **WBEM Services Specification JSR-0048**

**Jim Davis**  
Senior Staff Engineer  
Sun Microsystems, Inc.

# Overall Presentation Goals

To educate you about the use of Java™ Technology and Web Based Enterprise Management (WBEM) for system and network management



# Learning Objectives

As a result of this presentation, you will be able to:

- Define WBEM and when to use it

- Distinguish between WBEM and CIM

- Understand the basics of the Java™ WBEM APIs

- Understand the basics of the CIM Schema



# Speaker's Qualifications

## Jim Davis

Architect, Solaris Management

Sun Representative to the DMTF TC  
and DMTF Interoperability WG chair

Specification Lead for JSR-0048

## Guru Bhat

Sun WBEM Development Team and  
Sun Schema Review Team chair

## Andrea Westerinen

Senior Manager and Architect, Information Modeling

Cisco representative to the DMTF TC

Co-author of multiple IETF Internet-Drafts  
on policy modeling



# Agenda

WBEM

WBEM Services Specification (JSR-0048)

WBEM Services API

WBEM Services Examples

Modeling Information in CIM



# Benefits

Common model and schema for describing managed elements

Interoperability through standards based protocols (HTTP, XML) and schema (CIM)

Java™ technology-based representation of CIM objects

Standard API for the Java platform, for developing instrumentation and client applications



# What Is WBEM?

## Web-Based Enterprise Management

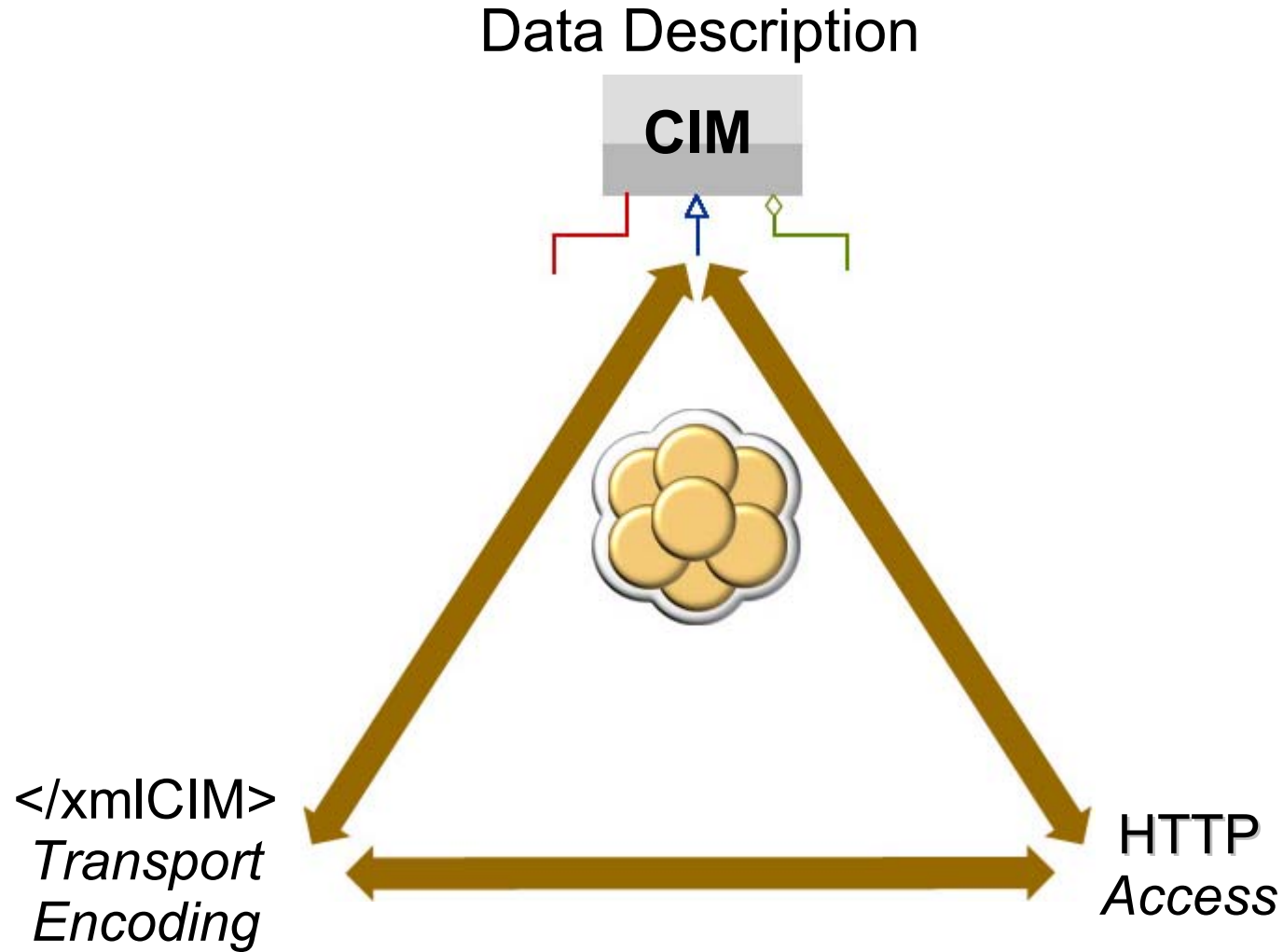
A set of management and internet standard technologies developed to unify the management of enterprise computing environments

