WBEM Overview

Denise Eckstein
Hewlett-Packard

Module Content

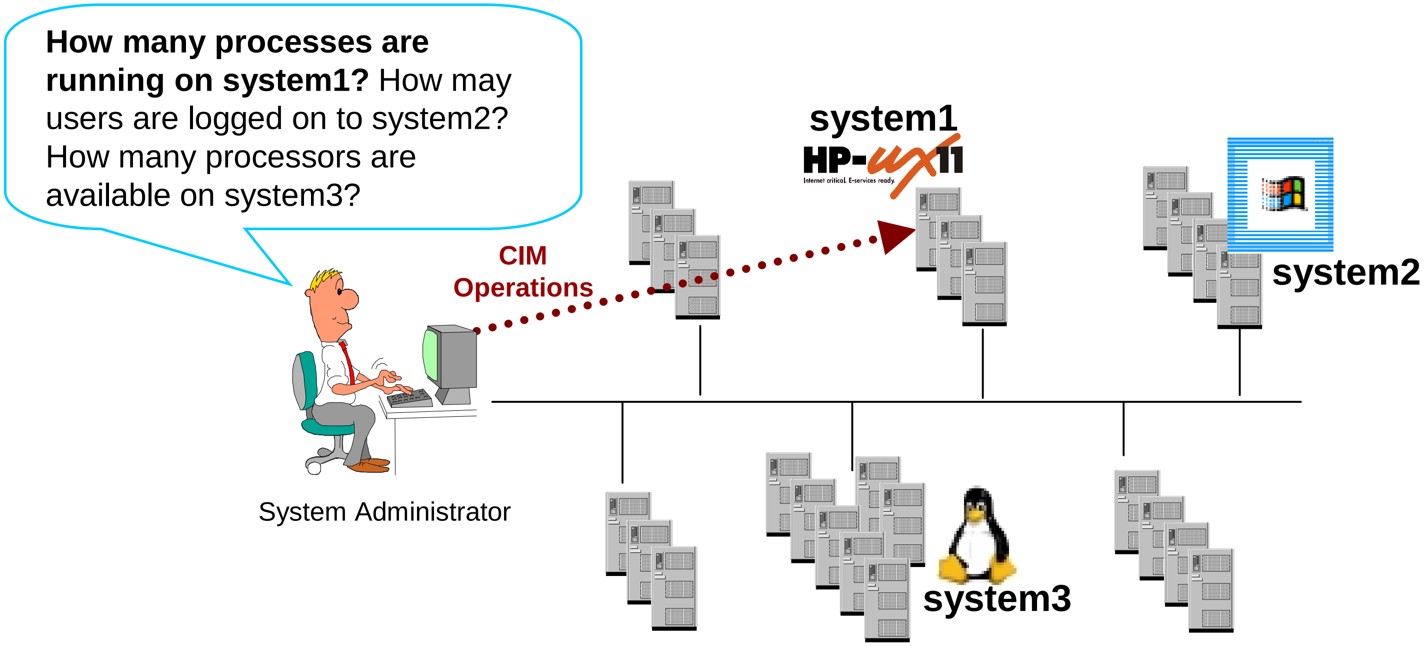
WBEM Overview

- **Problem Statement**
- Terminology
- CIM Data Model
- CIM Operations
- CIM-XML Communication Protocol

WBEM Overview

Web-Based Enterprise Management

Problem Statement


How many processes are running on system1? How may users are logged on to system2? How many processors are available on system3?
CIM Operations
system1
HP-UX 11
Internet critical. E-services ready.
system2
system3
System Administrator

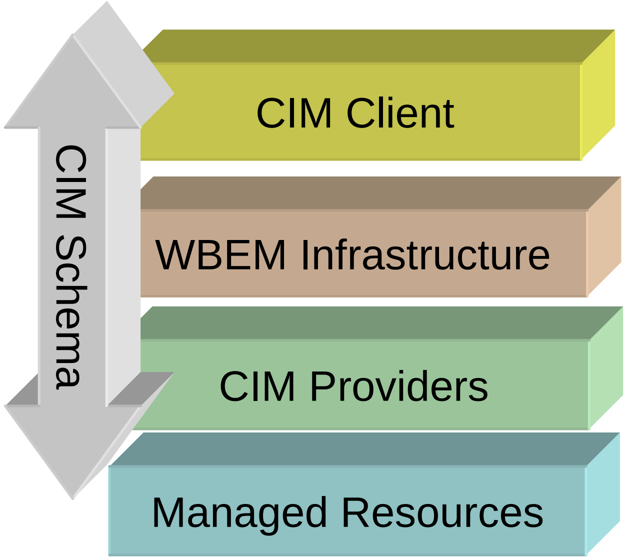
Critical Requirements

- Interoperable, Portable Management Solutions
  - Multi-Vendor and Multi-Platform Management Solutions
- "Solution" Management versus "Resource" Management
  - Management instrumentation as "pluggable" components of an integrated, customizable management solution.
  - Ever broadening definition of "resource" (e.g., system, printer, application, cluster, cell phone , etc .)
- Operational Efficiency
  - Resource Instrumentation Vendor
  - Management Solution Vendor
  - Resource Administrator
- Secure Management of "Everything" from "Anywhere"
- Faster Time To Market (TTM)
- Enhanced Opportunity for "Value–Add"

WBEM

WBEM simplifies adoption by …

- Advocating evolution not revolution
- Relying on existing, pervasive standards and technologies



WBEM challenges …

- Implementation of infrastructure imposes a high-cost of entry
- Standardizing data definitions involves significant community activity and agreement
- Delivering critical mass of applications and providers to move beyond the early adopters phase

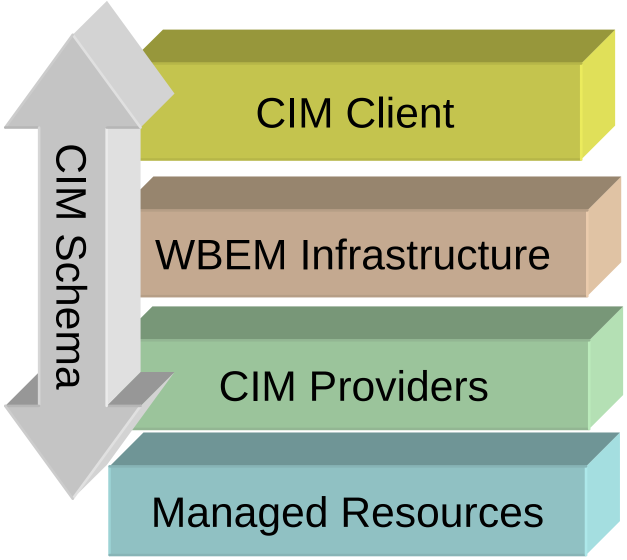
Benefits

Application Developers benefits:

- ❑ Common definition of data
- ❑ Standard mechanism to access management data
- ❑ Degree of isolation from hardware and software changes
- ❑ Reduced cost of developing multi-vendor, multi-platform, secure solutions

Instrumentation Developers benefits:

- ❑ Common definition of data
- ❑ Potential for low-cost, tight integration with a wealth of applications and systems
- ❑ Reduced cost of developing multi-vendor, multi-platform, secure solutions



Module Content

WBEM Overview

- Problem Statement
- **Terminology**
- CIM Data Model
- CIM Operations
- CIM-XML Communication Protocol

WBEM Overview

Web-Based Enterprise Management (WBEM) is a platform and resource independent DMTF standard that defines both a **common model (i.e., description) and protocol (i.e., interface)** for monitoring and controlling resources from **diverse sources** (e.g., different types of platforms or different types of resources on the same platform).

WBEM Defines

- Common Data Model
- Common Protocol

WBEM Provides

- Common view of data across platform and among resource on the same platform
- Framework that enables a "building block" approach to solution design and implementation

Web-Based Enterprise Mgmt

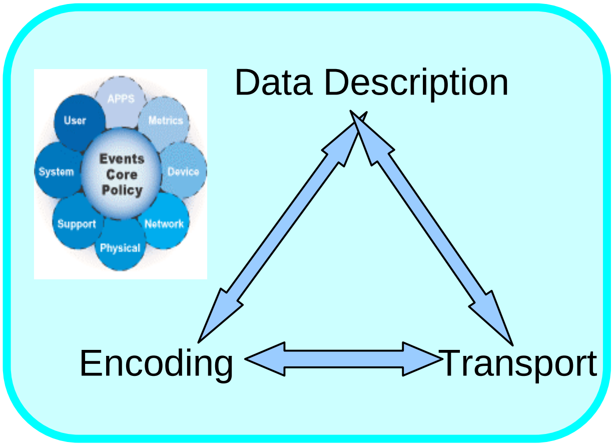
WBEM is a DMTF Standard that is defined by a set of standards that include:

WHAT

- **A Data Description Standard** that describes the resources to be managed.

HOW

- **A Communication Protocol Standard** that defines an encoding and a transport protocol.



DMTF Board Members: 3Com, Cisco, Dell Computer Corp., Hewlett-Packard Company, IBM/Tivoli Systems, Inc., Intel Corporation, Microsoft Corporation, NEC Corporation, Novell, Oracle, Sun Microsystems, Inc., Symantec Corporation, VERITAS Software

DMTF Specifications and Schema are available at: http://www.dmtf.org

Data Description

WHAT

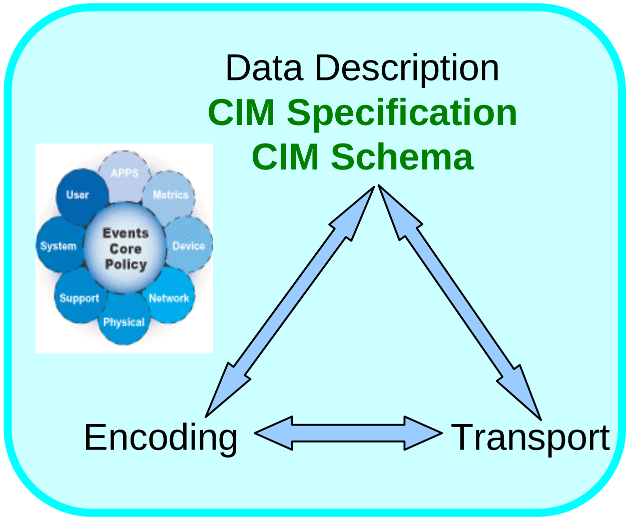
The **Common Information Model (CIM)** is the DMTF WBEM Standard for describing data. It includes:

RULES FOR DESCRIBING THE DATA

- the description of a meta-language for describing data (**CIM Specification**) and

DESCRIPTION OF DATA

- a description of the resources to be managed (DMTF **CIM Schema** + vendor extensions).



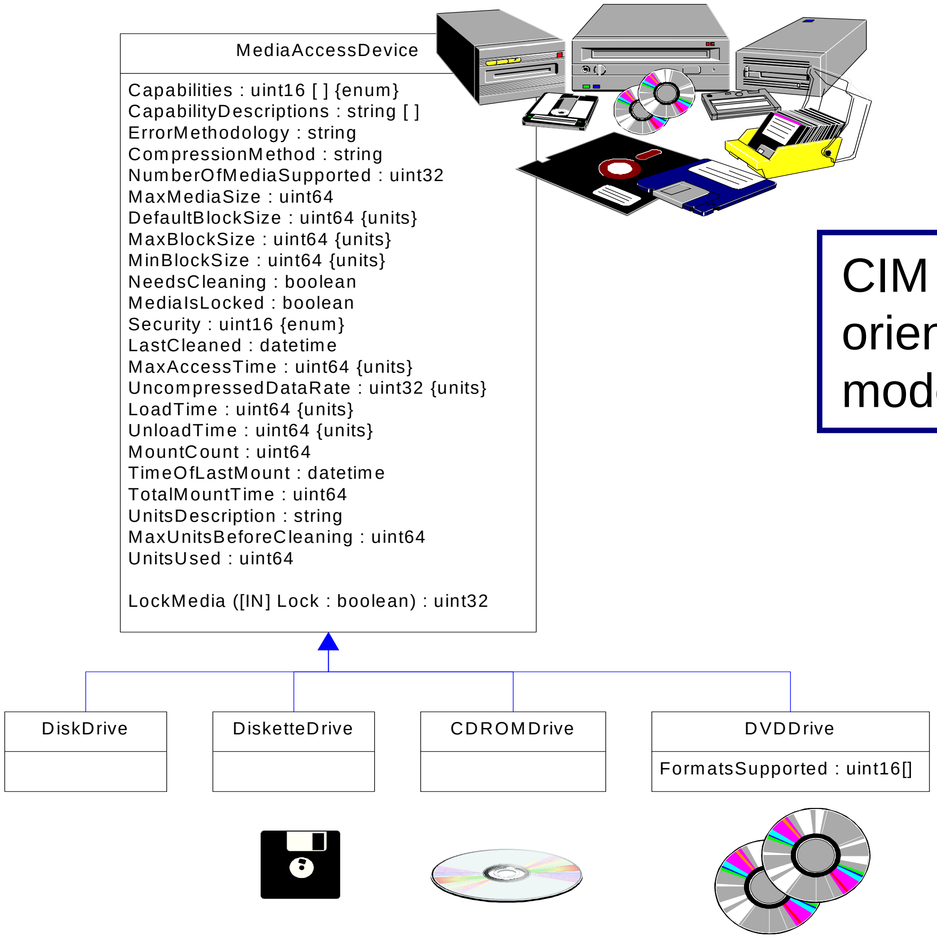
Data Description
CIM Specification
CIM Schema

Encoding

Transport

CIM Specification: Defines a formal language for describing data.
CIM Schema: Contains a description of the resources to be managed.

Common Information Model



MediaAccessDevice

Capabilities : uint16 [ ] {enum}
CapabilityDescriptions : string [ ]
ErrorMethodology : string
CompressionMethod : string
NumberOfMediaSupported : uint32
MaxMediaSize : uint64
DefaultBlockSize : uint64 {units}
MaxBlockSize : uint64 {units}
MinBlockSize : uint64 {units}
NeedsCleaning : boolean
MediaIsLocked : boolean
Security : uint16 {enum}
LastCleaned : datetime
MaxAccessTime : uint64 {units}
UncompressedDataRate : uint32 {units}
LoadTime : uint64 {units}
UnloadTime : uint64 {units}
MountCount : uint64
TimeOfLastMount : datetime
TotalMountTime : uint64
UnitsDescription : string
MaxUnitsBeforeCleaning : uint64
UnitsUsed : uint64

LockMedia ([IN] Lock : boolean) : uint32

DiskDrive

DisketteDrive

CDROMDrive

DVDDrive

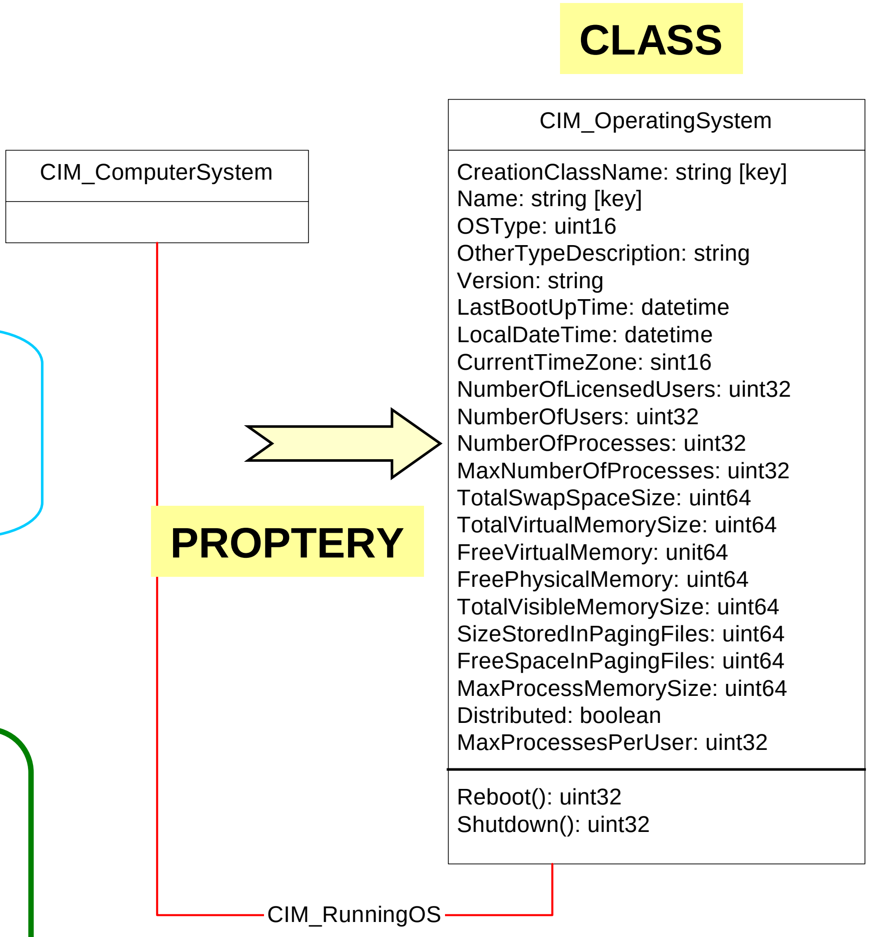
FormatsSupported : uint16[]

CIP uses a hierarchical, object oriented architecture for modeling managed resources.