The goal is to provide customers with the ability to manage all systems regardless of their instrumentation type, using a common standard

A DMTF Standard



# WBEM Environment



# Common Information Model (CIM)

## Common Information Model

Defines the Schema used to represent real-world objects being managed

Object oriented paradigm

## CIM Specification

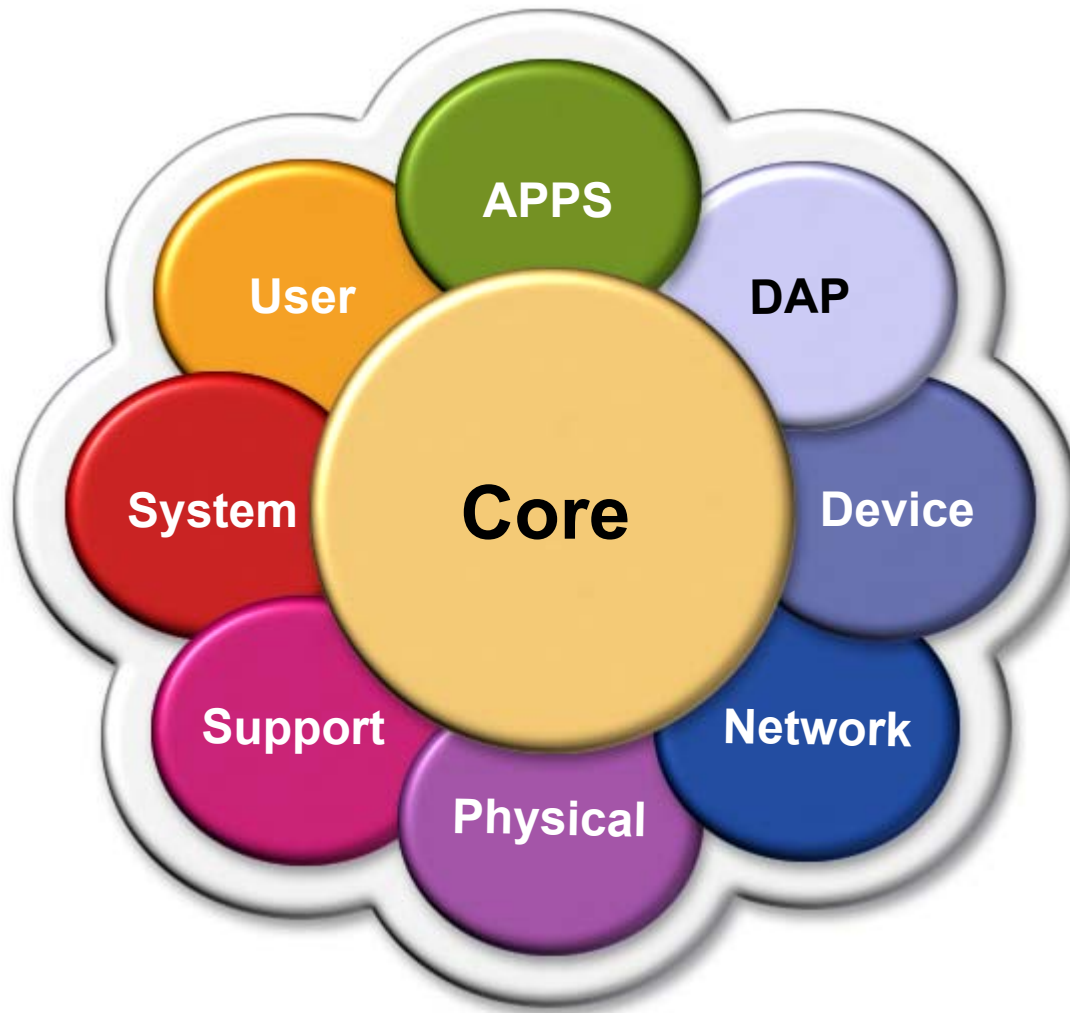
Meta model, high level concepts, and definition language (MOF)

## CIM Schema

Core and Common Model



# CIM Core and Common Model



# Representation of CIM in XML

## DTD for CIM

- Declarations for the CIM metaschema

- Messages for encapsulation over HTTP

CIM classes and instances are valid XML documents

## Advantages

- Only requires one DTD for CIM

- Straight forward



# Representation of CIM in XML Example

```
<CLASS NAME="CIM_LogicalPort" SUPERCLASS="CIM_LogicalDevice">
  <QUALIFIER TRANSLATABLE="true" NAME="Description" TYPE="string">
    <VALUE>The abstraction of a port or connection point of a Device.
    This object should be instantiated when the Port has independent
    management characteristics from the Device that includes it.
    Examples are a Fibre Channel Port and a USB Port. This class would
    not be instantiated for an Ethernet Port which is not managed
    independently of the EthernetAdapter.</VALUE>
  </QUALIFIER>
  <PROPERTY NAME="Speed" TYPE="uint64">
    <QUALIFIER TRANSLATABLE="true" NAME="Description" TYPE="string">
      <VALUE>The speed of the Port in Bits per Second.</VALUE>
    </QUALIFIER>
    <QUALIFIER TRANSLATABLE="true" NAME="Units" TYPE="string">
      <VALUE>Bits per Second</VALUE>
    </QUALIFIER>
  </PROPERTY>
  <PROPERTY NAME="MaxSpeed" TYPE="uint64">
    <QUALIFIER TRANSLATABLE="true" NAME="Description" TYPE="string">
      <VALUE>The max speed of the Port in Bits per Second.</VALUE>
    </QUALIFIER>
    <QUALIFIER TRANSLATABLE="true" NAME="Units" TYPE="string">
      <VALUE>Bits per Second</VALUE>
    </QUALIFIER>
  </PROPERTY>
</CLASS>
```



# CIM Operations Over HTTP

Supports both HTTP 1.0 and HTTP 1.1

Essential information is exposed in HTTP headers for efficient firewall/proxy handling