Common Information Model

A **CIM Schema** contains a formal description of data, and actions on data, that is of interest to management applications.

Oh … I see … NumberOfProcess is a Property of the class CIM_OperatingSystem.



CLASS

CIM_ComputerSystem

CIM_OperatingSystem

CreationClassName: string [key]
Name: string [key]
OSType: uint16
OtherTypeDescription: string
Version: string
LastBootUpTime: datetime
LocalDateTime: datetime
CurrentTimeZone: sint16
NumberOfLicensedUsers: uint32
NumberOfUsers: uint32
NumberOfProcesses: uint32
MaxNumberOfProcesses: uint32
TotalSwapSpaceSize: uint64
TotalVirtualMemorySize: uint64
FreeVirtualMemory: unit64
FreePhysicalMemory: uint64
TotalVisibleMemorySize: uint64
SizeStoredInPagingFiles: uint64
FreeSpaceInPagingFiles: uint64
MaxProcessMemorySize: uint64
Distributed: boolean
MaxProcessesPerUser: uint32

Reboot(): uint32
Shutdown(): uint32

PROPTERY

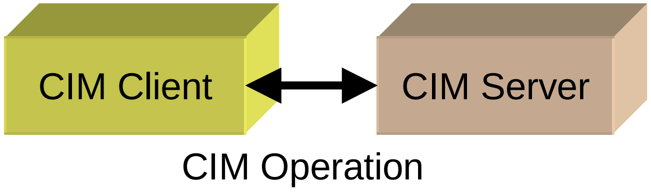
CIM_RunningOS

Using CIM, WBEM provides a platform and resource neutral mechanism for management applications to describe a request to access a managed resource.

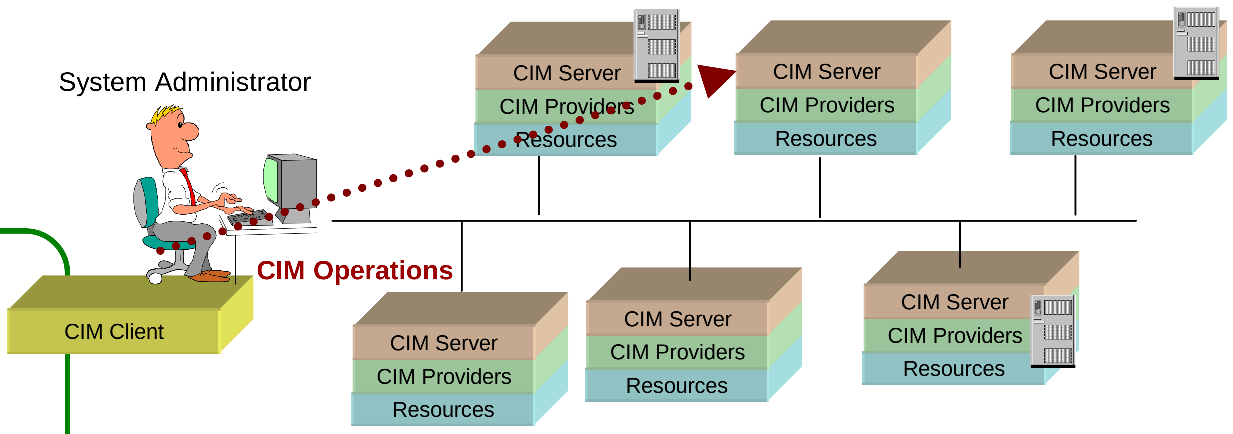
CIM Operations

A **CIM Operation** describes a management action (i.e., a monitor or control request) on a CIM modeled resource.

A **CIM Client** sends CIM Operation requests and receives CIM Operation responses.



A **CIM Server** receives CIM Operation requests and sends CIM Operation responses.



A CIM Client can be used to monitor and control local and/or remote resources.

Web-Based Enterprise Mgmt

WBEM is a DMTF Standard that is defined by a set of standards that include:

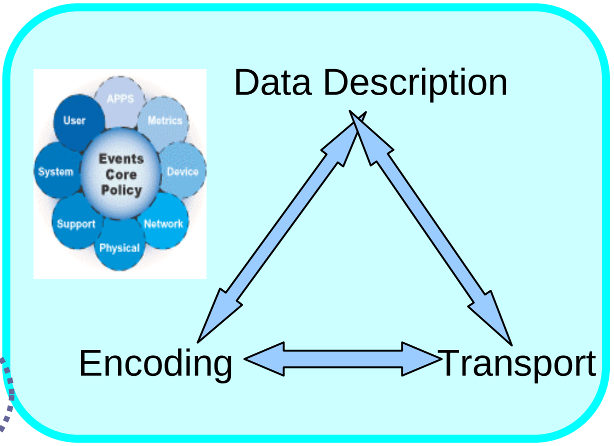
WHAT

- **A Data Description Standard** that describes the resources to be managed.

HOW

- **A Communication Protocol Standard** that defines an encoding and a transport protocol.

DMTF distributed management task force, inc.



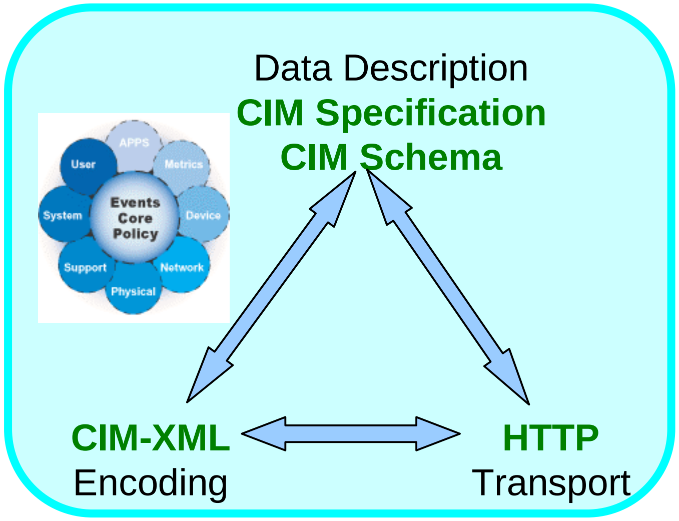
DMTF Board Members: 3Com, Cisco, Dell Computer Corp., Hewlett-Packard Company, IBM/Tivoli Systems, Inc., Intel Corporation, Microsoft Corporation, NEC Corporation, Novell, Oracle, Sun Microsystems, Inc., Symantec Corporation, VERITAS Software

DMTF Specifications and Schema are available at: http://www.dmtf.org

Communication Protocol

CIM-XML is a DMTF Standard WBEM Communication Protocol. It includes:

- Use of **the CIM Specification & CIM Schema** for the representation of managed resources.
- **A CIM-XML Encoding:** A standard for encoding CIM data and operations into XML.
- **An HTTP Transport Protocol:** The definition of a standard protocol for transporting CIM-XML encoded requests and responses over HTTP.



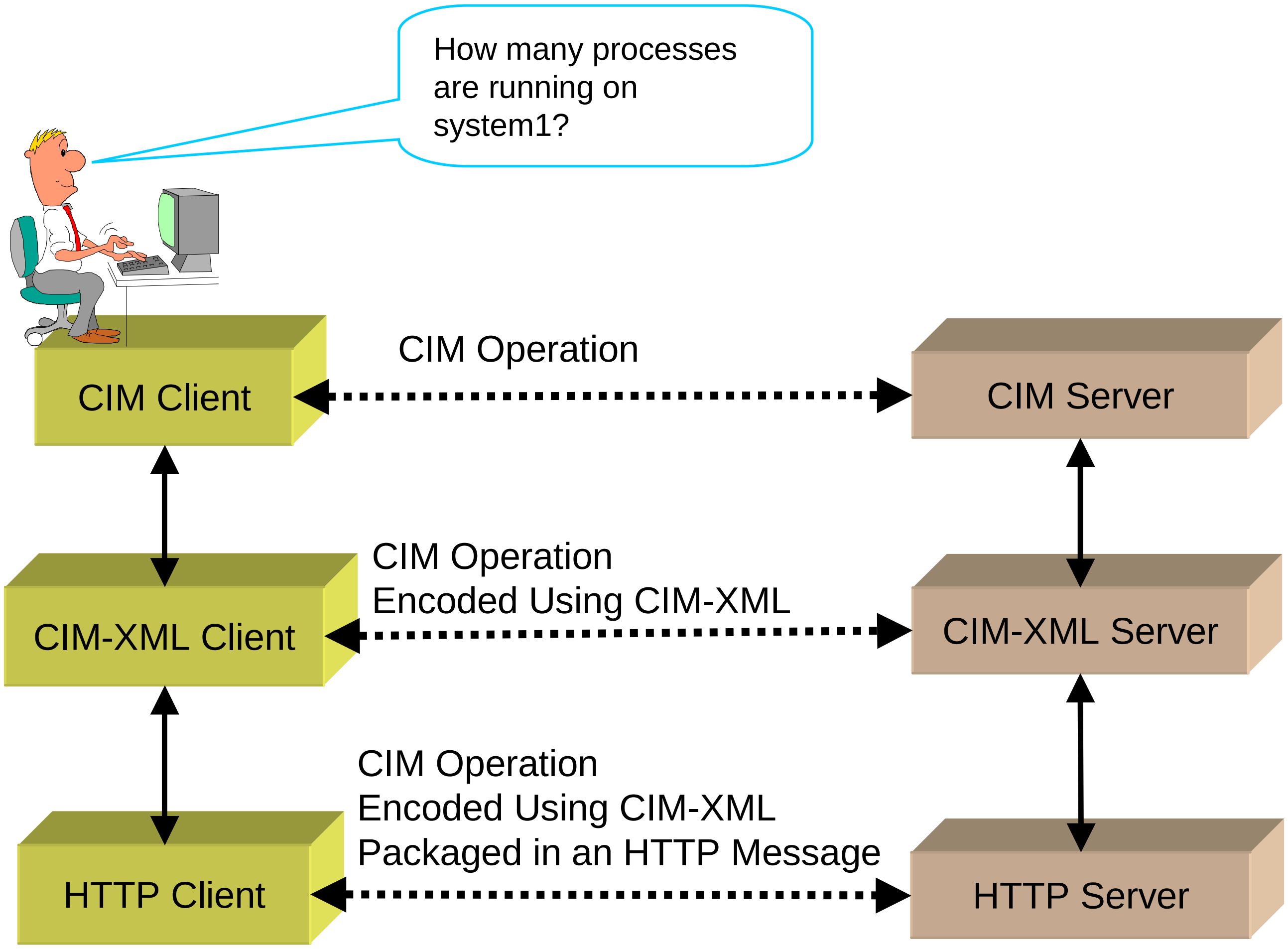
Data Description
CIM Specification
CIM Schema

CIM-XML
Encoding

HTTP
Transport

HOW

CIM-XML WBEM Protocol

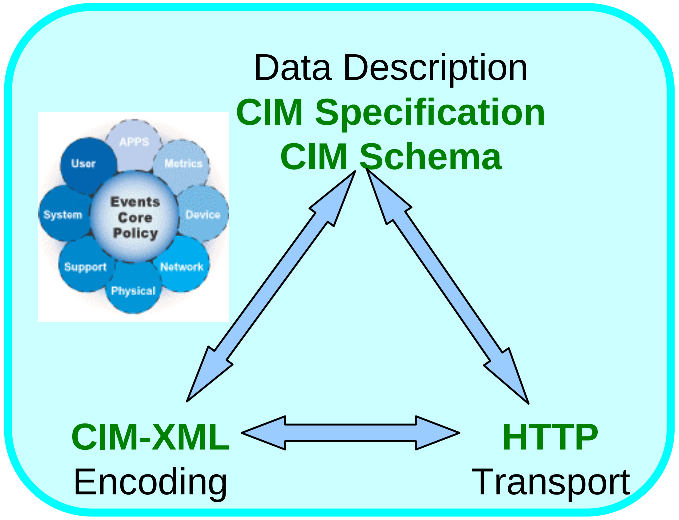


Review

- **Web-Based Enterprise Management (WBEM)** is a DMTF standard for monitoring and controlling resources from diverse sources.

- The **Common Information Model (CIM)** is the DMTF WBEM Standard
for describing data.

- **CIM-XML** is a Standard WBEM Communication Protocol.



Data Description
CIM Specification
CIM Schema

CIM-XML
Encoding

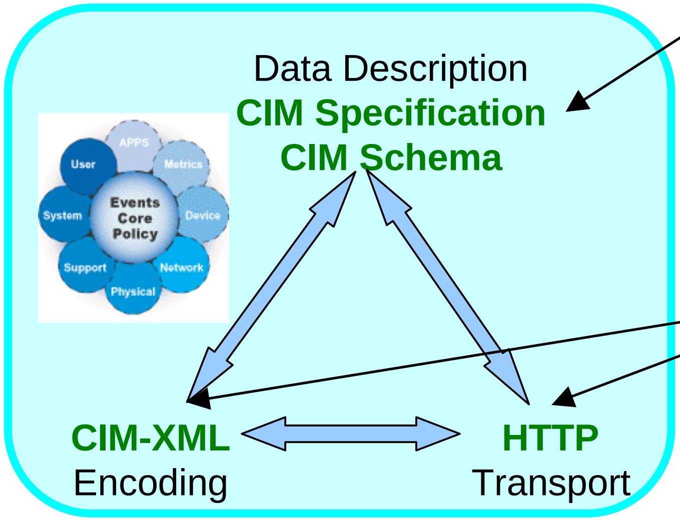
HTTP
Transport

Module Content

WBEM Overview

- Problem Statement
- Terminology
- **CIM Data Model**
- CIM Operations
- CIM-XML Communication Protocol

Terminology



Data Description - Formal description of the semantics of the data. CIM addresses the **WHAT?** component of the standard.

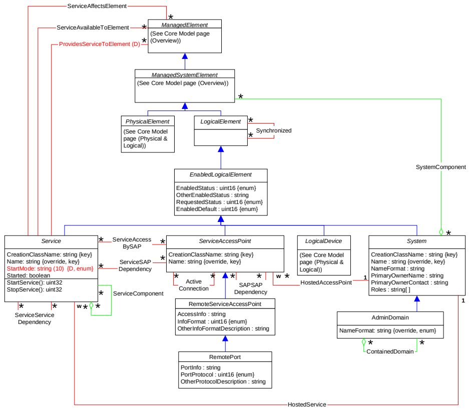
Communication Protocol - Formal description of the protocol and encoding used between the client and sever. These components address the **HOW?** aspects the standard.