Each CIM operation is described completely in XML

Supports simple and multiple method calls Operations

- Data

- Meta data

- Queries

- Methods



# CIM Operations over HTTP 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: Java1.2.1
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>
```



# DMTF

Distributed Management Task Force

Over 200 member companies

Led by 14 Board member companies

3Com, Cisco, Compaq, Dell, HP, IBM/Tivoli,  
Intel, Lucent, Microsoft, NEC, Novell, SCO,  
Sun Microsystems, Symantec

Standards developed since 1992

Desktop Management Interface (DMI)

Service Incident Exchange Standard (SIS)  
and Service Exchange Standard (SES)

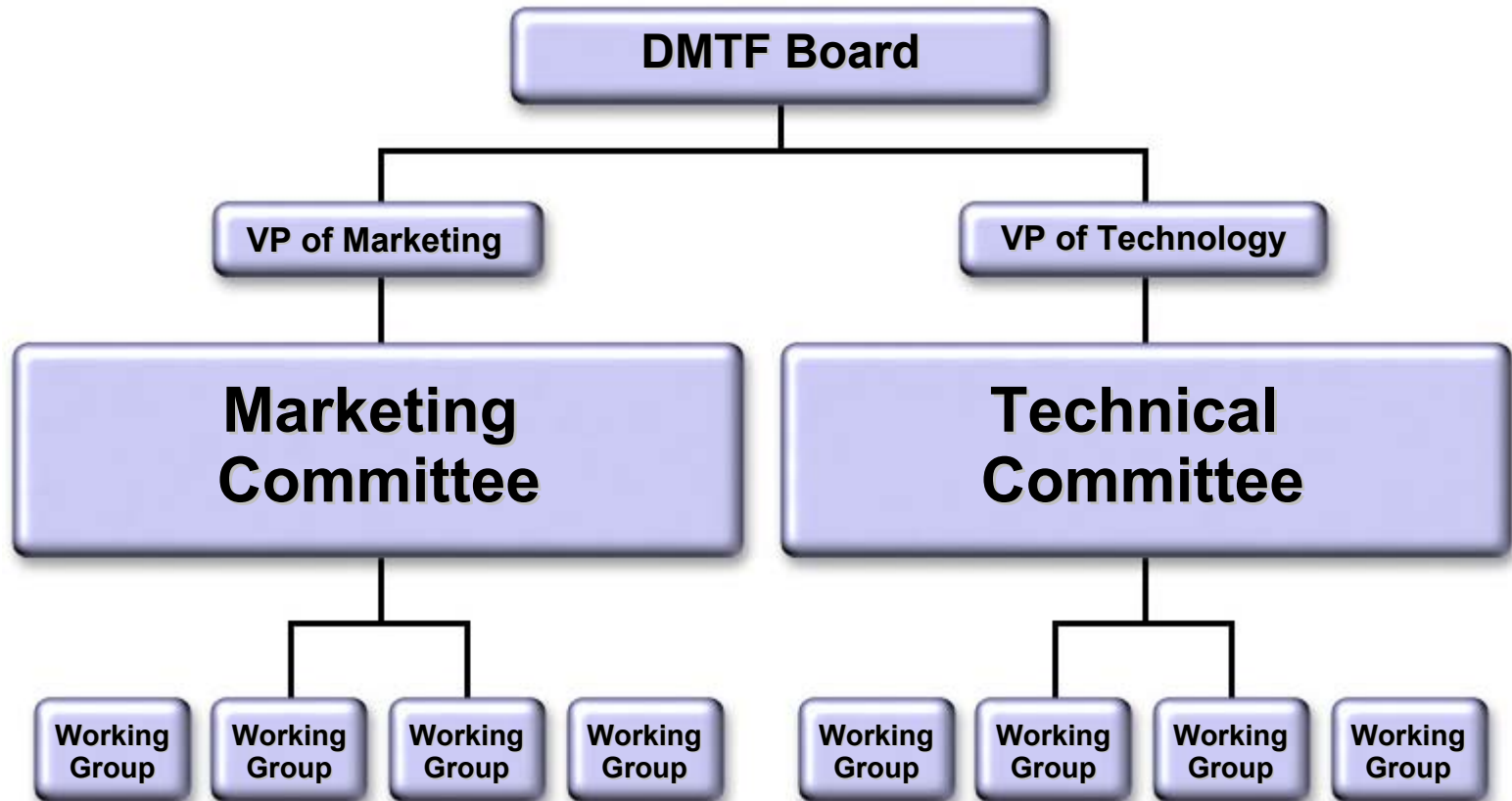
Common Information Model (CIM)

Web-Based Enterprise Management (WBEM)

Directory Enabled Networks (DEN)



# DMTF Organization



# WBEM Services JSR Objectives

Provide complete support for WBEM on the Java™ platform

To have a Java VM-based platform be an effective environment for developing WBEM platform-independent applications and instrumentation

Create a set of Java technology-based APIs for WBEM

WBEM enable any virtual machine for the Java platform (Java virtual machine)

Conformance Test Suite (CTS)



# JSR Member Companies

Sun Microsystems

IBM

Nokia

Compaq

Veritas

SNIA

Caldera

Computer Associates

OpenMaster

Hewlett Packard

Cisco Systems

Lucent Technologies



# JSR Status

|                             |       |
|-----------------------------|-------|
| JSR Submitted               | 12/99 |
| JSR Accepted                | 01/00 |
| CAFE Closed                 | 03/00 |
| Participant Review (target) | 07/01 |
| Public Review (target)      | 09/01 |



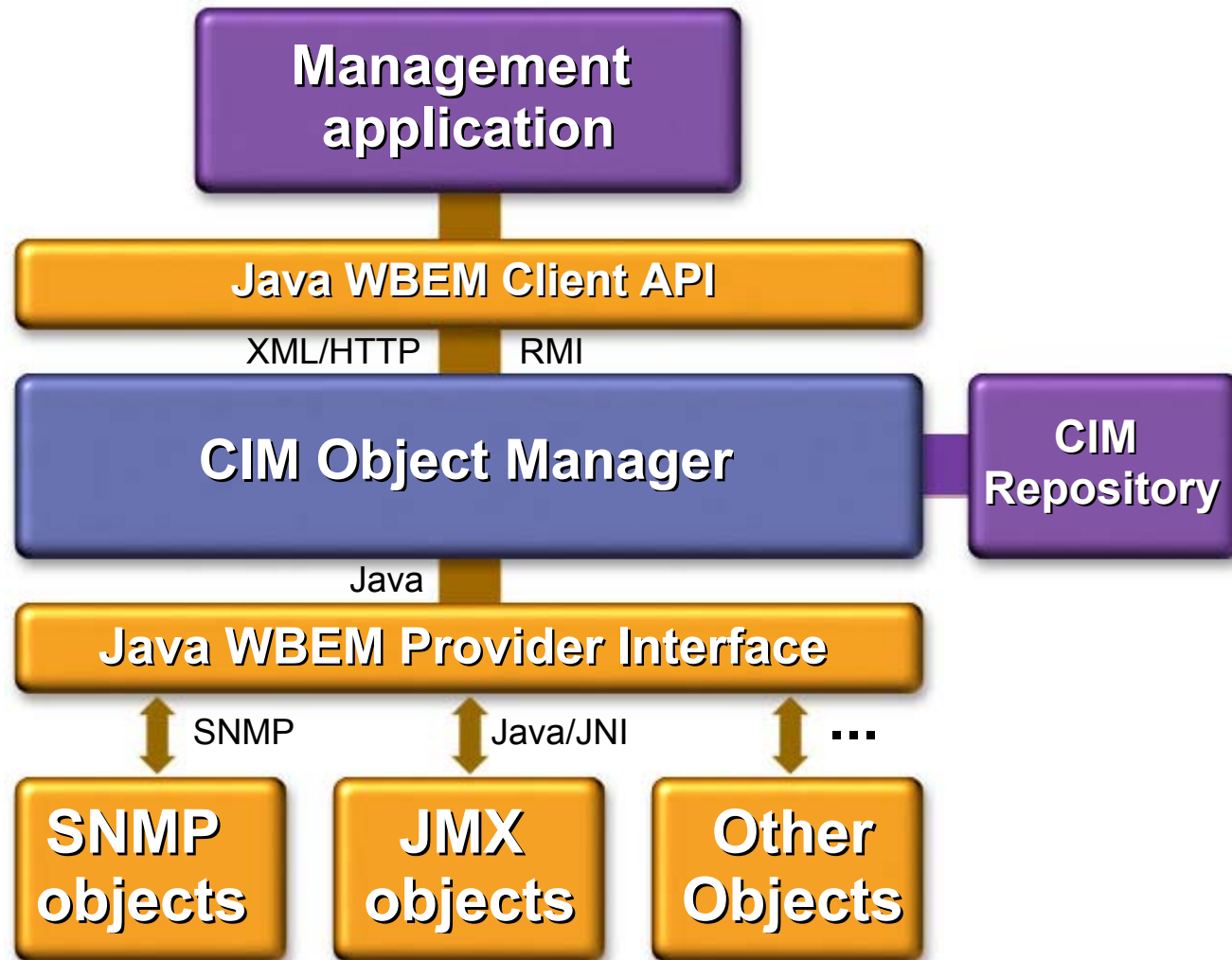


# WBEM Services API And Examples

**Guru Bhat**

Senior Software Engineer  
Sun Microsystems, Inc.

# Basic WBEM Architecture



# Java™ WBEM API

## CIM API

“Java mapping” of the CIM model

## Client API

Supports RMI and XML/HTTP

XML/HTTP defined by CIM spec

## Provider API

Enables data retrieval from user defined modules



# CIM API

The CIM API represents CIM  
in Java™ technology:

CIMClass

CIMInstance

CIMProperty

CIMValue

CIMObjectPath

CIMNameSpace

CIMDataType

...



# Client API Operations

Instance, Class, QualifierType

Create, delete, get, enumerate, set

Event Subscription & Notification

Property (get, set)

NameSpace

Create, delete, enumerate

Method invocation

Execute Query

Association traversal

Associators, associatorNames

References, referenceNames



# Provider API

Instance provider

Supply instances of a given class

Property provider

Supply property values

Method provider

Implements method invocation for classes

Associator provider

Implements association traversal

Event Provider

Handles Event Subscription/Notification



# Developing Client Applications

Steps involved in writing clients

Connect to the CIMOM

Manipulate CIM information locally

Retrieve/set remote information



# Client Example

## Enumerate instances of process information