CIM Data Model

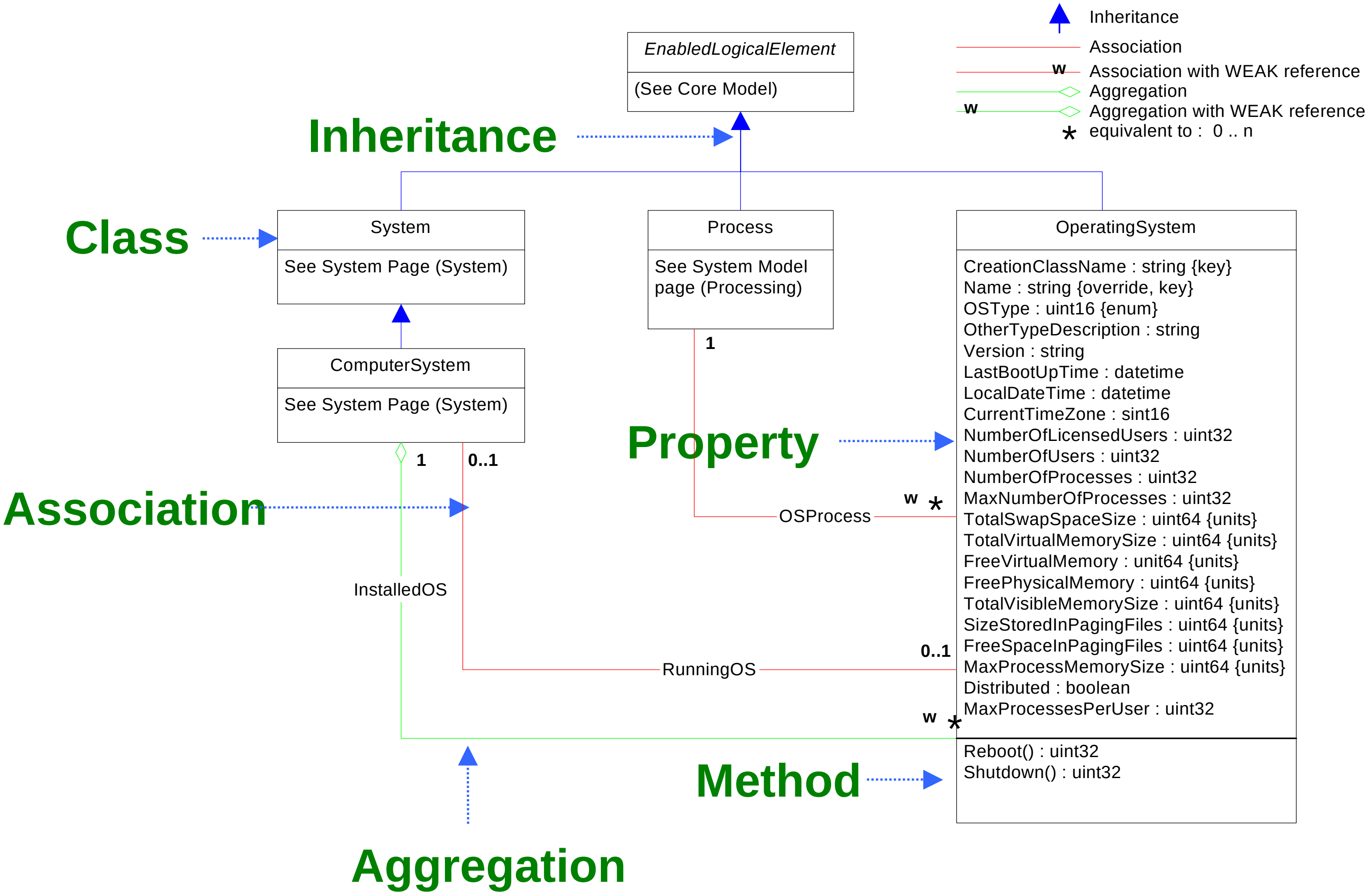
The Data Description consists of

1. A formal definition of how the data is described (contained in the *Common Information Model (CIM) Specification Version 2.2*: http://www.dmtf.org/standards/cim_spec_v22/).

2. A formal description of the resources to be managed.
   a) DMTF-defined Schema (Application, Core, Device, Event, InterOp, Metric, Physical, Policy, Support, System, User, Network) DMTF CIM Schema v2.7 (Final), v2.8 (Preliminary)
   b) Vendor or Customer-defined Extensions to the Schema


ServiceAffectsElement
ServiceAvailableToElement
ProvidesServiceToElement (D)
ManagedElement
(See Core Model page (Overview))
ManagedSystemElement
(See Core Model page (Overview))
PhysicalElement
(See Core Model page (Physical & Logical))
LogicalElement
Synchronized
EnabledLogicalElement
EnabledStatus : uint16 {enum}
OtherEnabledStatus : string
RequestedStatus : uint16 {enum}
EnabledDefault : uint16 {enum}
SystemComponent
Service
CreationClassName: string {key}
Name: string {override, key}
StartMode: string (10) {D, enum}
Started: boolean
StartService(): uint32
StopService(): uint32
ServiceAccess BySAP
ServiceSAP Dependency
ServiceComponent
ServiceService Dependency
ServiceAccessPoint
CreationClassName: string {key}
Name: string {override, key}
Active Connection
SAPSAP Dependency
HostedAccessPoint
LogicalDevice
(See Core Model page (Physical & Logical))
System
CreationClassName : string {key}
Name : string {override, key}
NameFormat : string
PrimaryOwnerName : string
PrimaryOwnerContact : string
Roles : string[ ]
RemoteServiceAccessPoint
AccessInfo : string
InfoFormat : uint16 {enum}
OtherInfoFormatDescription : string
RemotePort
PortInfo : string
PortProtocol : uint16 {enum}
OtherProtocolDescription : string
AdminDomain
NameFormat: string {override, enum}
ContainedDomain
HostedService
*
w
1

CIM Concepts



Inheritance

Association

w Association with WEAK reference

Aggregation

w Aggregation with WEAK reference

* equivalent to : 0 .. n

EnabledLogicalElement

(See Core Model)

Inheritance

Class

System

See System Page (System)

Process

See System Model page (Processing)

OperatingSystem

CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32

Reboot() : uint32
Shutdown() : uint32

ComputerSystem

See System Page (System)

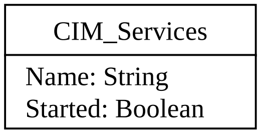
CIM Concepts

Class

CIM_Service

A **Class** is a collection of entities, called **instances**, that can be described using a common set of properties and operations.

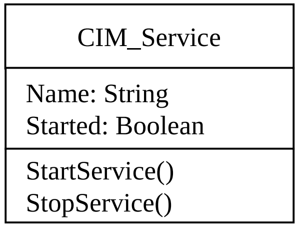
Class with Properties


CIM_Services

Name: String
Started: Boolean

A **Property** is the definition of a value that describes a characteristic of the instances of a class.

Methods

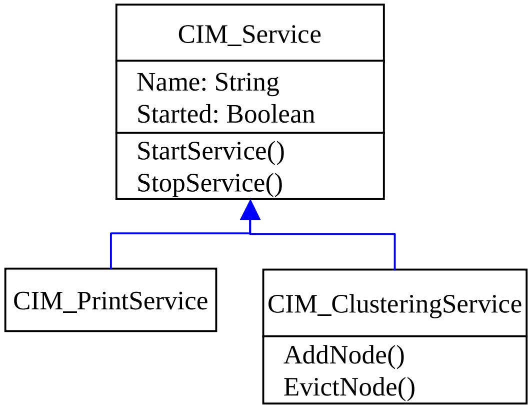

CIM_Service

Name: String
Started: Boolean

StartService()
StopService()

A **Method** is the definition of an operation on the class or instances of the class.

Inheritance


CIM_Service

Name: String
Started: Boolean

StartService()
StopService()

CIM_PrintService

CIM_ClusteringService

AddNode()
EvictNode()

Inheritance is used to describe the existence of a superclass/subclass relationship between classes. In particular, properties and methods declared for a superclass are inherited by all of it subclasses.

Inheritance : Blue
Association: Red
Aggregation: Green

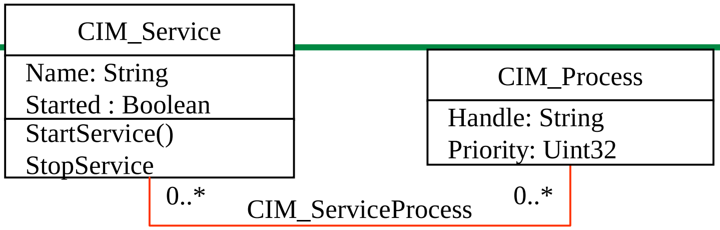
UML = Unified Modeling Language

CIM Concepts

An **Association** describes a relationship between two classes.

An **Aggregation** is a special type of association where an instance of one class is described by a grouping of instances of the other class.

Association

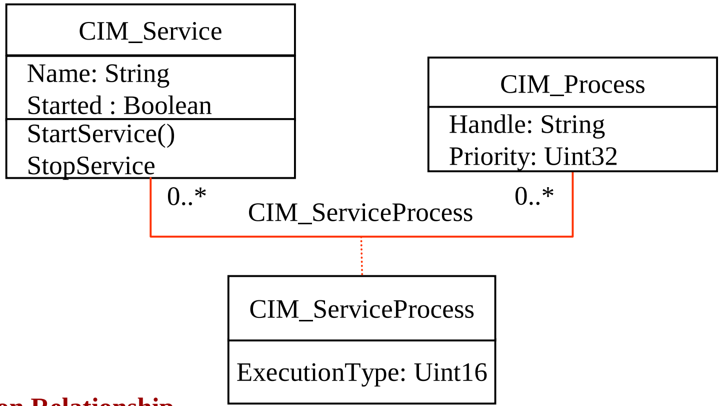


CIM_Service
Name: String
Started : Boolean
StartService()
StopService

CIM_Process
Handle: String
Priority: Uint32

0..* CIM_ServiceProcess 0..*

Association with Properties



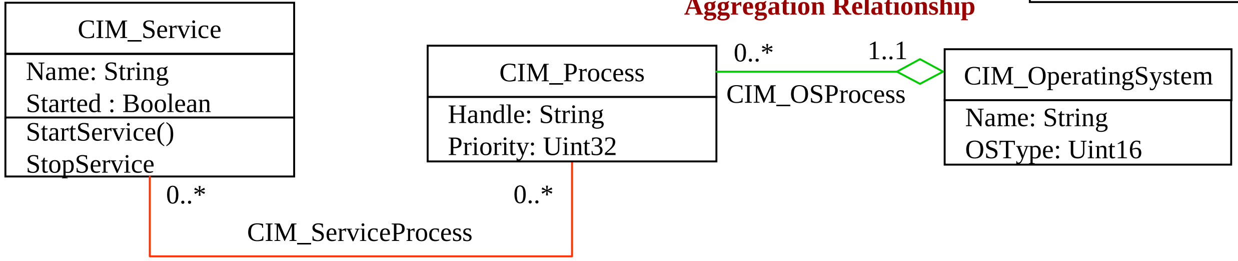
CIM_Service
Name: String
Started : Boolean
StartService()
StopService

CIM_Process
Handle: String
Priority: Uint32

0..* CIM_ServiceProcess 0..*

CIM_ServiceProcess
ExecutionType: Uint16

Aggregation Relationship



CIM_Service
Name: String
Started : Boolean
StartService()
StopService

CIM_Process
Handle: String
Priority: Uint32

0..* CIM_ServiceProcess 0..*

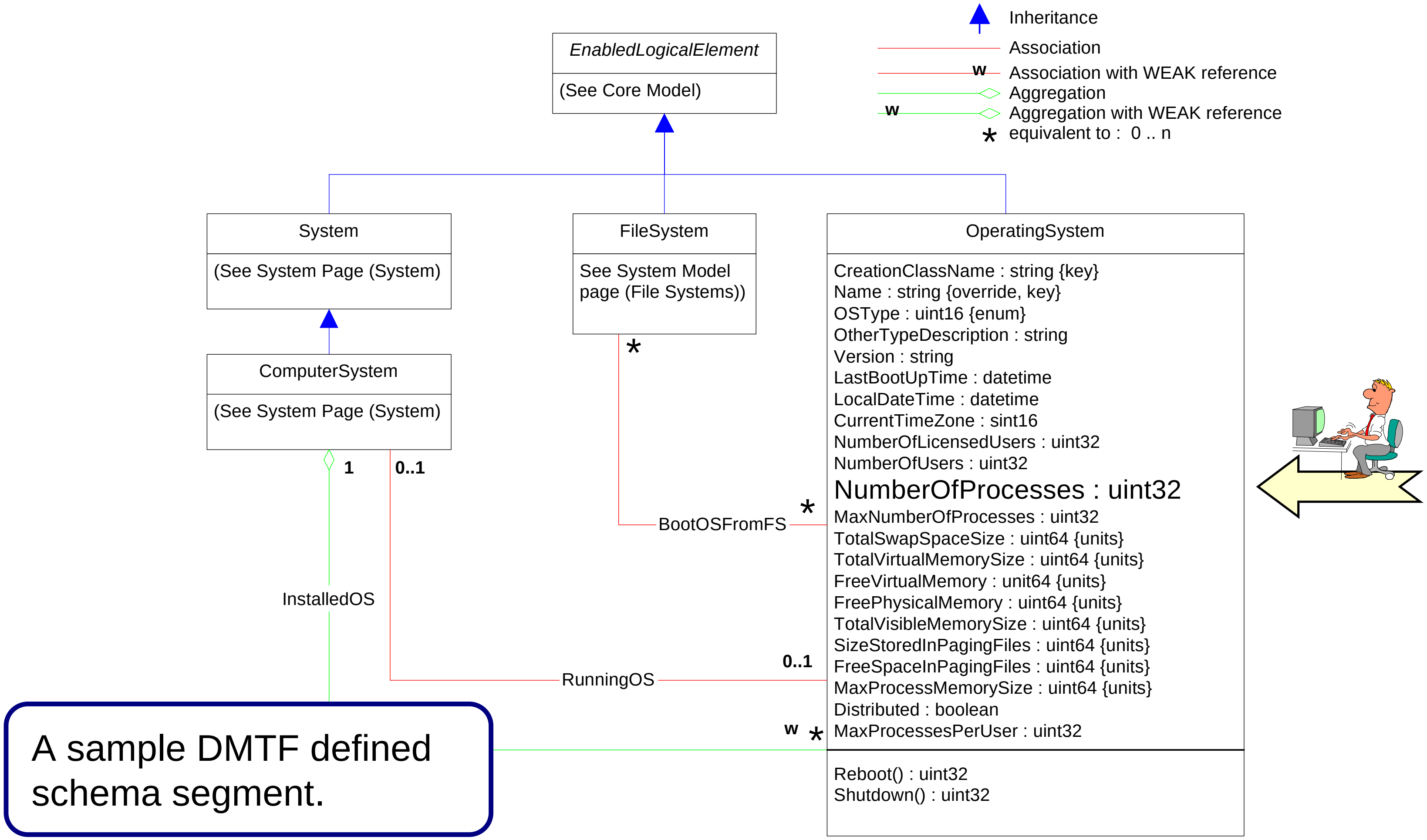
0..* 1..1
CIM_OSProcess

CIM_OperatingSystem
Name: String
OSType: Uint16

Inheritance : Blue
Association: Red
Aggregation: Green

THE Open GROUP
OpenPegasus Developer Conference

CIM Example



▲ Inheritance
🟥 Association
🟥 w Association with WEAK reference
🟩◇ Aggregation
🟩 w ◇ Aggregation with WEAK reference
* equivalent to : 0 .. n

EnabledLogicalElement
(See Core Model)

System
(See System Page (System))

ComputerSystem
(See System Page (System))

FileSystem
See System Model page (File Systems))

OperatingSystem

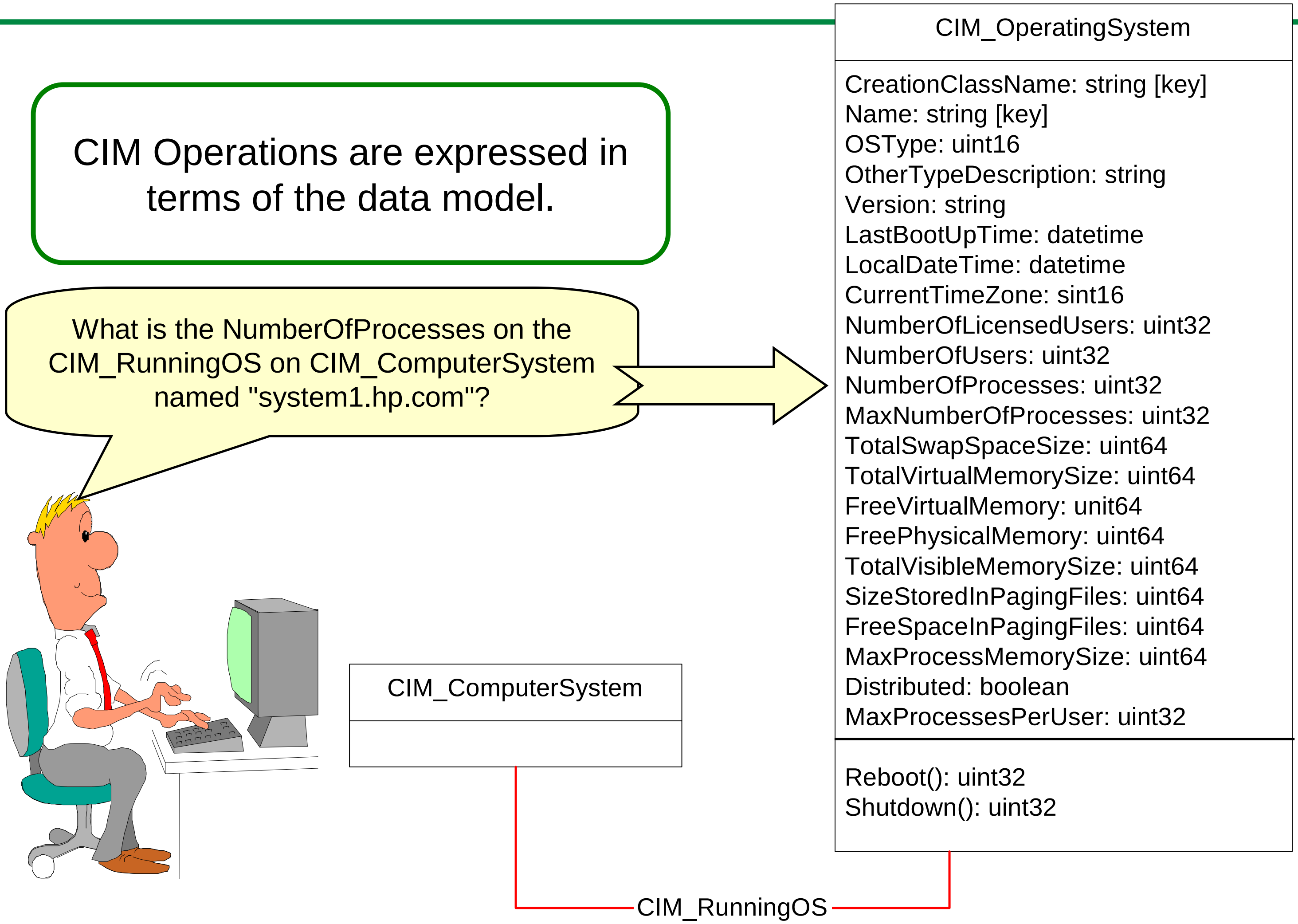
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32

Reboot() : uint32
Shutdown() : uint32

A sample DMTF defined schema segment.

CIM Operation

CIM Operations are expressed in terms of the data model.



What is the NumberOfProcesses on the CIM_RunningOS on CIM_ComputerSystem named "system1.hp.com"?

CIM_ComputerSystem

CIM_OperatingSystem

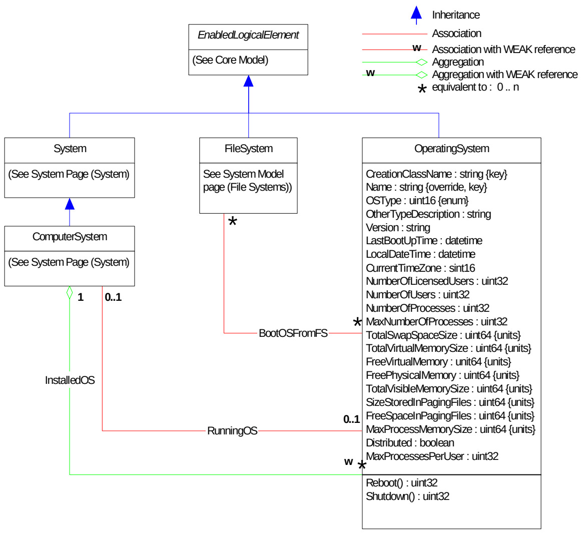
CreationClassName: string [key]
Name: string [key]
OSType: uint16
OtherTypeDescription: string
Version: string
LastBootUpTime: datetime
LocalDateTime: datetime
CurrentTimeZone: sint16
NumberOfLicensedUsers: uint32
NumberOfUsers: uint32
NumberOfProcesses: uint32
MaxNumberOfProcesses: uint32
TotalSwapSpaceSize: uint64
TotalVirtualMemorySize: uint64
FreeVirtualMemory: unit64
FreePhysicalMemory: uint64
TotalVisibleMemorySize: uint64
SizeStoredInPagingFiles: uint64
FreeSpaceInPagingFiles: uint64
MaxProcessMemorySize: uint64
Distributed: boolean
MaxProcessesPerUser: uint32

Reboot(): uint32
Shutdown(): uint32

CIM_RunningOS

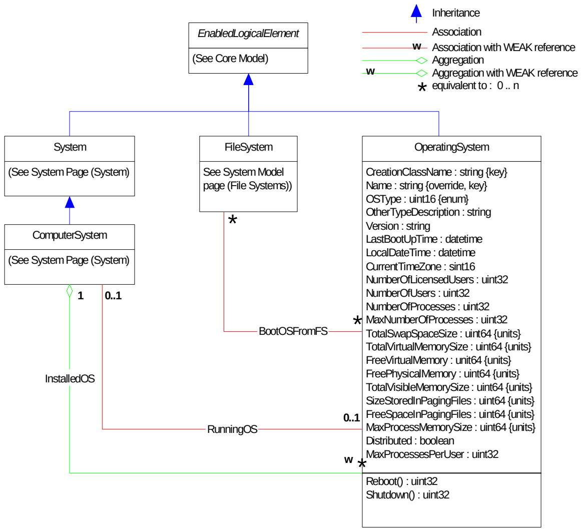
Managed Object Format

We need a formal syntax for expressing Schema.


Inheritance
Association
W
Association with WEAK reference
Aggregation
W
Aggregation with WEAK reference
* equivalent to : 0 .. n
EnabledLogicalElement
(See Core Model)
System
(See System Page (System)
FileSystem
See System Model page (File Systems))
OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32
ComputerSystem
(See System Page (System)
1
0..1
*
*
BootOSFromFS
InstalledOS
RunningOS
0..1
W
*

```
class CIM_OperatingSystem : CIM_EnabledLogicalElement {

…


    [Description ("Time when the OperatingSystem was last booted."),
        MappingStrings {"MIF.DMTF|General Information|001.5"} ]
    datetime LastBootUpTime;

    [Description (
        "OperatingSystem's notion of the local date and time of day."),
        MappingStrings {"MIB.IETF|HOST-RESOURCES-MIB.hrSystemDate",
            "MIF.DMTF|General Information|001.6"} ]
    datetime LocalDateTime;

    [Description (
        "Number of process contexts currently loaded or running on "
        "the OperatingSystem."), Gauge,
        MappingStrings {"MIF.DMTF|Host System|001.5",
            "MIB.IETF|HOST-RESOURCES-MIB.hrSystemProcesses"} ]
    uint32 NumberOfProcesses;

    [Description (
        "Number of user licenses for the OperatingSystem. "
        "If unlimited, enter 0.") ]
    uint32 NumberOfLicensedUsers;


…. .
```

Managed Object Format


Inheritance
Association
W Association with WEAK reference
Aggregation
W Aggregation with WEAK reference
* equivalent to : 0 .. n
EnabledLogicalElement
(See Core Model)
System
(See System Page (System)
ComputerSystem
(See System Page (System)
FileSystem
See System Model page (File Systems))
OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32
BootOSFromFS
*
*
1
0..1
InstalledOS
RunningOS
0..1
w
*

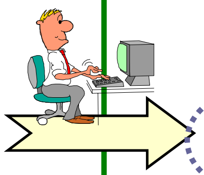
```
class CIM_OperatingSystem : CIM_EnabledLogicalElement {

…


    [Description ("Time when the OperatingSystem was last booted."),
        MappingStrings {"MIF.DMTF|General Information|001.5"} ]
    datetime LastBootUpTime;

    [Description (
        "OperatingSystem's notion of the local date and time of day."),
        MappingStrings {"MIB.IETF|HOST-RESOURCES-MIB.hrSystemDate",
            "MIF.DMTF|General Information|001.6"} ]
    datetime LocalDateTime;

    [Description (
        "Number of process contexts currently loaded or running on "
        "the OperatingSystem."), Gauge,
        MappingStrings {"MIF.DMTF|Host System|001.5",
            "MIB.IETF|HOST-RESOURCES-MIB.hrSystemProcesses"} ]
    uint32 NumberOfProcesses;

    [Description (
        "Number of user licenses for the OperatingSystem. "
        "If unlimited, enter 0.") ]
    uint32 NumberOfLicensedUsers;


…. .
```

Managed Object Format (MOF) is the language defined by the DMTF for describing classes and instances.

A **MOF File** is a text file that contains definitions of classes and instances expressed using the MOF language.

Managed Object Format

```
class CIM_OperatingSystem : CIM_EnabledLogicalElement {

…
    [Description ("Time when the OperatingSystem was last booted."),
        MappingStrings {"MIF.DMTF|General Information|001.5"} ]
    datetime LastBootUpTime;

    [Description (
        "Number of process contexts currently loaded or "
        "running on the OperatingSystem."), Gauge,
        MappingStrings {"MIF.DMTF|Host System|001.5",
            "MIB.IETF|HOST-RESOURCES-MIB.hrSystemProcesses"} ]
    uint32 NumberOfProcesses;

    [Description (
        "Number of user licenses for the OperatingSystem. "
        "If unlimited, enter 0.") ]
    uint32 NumberOfLicensedUsers;
…..
```



Module Content

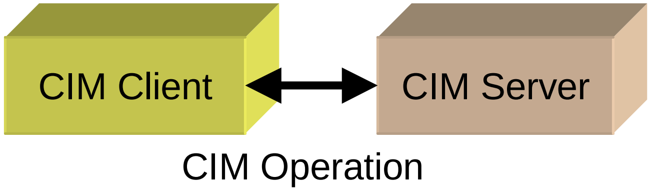
WBEM Overview

- Problem Statement
- Terminology
- CIM Data Model
- **CIM Operations**
- CIM-XML Communication Protocol

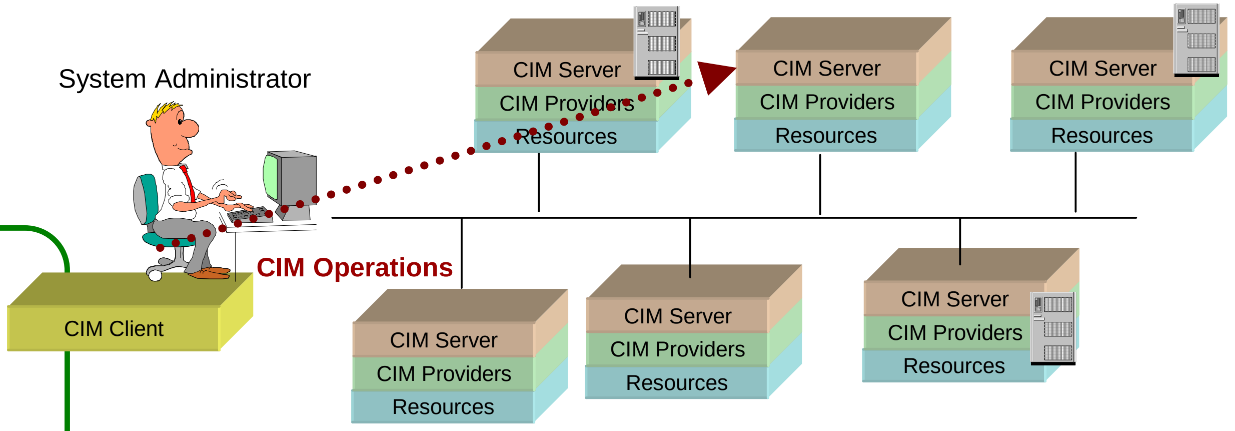
Terminology

A **CIM Operation** describes a management action (i.e., a monitor or control request) on a CIM modeled resource.

A **CIM Client** sends CIM Operation requests and receives CIM Operation responses.



A **CIM Server** receives CIM Operation requests and sends CIM Operation responses.

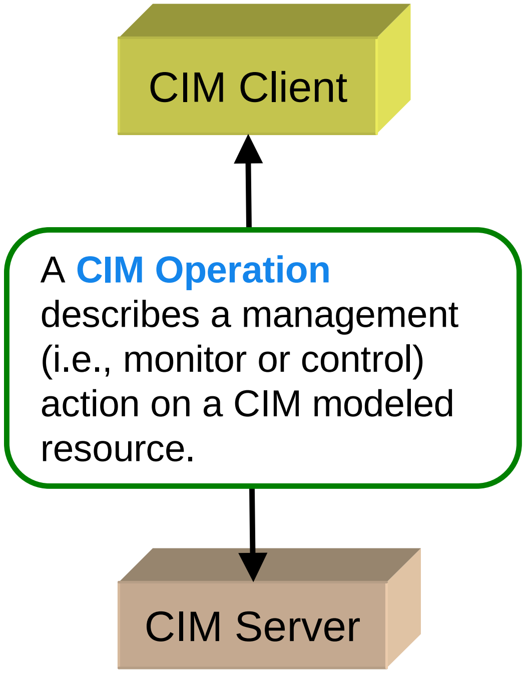


A CIM Client can be used to monitor and control local and/or remote resources.