```
CIMNameSpace cns = new CIMNameSpace(hostName);
CIMClient cc = new CIMClient(cns, principal, credential);

CIMObjectPath op = new CIMObjectPath("CIM_Process");
Enumeration e = cc.enumerateInstances(op);
while(e.hasMoreElements()) {
    System.out.println((CIMInstance)e.nextElement());
}
```



# Developing Providers

Determine properties, methods

Define the MOF classes

Implement provider interface(s)

“Publish” the classes (compile)



# Provider Example

```
public class MyProvider implements InstanceProvider,
                                   MethodProvider {

    public void initialize(CIMOMHandle cimom) {}

    public void cleanup() {}

    public void getInstance(CIMObjectPath op,
                           CIMClass cc,
                           boolean localOnly)
        throws CIMException {
        CIMInstance = cc.newInstance();
        ci.setProperty("name", value);
        ...
        return ci
    }
}
```





# Modeling Information In CIM

**Andrea Westerinen**  
Architect, Intelligent Network Services  
Cisco Systems

# CIM Model

Core

System

Device

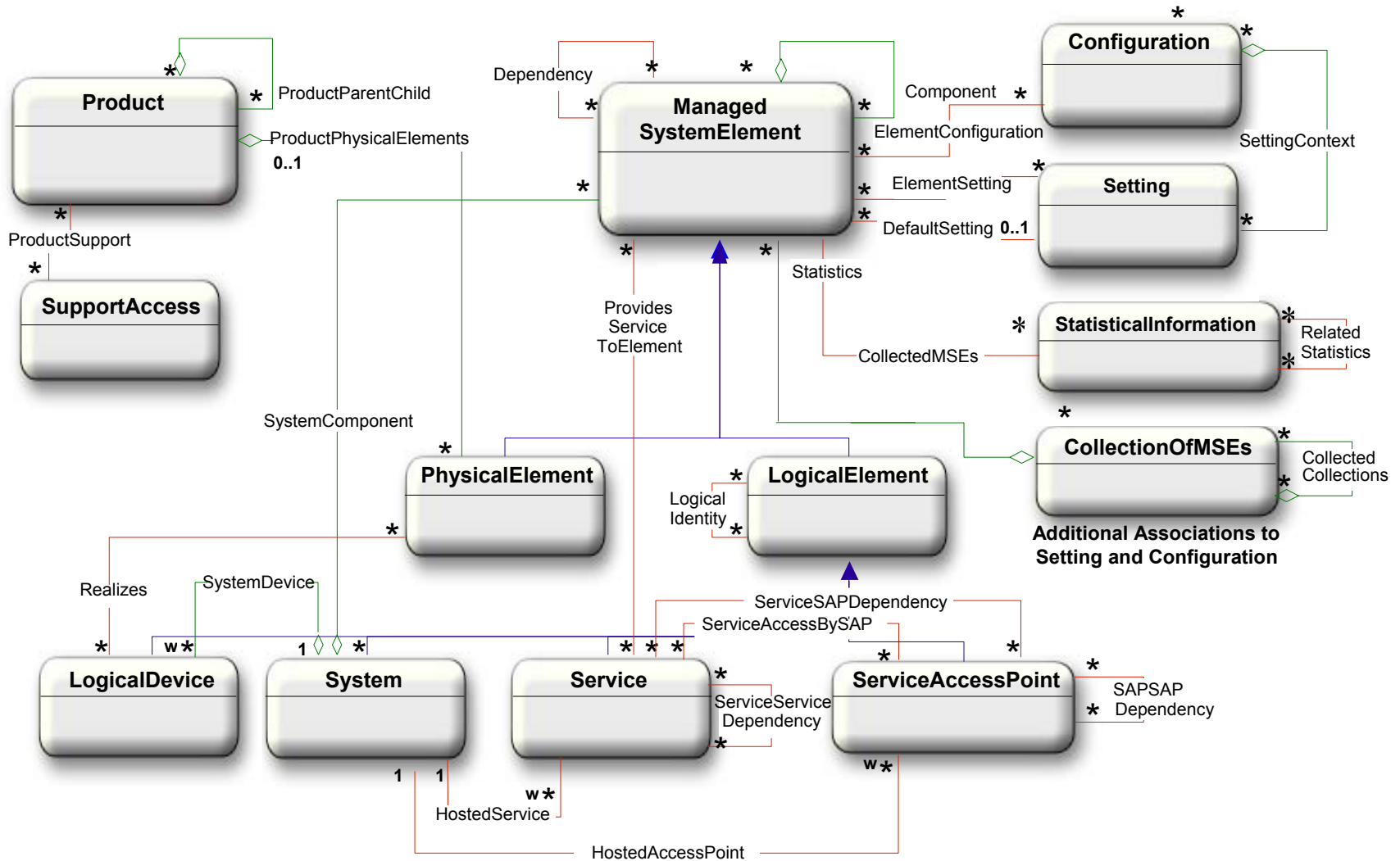
Application

Physical

Network



# Core Model



# Core Model Objects

Managed System Element

Logical/Physical Elements