CIM Operation Overview

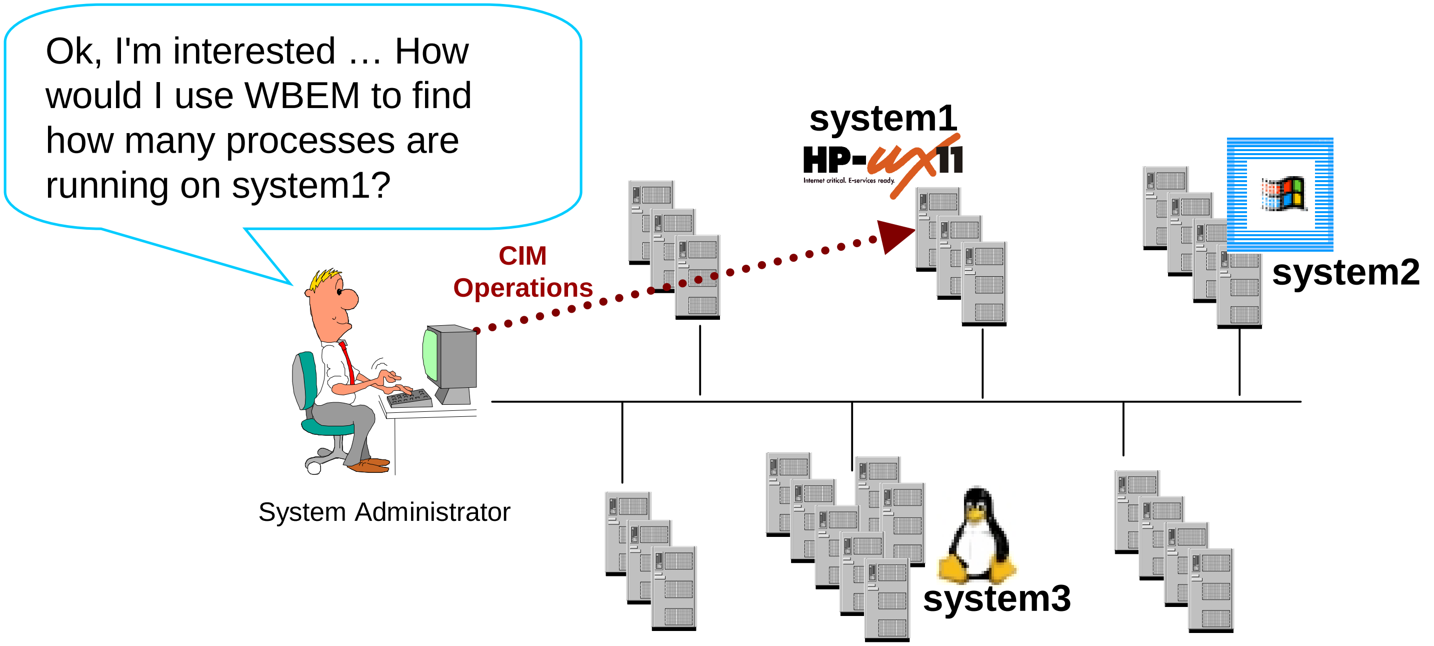
The DMTF has defined a set of CIM Operations.

Functional Group	CIM Operations
Basic read	GetClass, EnumerateClasses, EnumerateClassNames, GetInstance, EnumerateInstances, EnumerateInstanceNames, GetProperty
Basic Write	SetProperty
Schema Manipulation	CreateClass, ModifyClass, DeleteClass
Instance Manipulation	CreateInstance, ModifyInstance, DeleteInstance
Association Traversal	Associators, AssociatorNames, References, ReferenceNames
Query	ExecQuery
Qualifier Declaration	GetQualifier, SetQualifier, DeleteQualifier, EnumerateQualifier



A **CIM Operation** describes a management (i.e., monitor or control) action on a CIM modeled resource.

CIM Operation Example



Ok, I'm interested … How would I use WBEM to find how many processes are running on system1?

GetProperty Example

The **GetProperty** operation can be used to retrieve a single property value from a CIM Instance in the target Namespace.

I understand that
(1) the value of PropertyName will be "NumberOfProcesses" and
(2) the actual value will be returned as a 32-bit unsigned integer …
But what is InstanceName? …
And what's a Namespace? …

GetProperty

<propertyValue> **GetProperty** (
    [IN] <instanceName> InstanceName,
    [IN] <string> PropertyName)

MOF Fragment Describing NumberOfProcesses Property

```
[Description (
        "Number of process contexts currently loaded or "
        "running on the OperatingSystem."), Gauge,
        MappingStrings {"MIF.DMTF|Host System|001.5",
"MIB.IETF|HOST-RESOURCES-MIB.hrSystemProcesses"} ]
uint32 NumberOfProcesses;
```

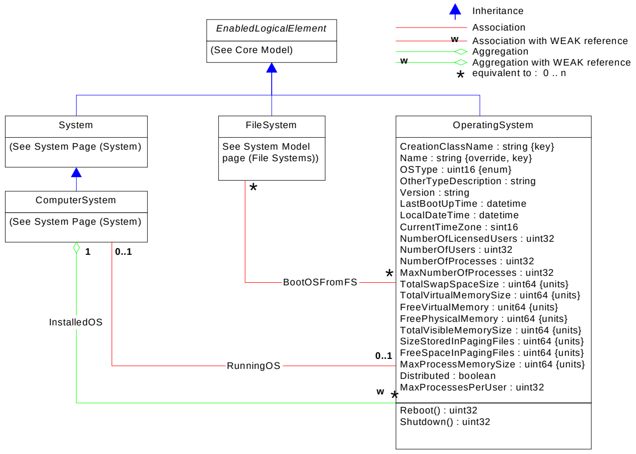
Namespaces

A **Namespace** provides a context for grouping together the definitions of classes that describe the objects of interest in a managed environment.

Key Fact: All CIM Operations are associated with a target namespace.

Ok … So I also need to know the Namespace that contains the definition of CIM_OperatingSystem class.

Managed System Namespace



Inheritance
Association
Association with WEAK reference
Aggregation
Aggregation with WEAK reference
* equivalent to : 0 .. n

EnabledLogicalElement
(See Core Model)

System
(See System Page (System)

ComputerSystem
(See System Page (System)

FileSystem
See System Model page (File Systems))

OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32

Instance Names

An **<instanceName>** uniquely identifies a CIM Instance within a Namespace. It is comprised of a **class name** and the **key properties** defined for that class.

```
[Propagated ("CIM_ComputerSystem.CreationClassName"),
    Key, MaxLen (256), Description (
        "The scoping ComputerSystem's CreationClassName.") ]
string CSCreationClassName;

[Propagated ("CIM_ComputerSystem.Name"), Key, MaxLen (256),
    Description ("The scoping ComputerSystem's Name.") ]
string CSName;

[Key, MaxLen (256), Description (
    "CreationClassName indicates the name of the class or the "
    "subclass used in the creation of an instance. When used "
    "with the other key properties of this class, this property "
    "allows all instances of this class and its subclasses to "
    "be uniquely identified.") ]
string CreationClassName;

[Override ("Name"), Key, MaxLen (256), Description (
    "The inherited Name serves as key of an OperatingSystem "
    "instance within a ComputerSystem."),
    MappingStrings {"MIF.DMTF|Operating System|001.2"} ]
string Name;
```

The class name is easy. It's CIM_OperatingSystem. But what do I use for the key values?

Instance Names

In some cases determining an instance name can be tricky. There are two basic approaches.

1. The client "understands" the semantics of the keys and how they are constructed. In this case, the client can generate the key properties.
2. The client first "enumerates" the instance names and "selects" the correct one.

Parameters

GetProperty

<propertyValue> **GetProperty** (
[IN] <instanceName> InstanceName,
[IN] <string> PropertyName)

An **<instanceName>** uniquely identifies a CIM Instance within a Namespace. It is comprised of a class name and the key properties defined for that class.

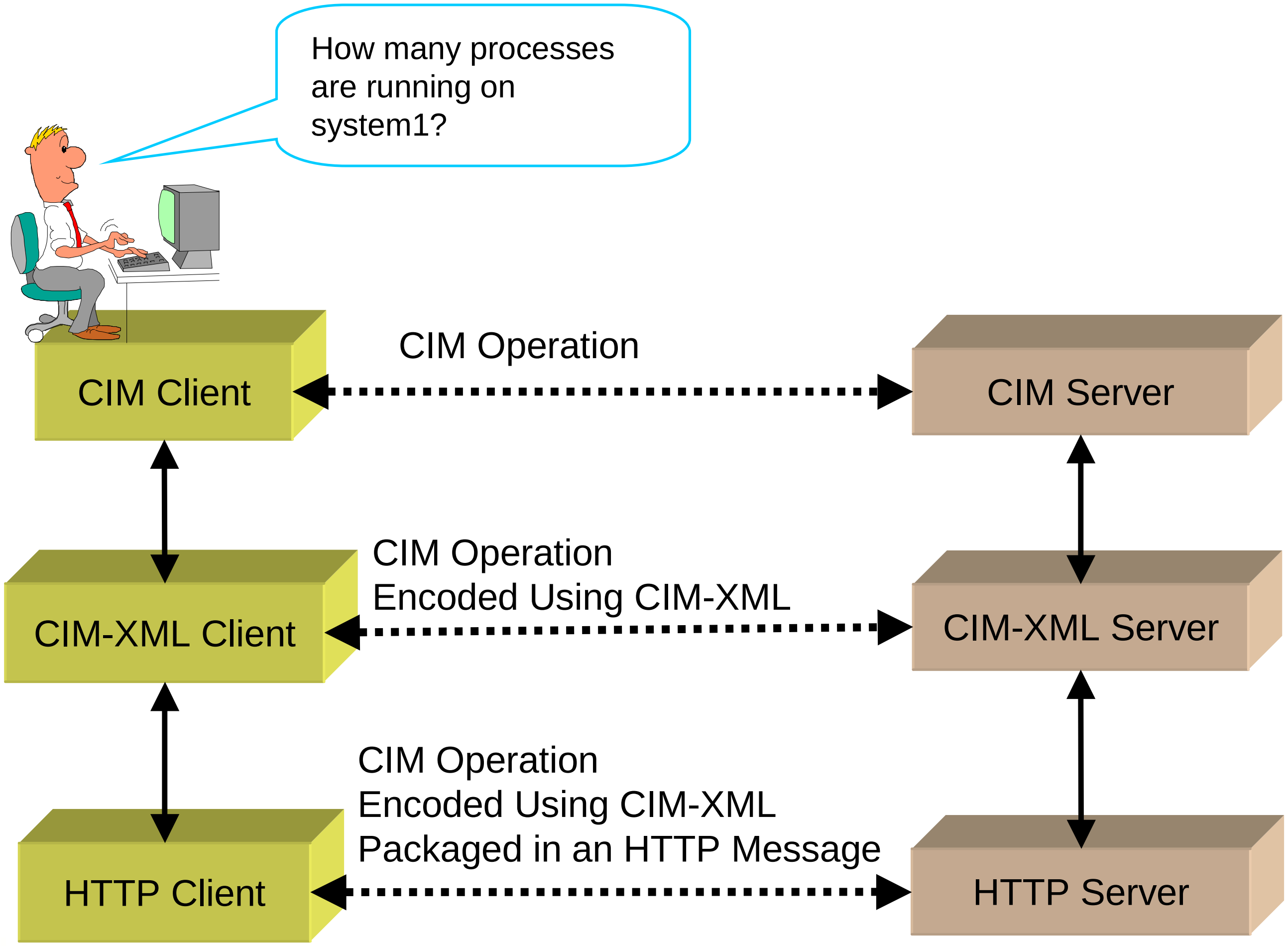
GetProperty		
Parameter		Value
Namespace		root/cimv2
InstanceName	Class Name	CIM_OperatingSystem
InstanceName	CSCreationClassName	CIM_UnitaryComputerSystem
InstanceName	CSName	system1.hp.com
InstanceName	CreateClassName	CIM_OperatingSystem
InstanceName	Name	HP-UX
PropertyName		NumberOfProcesses

Module Content

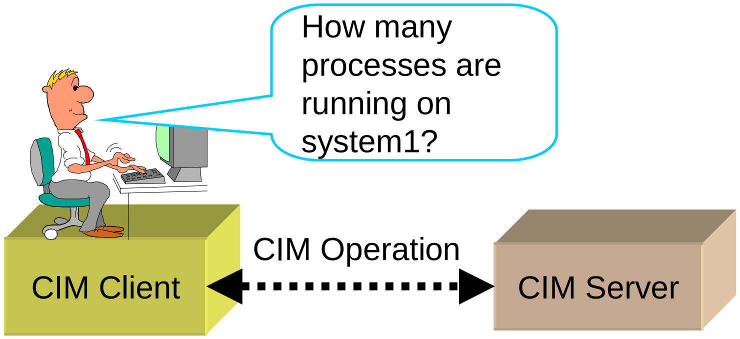
WBEM Overview

- Problem Statement
- Terminology
- CIM Data Model
- CIM Operations
- **CIM-XML Communication Protocol**

Communication Protocol



CIM Operation



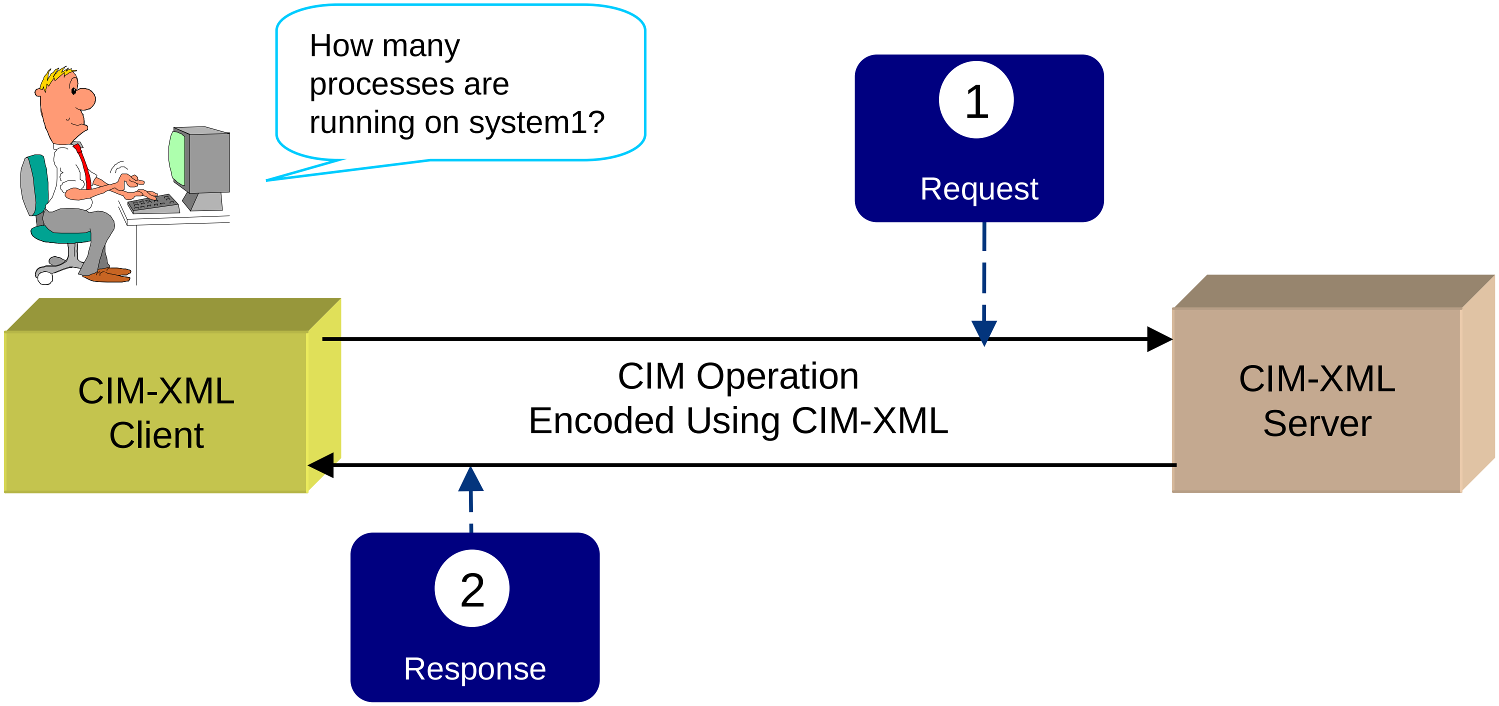
GetProperty

<propertyValue> **GetProperty** (
[IN] <instanceName> InstanceName,
[IN] <string> PropertyName)

GetProperty			
Parameter			Value
Type	Name		
Input	Namespace		root/cimv2
Input	InstanceName	CSCreationClassName	CIM_UnitaryComputerSystem
Input	InstanceName	CSName	system1.hp.com
Input	InstanceName	CreateClassName	CIM_OperatingSystem
Input	InstanceName	Name	HP-UX
Input	PropertyName		NumberOfProcesses
Output			74

CIM-XML Encoding

CIM-XML Encoding: Defines a standard encoding of CIM Operations and CIM data into XML.