Logical Device

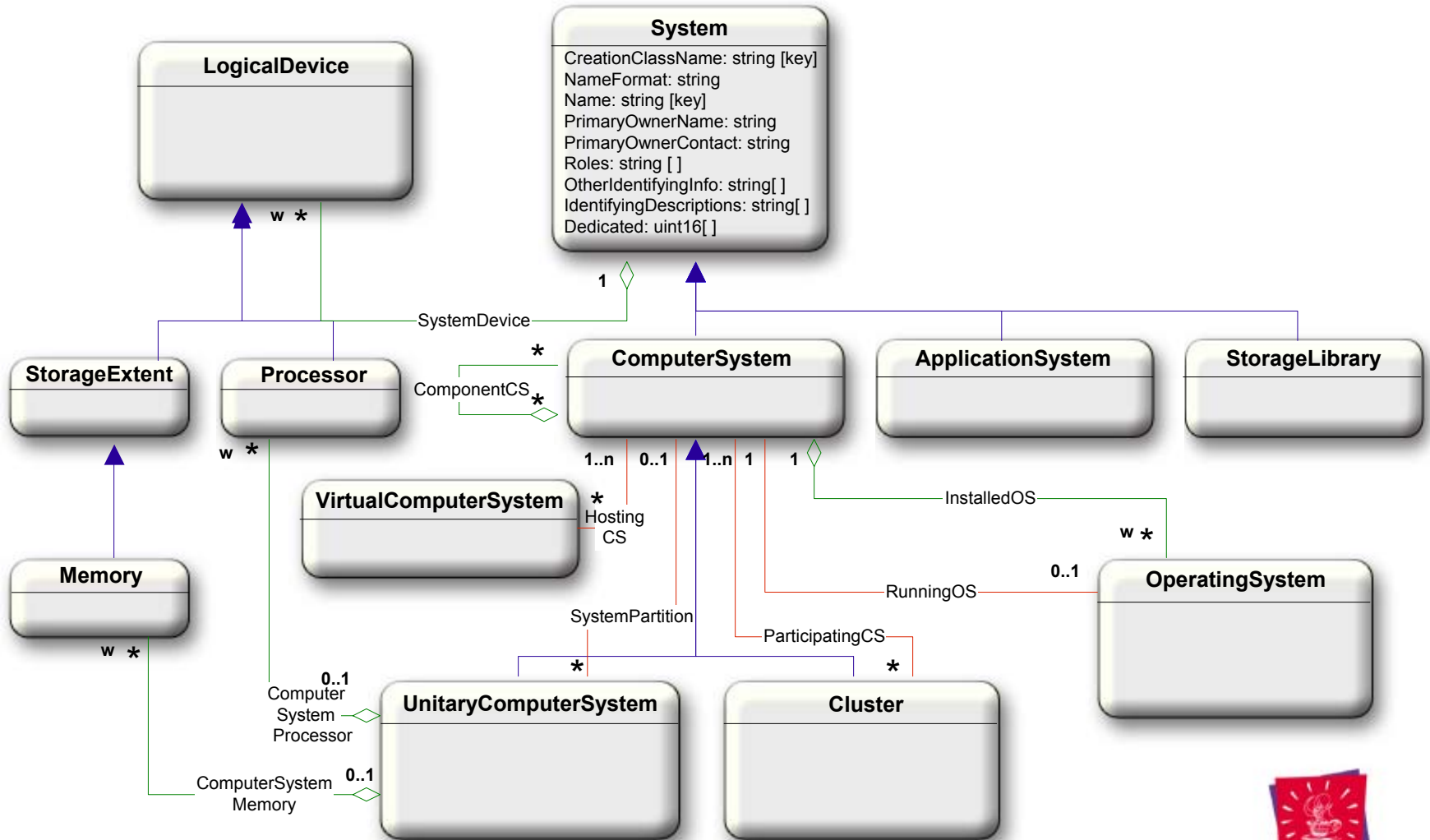
System and ComputerSystem

Service and ServiceAccessPoint

Settings and Configurations



# System Model



# System Model Objects

UnitaryComputerSystem and  
VirtualComputerSystem

Cluster

ComponentCS and SystemPartition associations

BootServices

OperatingSystem, Process and Thread

FileSystem (Local and Remote)

Job and JobDestination

BIOS and the SystemBIOS association



# Physical Model Objects

## Goals of the Model

- Inventory and asset management

- Location of an object for service replacement or upgrade

## Inherits from PhysicalElement

## Some Physical Objects

- PhysicalPackage—“Contains” or hosts other components

- ReplacementSet

- Location

- PhysicalLink and PhysicalConnector



# Device Model Objects

“Realized” in hardware (Physical Elements)

Typically have “Connection” to other Devices

Associations indicate Devices’ “controlled-controller” relationships

Redundancy Groups

System Resources

Many subclasses and much detail

Storage, NetworkAdapters and Modems, Printers and the queuing of PrintJobs, and PCI/USB Devices, Sensors, WatchDog timer, AGPController Devices, power domains can be described using the SuppliesPower association, and much more

# Application Model

Manages the Installation of Software

Software Product

Software Feature

Collection of Software Elements that performs a particular function or role in a software Product

Software Element

Individually managed part of a Software Feature

Checks

Actions

# Network Model Objects

## AdminDomain

Used to group together various network resources that must be administered the same way

## NetworkService

An abstract base class

The root of the network service hierarchy

Represents generic functions available from the network that configure and/or modify the traffic being sent

## Protocol End Point

## Logical Networks

# Summary

Use CIM to model information.

Publish your instrumentation information using WBEM by developing providers using the Java WBEM Provider API

Use Java™ WBEM Client API to manage instrumented resources



# For More Information

WBEM Services Specification for the  
Java™ Platform

<http://java.sun.com>

WBEM Services Specification for the  
Solaris™ Platform

<http://sun.com/solaris/webm>

Distributed Management Task Force

<http://dmtf.org/>





**JavaOne**<sup>SM</sup>  
Sun's 2001 Worldwide Java Developer Conference™

**Q&A**



**JavaOne**<sup>SM</sup>

Sun's 2001 Worldwide Java Developer Conference\*