How many processes are running on system1?

1 Request

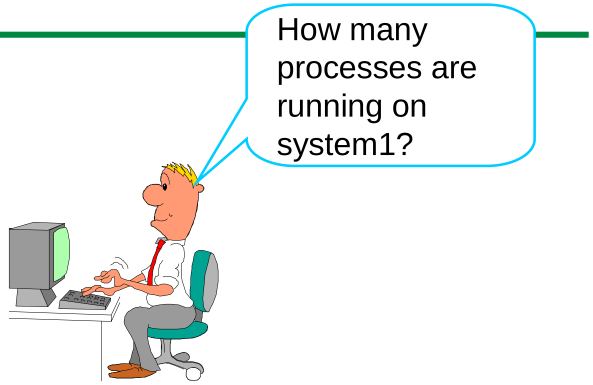
CIM-XML Client

CIM Operation Encoded Using CIM-XML

CIM-XML Server

2 Response

CIM-XML Request



```
<?xml version="1.0" encoding="utf-8" ?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
 <MESSAGE ID="82000" PROTOCOLVERSION="1.0">
  <SIMPLEREQ>
   <IMETHODCALL NAME="GetProperty">
    <LOCALNAMESPACEPATH>
     <NAMESPACE NAME="root" />
     <NAMESPACE NAME="cimv2" />
    </LOCALNAMESPACEPATH>
    <IPARAMVALUE NAME="InstanceName">
     <INSTANCENAME CLASSNAME="CIM_OperatingSystem">
      <KEYBINDING NAME="CSCreationClassName">
       <KEYVALUE VALUETYPE="string">CIM_UnitaryComputerSystem</KEYVALUE>
      </KEYBINDING>
      <KEYBINDING NAME="CSName">
       <KEYVALUE VALUETYPE="string">system1.hp.com</KEYVALUE>
      </KEYBINDING>
      <KEYBINDING NAME="CreationClassName">
       <KEYVALUE VALUETYPE="string">CIM_OperatingSystem</KEYVALUE>
      </KEYBINDING>
      <KEYBINDING NAME="Name">
       <KEYVALUE VALUETYPE="string">HP-UX</KEYVALUE>
      </KEYBINDING>
     </INSTANCENAME>
    </IPARAMVALUE>
    <IPARAMVALUE NAME="PropertyName">
     <VALUE>NumberOfProcesses</VALUE>
    </IPARAMVALUE>
    …
```

Operation

Namespace

Class of Interest

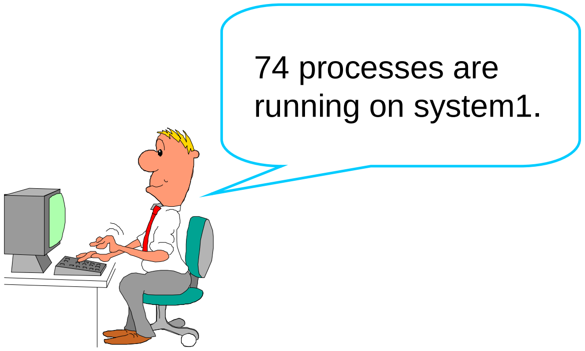
Keys identifying the Instance of Interest

Properties of Interest

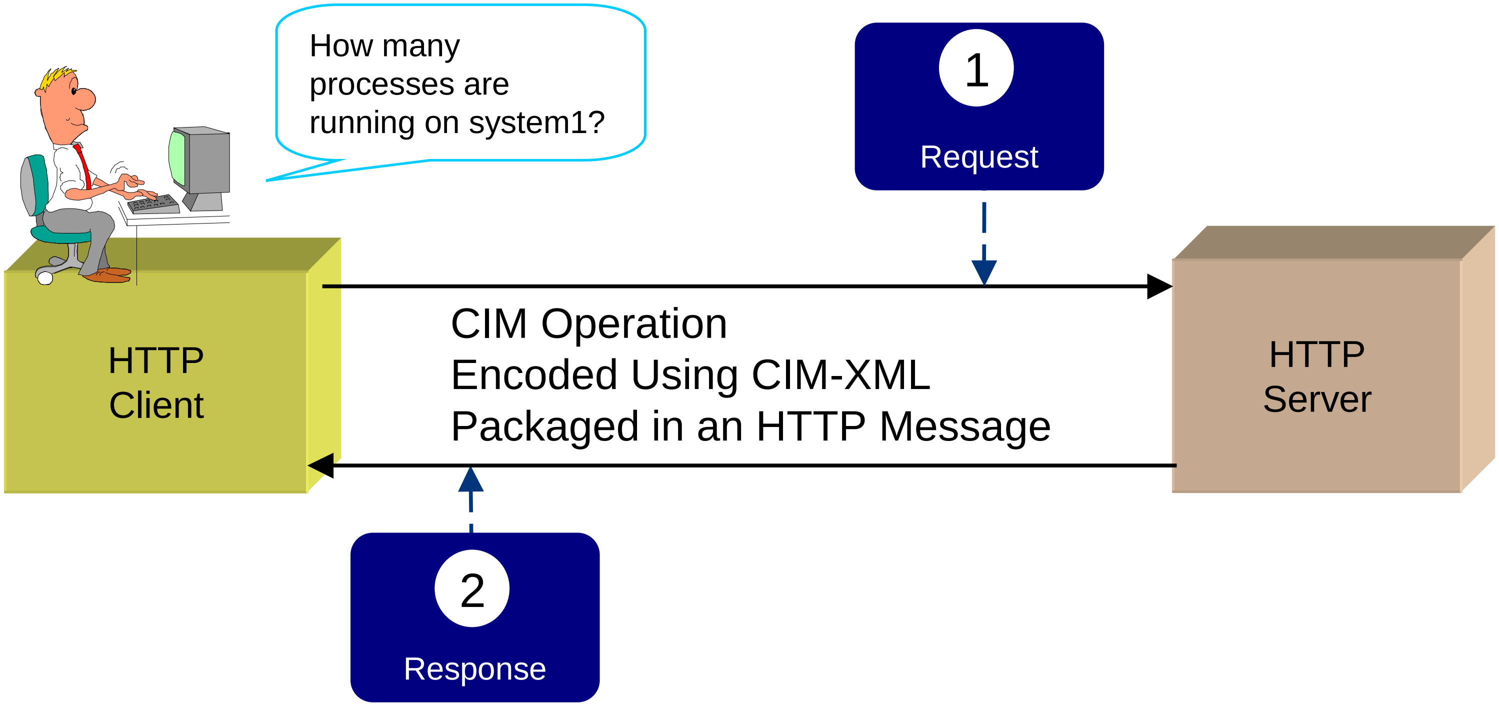
CIM-XML Response

```
<?xml version="1.0" encoding="utf-8" ?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
 <MESSAGE ID="82000" PROTOCOLVERSION="1.0">
  <SIMPLERSP>
   <IMETHODRESPONSE NAME="GetProperty">
    <IRETURNVALUE>
     <VALUE>74</VALUE>
    </IRETURNVALUE>
   </IMETHODRESPONSE>
  </SIMPLERSP>
 </MESSAGE>
</CIM>
```

Return Value



HTTP Transport



How many processes are running on system1?

1 Request

HTTP Client

CIM Operation
Encoded Using CIM-XML
Packaged in an HTTP Message

HTTP Server

2 Response

HTTP Message

```
M-POST /cimom HTTP/1.1
Host: system1
Content-Type: application/xml; charset="utf-8"
Content-Length: 1343
Man: http://www.dmtf.org/cim/mapping/http/v1.0;ns=34
34-CIMProtocolVersion: 1.0
34-CIMOperation: MethodCall
34-CIMMethod: GetProperty
34-CIMObject: root/cimv2
        <CIM-XML payload containing CIM Operation Request>
```

1

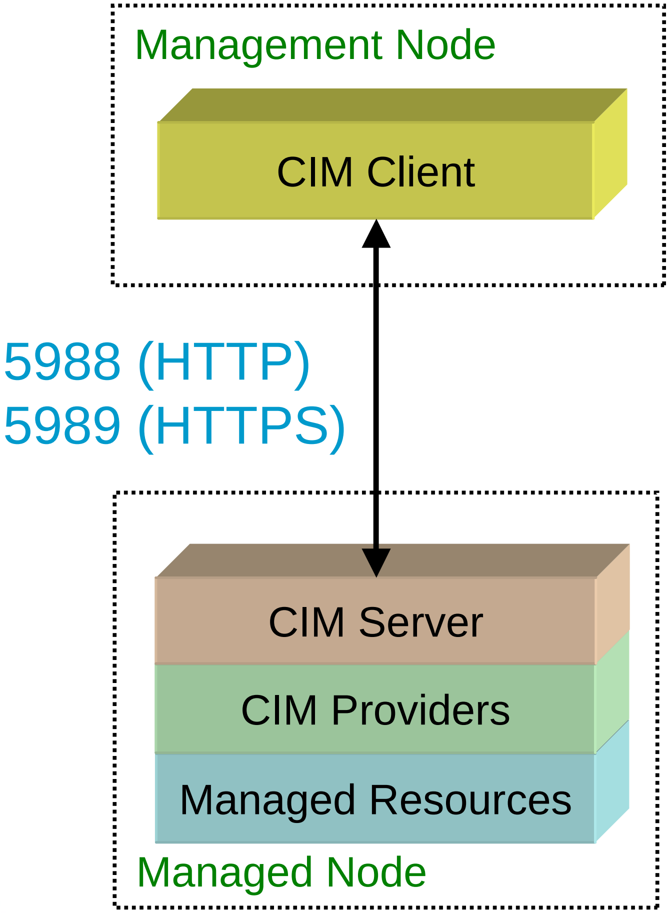
Request

```
HTTP/1.1 200 OK
Content-Type: application/xml; charset="utf-8"
Content-Length: 271
Cache-Control: no-cache
Man: http://www.dmtf.org/cim/mapping/http/v1.0; ns=64
64-CIMOperation: MethodResponse
        <CIM-XML payload containing CIM Operation Response>
```

2

Response

Port Numbers



The DMTF recommends the use of the following well-known IP ports for use in compliant CIM Servers. This is a recommendation only and not a requirement for compliance with this specification. These port addresses have been acquired from IANA by the DMTF and are registered with IANA so are for the exclusive use for DMTF functions, in particular CIM Servers.

CIM-XML(http)	5988/tcp
CIM-XML(http)	5988/udp
CIM-XML(https)	5989/tcp
CIM-XML(https)	5989/udp

OpenPegasus Overview

Hewlett-Packard

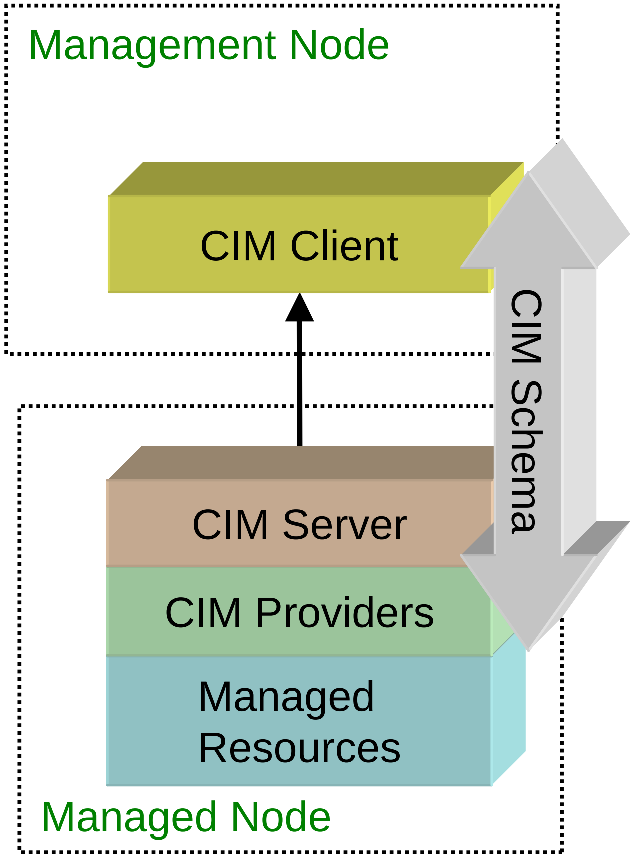
Module Content

OpenPegasus Architecture Overview

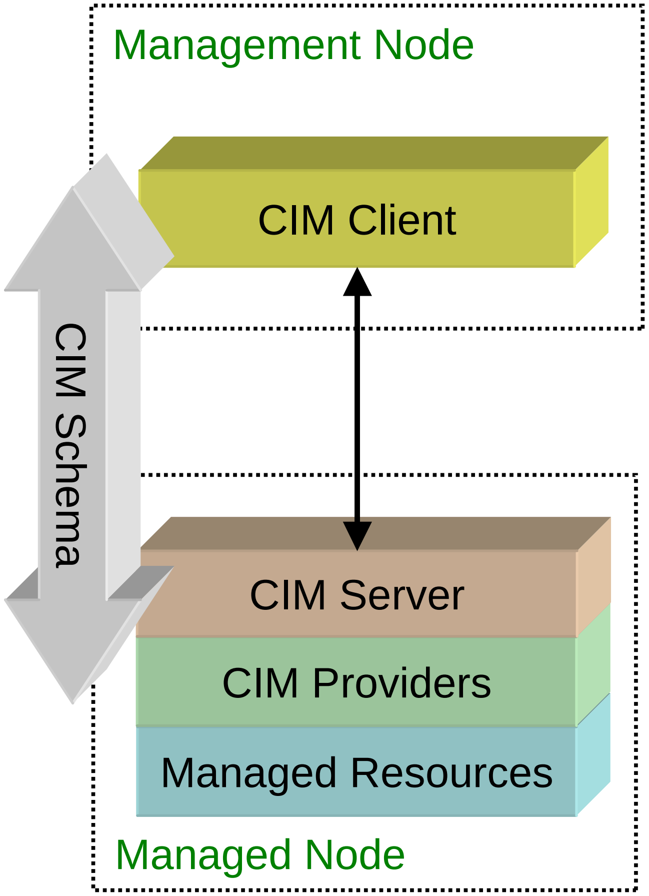
- **CIM Client**
- CIM Provider
- Repository

Architecture Components

- A **CIM Client** issues CIM Operation requests and receives and processes CIM Operation responses.
- A **CIM Server** receives and processes CIM Operation requests and issues CIM Operation responses.
- A **CIM Provider** is responsible for the actual processing of CIM Operations on one or more managed resources. It provides the mapping between the CIM interface and a resource-specific interface.
- A **Managed Resource** is a manageable entity (e.g., memory, process, system, application, network) plus the resource-specific instrumentation capable of monitoring and controlling the resource.

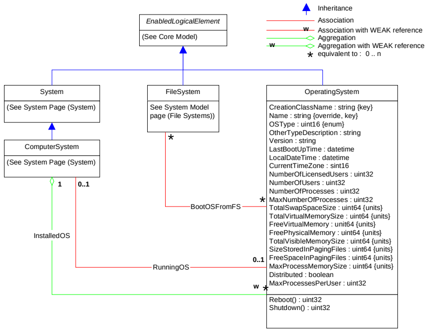


CIM Client



A **CIM Client** issues CIM Operation requests and receives and processes CIM Operation responses.

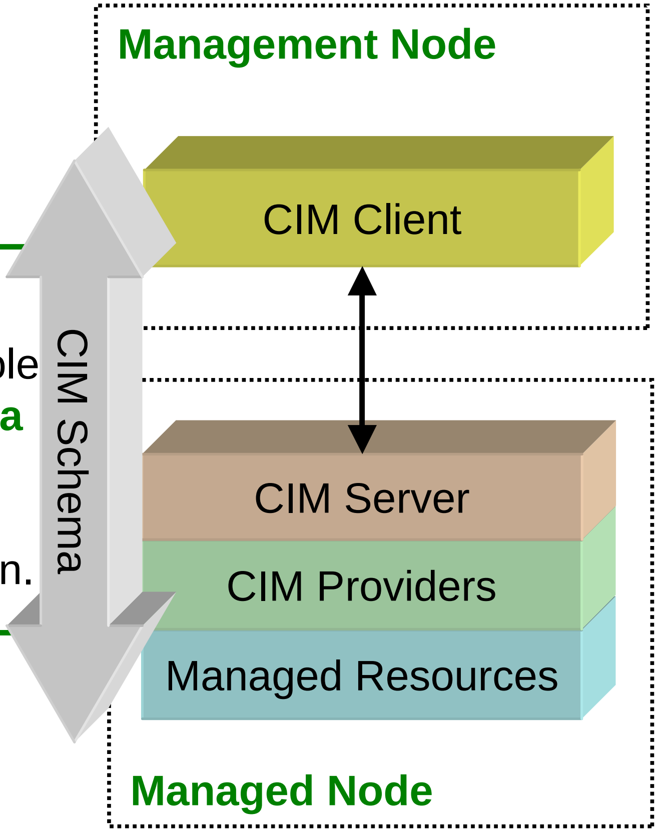
What is the value of CIM_OperatingSystem. NumberOfProcesses?


Inheritance
Association
w
Association with WEAK reference
Aggregation
w
Aggregation with WEAK reference
* equivalent to : 0 .. n
EnabledLogicalElement
(See Core Model)
System
(See System Page (System)
FileSystem
See System Model page (File Systems))
OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32
ComputerSystem
(See System Page (System)
1
0..1
*
BootOSFromFS
*
InstalledOS
RunningOS
0..1
w
*

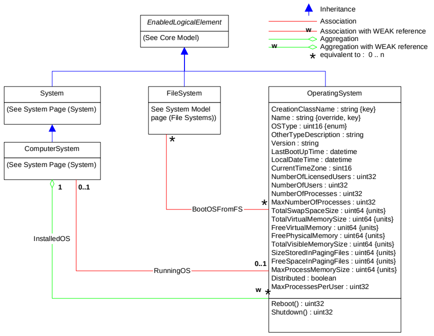
A **CIM Server** receives and processes CIM Operation requests and issues CIM Operation responses.

Common Information Model

It's important not to underestimate the role of **CIM**, both **Schema** and **Operations**, in creating an interoperable solution.



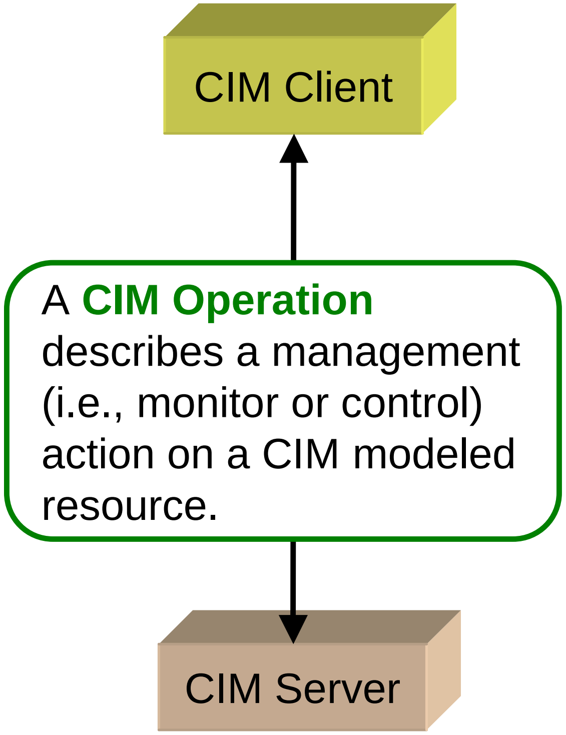
Key Fact: Communication between the CIM Server and CIM Clients and between the CIM Server and CIM Providers is based on CIM.


Inheritance
Association
Association with WEAK reference
Aggregation
Aggregation with WEAK reference
* equivalent to : 0 .. n
EnabledLogicalElement
(See Core Model)
System
(See System Page (System)
ComputerSystem
(See System Page (System)
FileSystem
See System Model page (File Systems))
OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32
BootOSFromFS
RunningOS
InstalledOS
1
0..1
0..1
w
*
*
*

CIM Operations

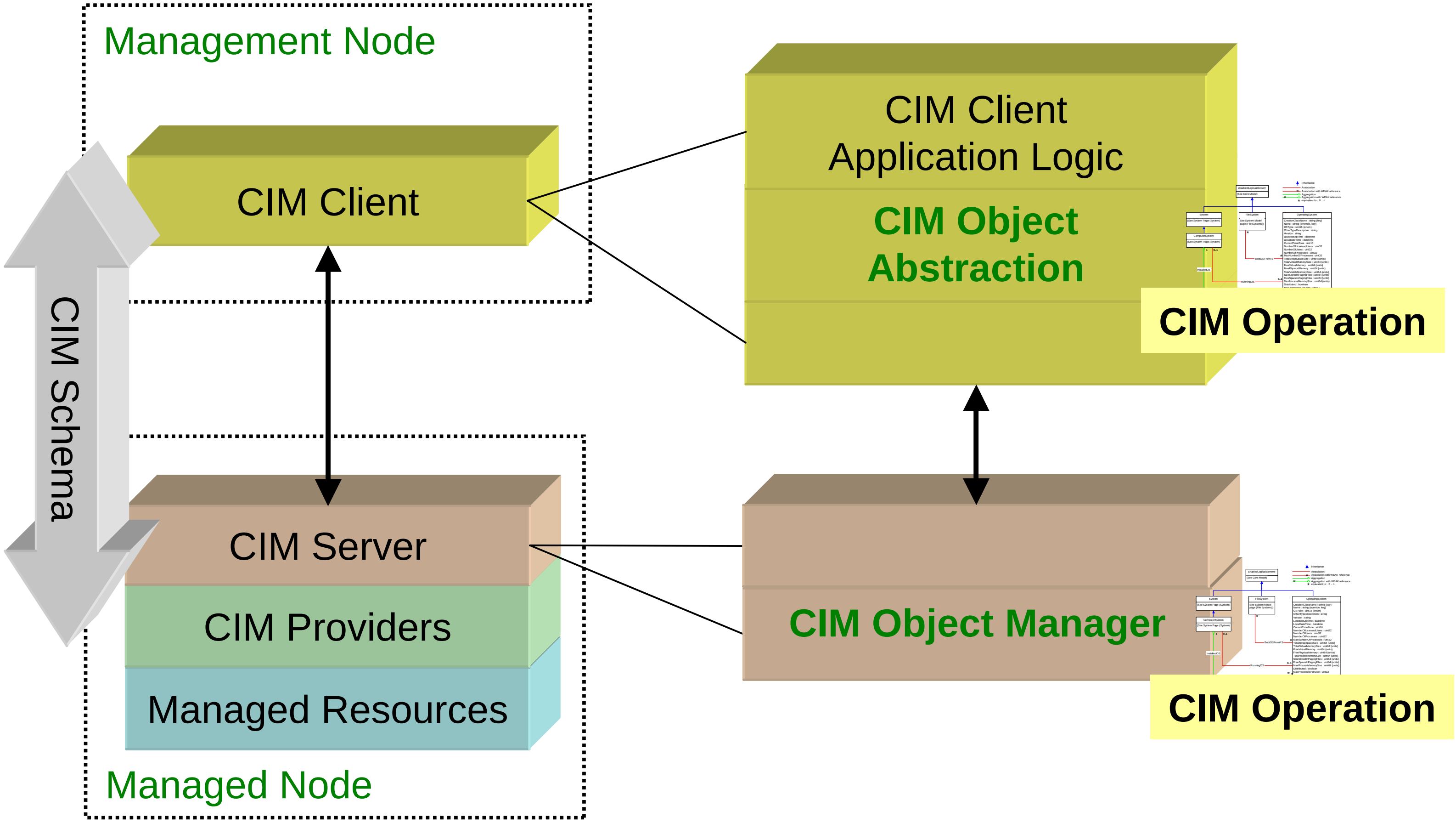
The DMTF has defined a set of CIM Operations.

Functional Group	CIM Operations
Basic read	GetClass, EnumerateClasses, EnumerateClassNames, GetInstance, EnumerateInstances, EnumerateInstanceNames, GetProperty
Basic Write	SetProperty
Schema Manipulation	CreateClass, ModifyClass, DeleteClass
Instance Manipulation	CreateInstance, ModifyInstance, DeleteInstance
Association Traversal	Associators, AssociatorNames, References, ReferenceNames
Query	ExecQuery
Qualifier Declaration	GetQualifier, SetQualifier, DeleteQualifier, EnumerateQualifier

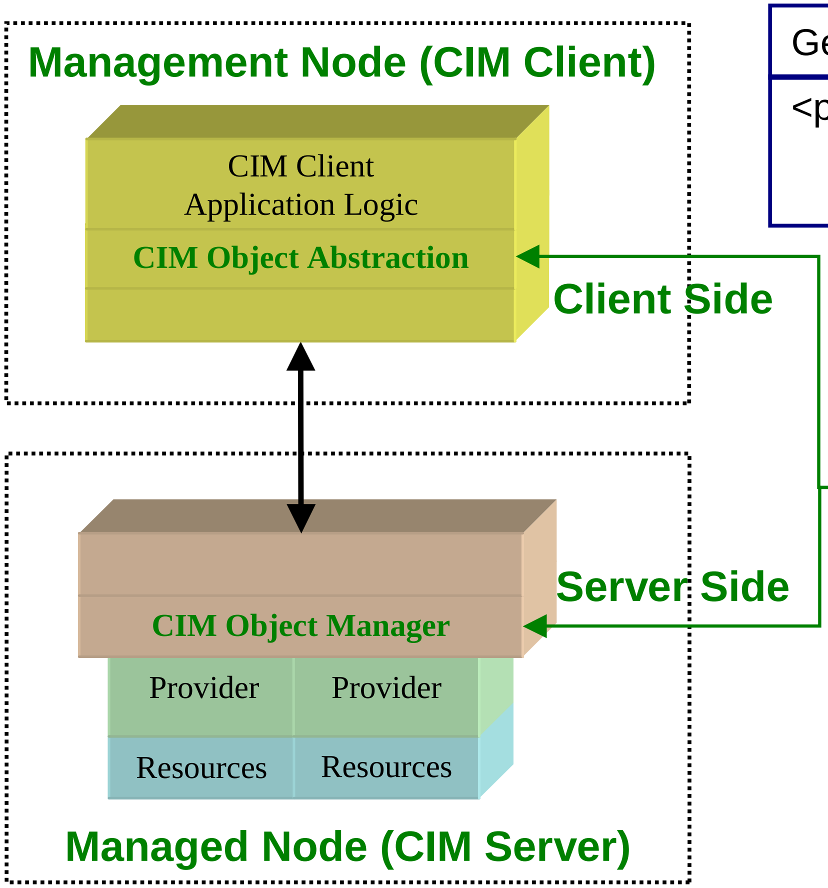


A **CIM Operation** describes a management (i.e., monitor or control) action on a CIM modeled resource.

CIM Abstraction Layer



CIM Abstraction Layer



CIM Operation

GetProperty

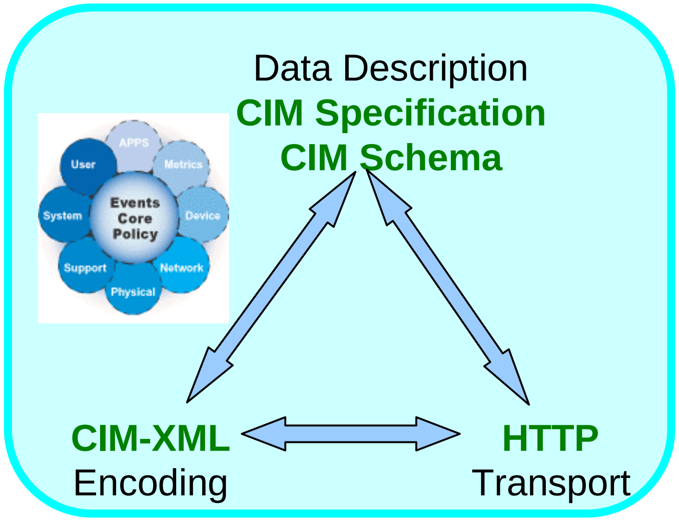
<propertyValue> **GetProperty** (
[IN] <instanceName> InstanceName,
[IN] <string> PropertyName)

The **CIM Object Abstraction Layer** is responsible for the implementation of the "protocol independent" semantics of CIM objects and operations.

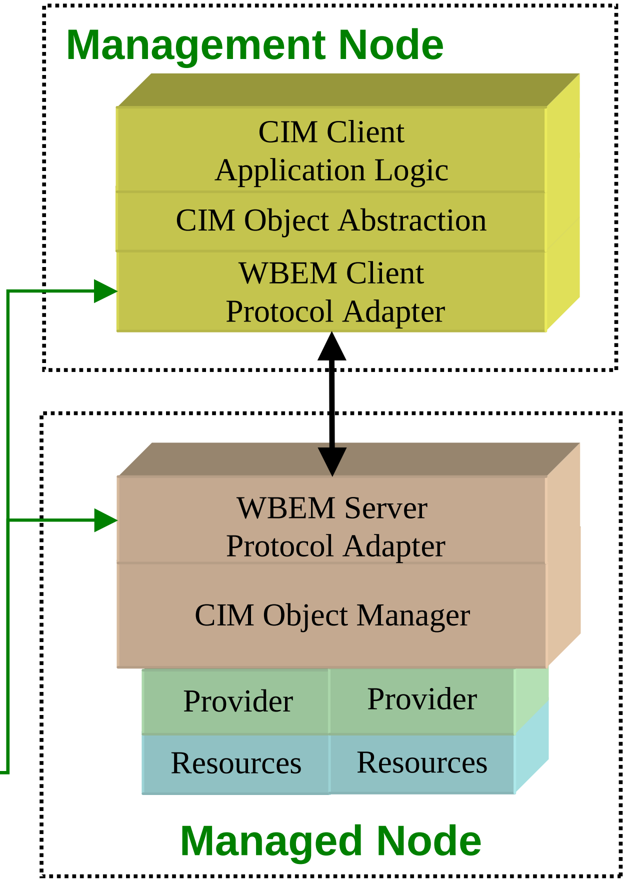
THE Open GROUP
OpenPegasus Developer Conference

hp invent

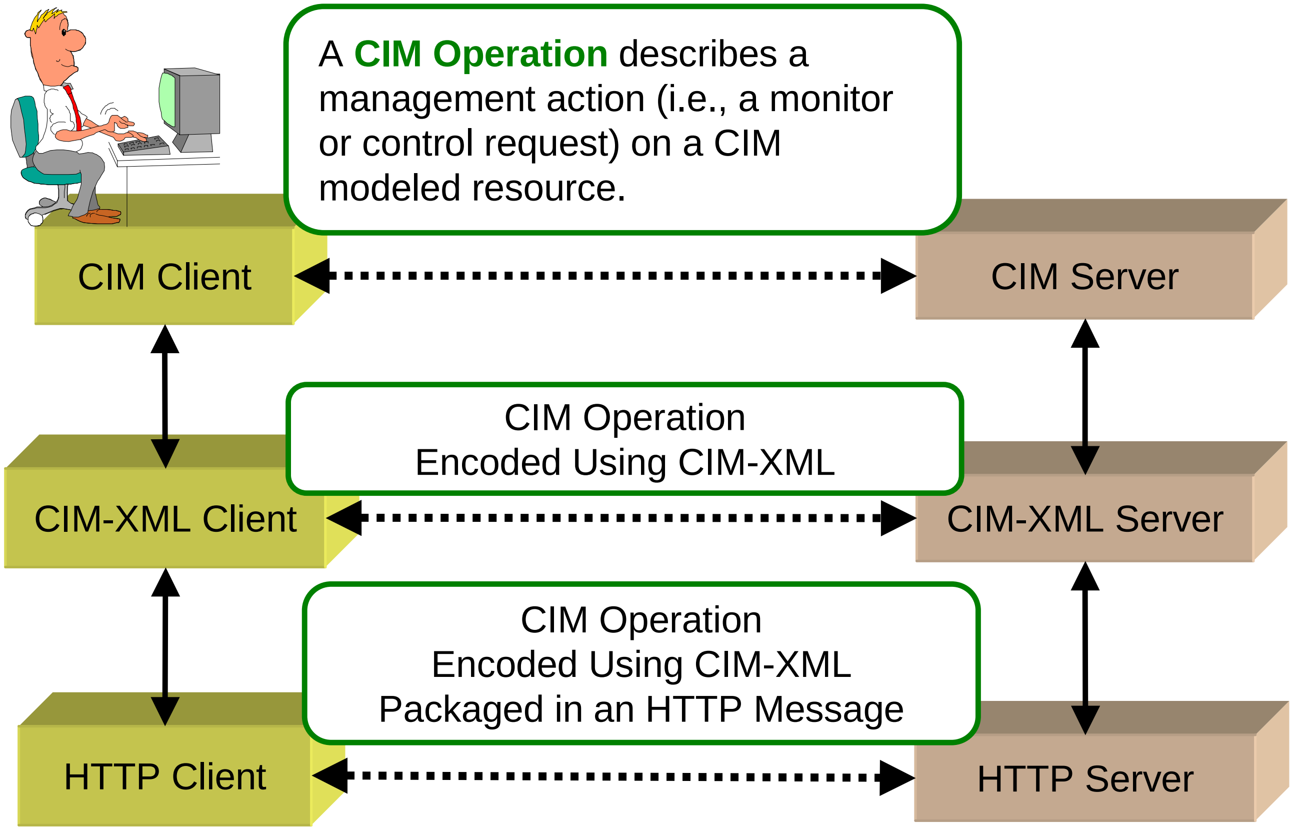
Communication Protocol



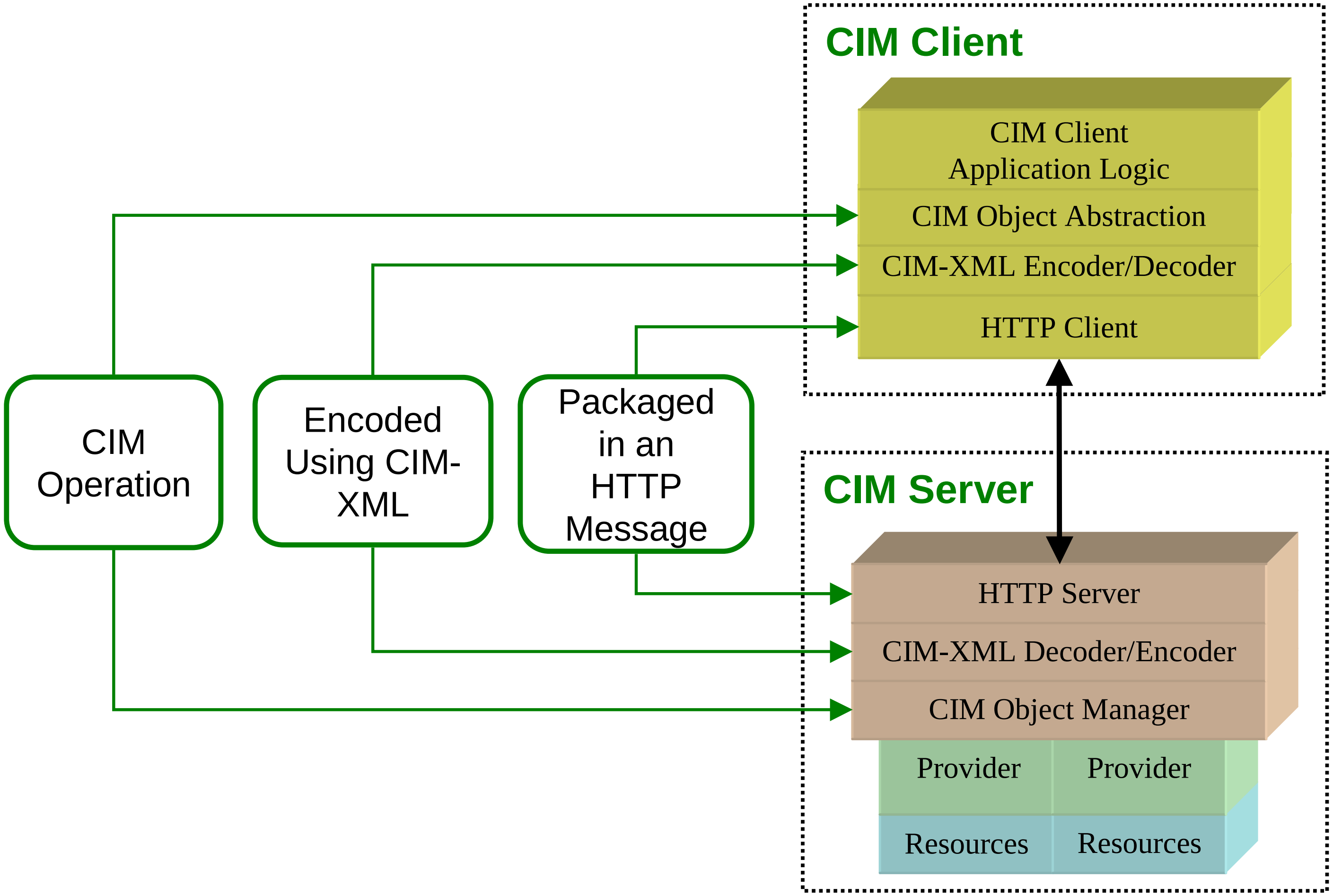
A **Protocol Adapter** is responsible for the implementation of the encoding and transport components of a WBEM standard protocol.



CIM-XML Protocol Adapter


A **CIM Operation** describes a management action (i.e., a monitor or control request) on a CIM modeled resource.
CIM Client
CIM Server
CIM Operation
Encoded Using CIM-XML
CIM-XML Client
CIM-XML Server
CIM Operation
Encoded Using CIM-XML
Packaged in an HTTP Message
HTTP Client
HTTP Server

CIM-XML Protocol Adapter



Module Content

OpenPegasus Architecture Overview

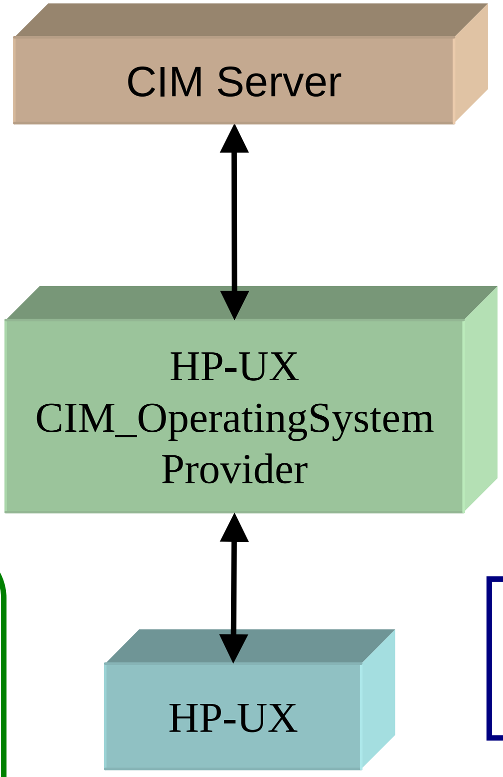
- CIM Client
- **CIM Provider**
- Repository

CIM Providers

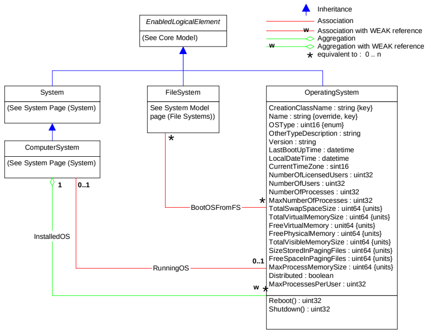
A **CIM Server** receives CIM Operation requests, coordinates the processing of requests and responses among the Providers and sends CIM Operation responses back to the CIM Client.

A **CIM Provider** translates CIM formatted requests into resource specific operations and translates resource-specific responses into CIM formatted responses.

A **Managed Resource** is a manageable entity (e.g., memory, process, system, application, network) plus the resource-specific instrumentation capable of monitoring and controlling the resource.



What is the value of CIM_OperatingSystem. NumberOfProcesses?

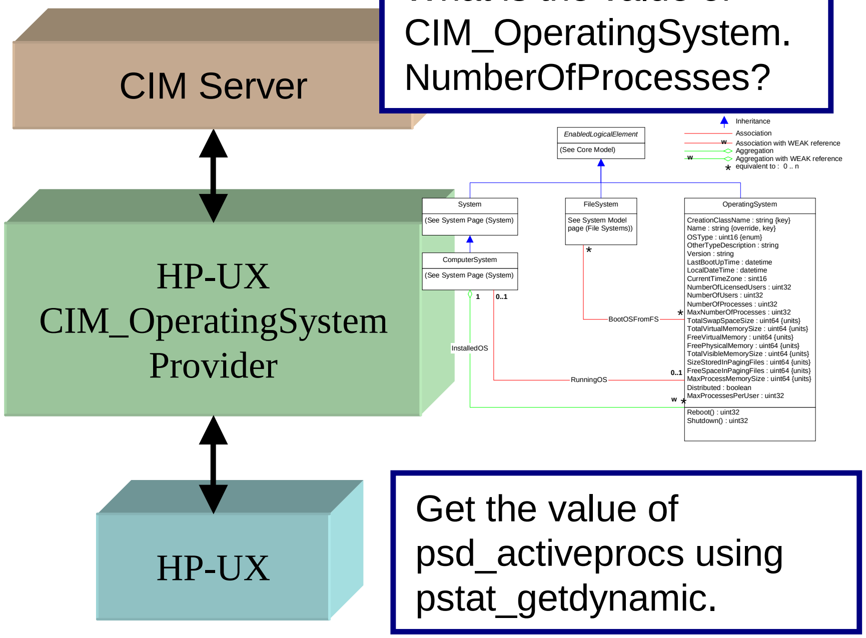

EnabledLogicalElement
(See Core Model)
Inheritance
Association
Association with WEAK reference
Aggregation
Aggregation with WEAK reference
* equivalent to : 0 .. n
System
(See System Page (System)
FileSystem
See System Model page (File Systems))
OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32
ComputerSystem
(See System Page (System)
1
0..1
*
BootOSFromFS
InstalledOS
RunningOS
0..1
w
*

Get the value of psd_activeprocs using pstat_getdynamic.

CIM Providers

Essential Fact: In order for the CIM Server to process a request on a managed resource, there MUST be a Provider capable of handling the operation.

What is the value of CIM_OperatingSystem. NumberOfProcesses?

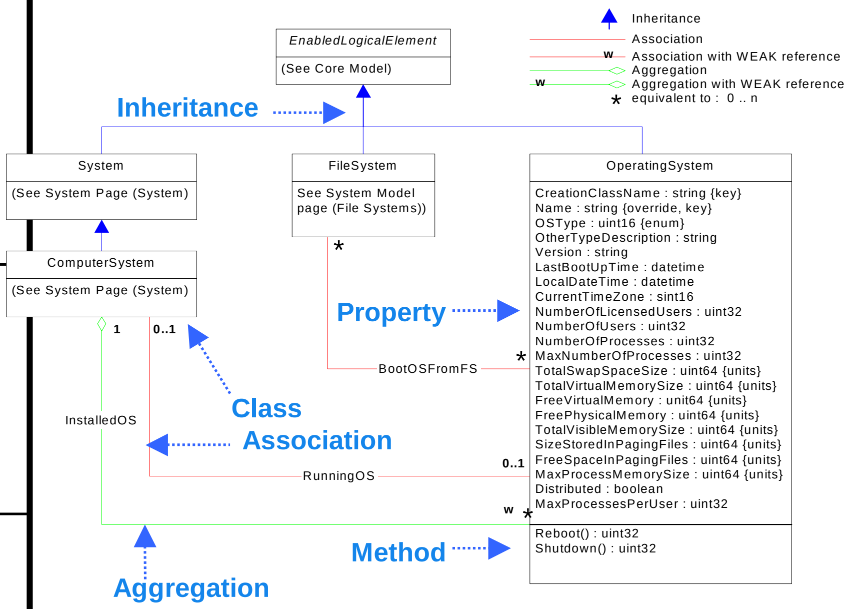

CIM Server
HP-UX CIM_OperatingSystem Provider
HP-UX
Inheritance
Association
Association with WEAK reference
Aggregation
Aggregation with WEAK reference
w
w
* equivalent to : 0 .. n
EnabledLogicalElement
(See Core Model)
System
(See System Page (System)
ComputerSystem
(See System Page (System)
FileSystem
See System Model page (File Systems))
OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32
*
BootOSFromFS
*
1
0..1
InstalledOS
RunningOS
0..1
w
*

Get the value of psd_activeprocs using pstat_getdynamic.

The CIM Server will return an error (e.g., CIM_ERR_NOT_SUPPORTED) if there is no Provider capable of handling the request.

CIM Provider Types

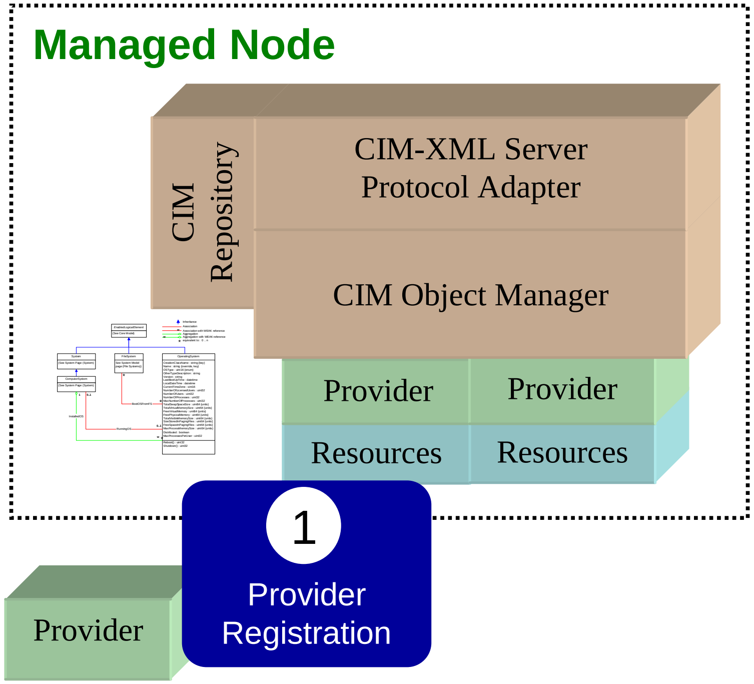
CIM Operation	Implementation Owner[1]
GetClass, CreateClass, ModifyClass, DeleteClass, GetQualifier, SetQualifier, DeleteQualifier, EnumerateQualifier, EnumerateClasses, EnumerateClassNames,	**CIMOM**
GetInstance, EnumerateInstances, EnumerateInstanceNames, CreateInstance, ModifyInstance, DeleteInstance	**Instance Provider**
InvokeMethod	**Method Provider**
References, ReferenceNames, Associators, AssociatorNames	**Association Providers**


Inheritance
Association
Association with WEAK reference
Aggregation
Aggregation with WEAK reference
equivalent to : 0 .. n
EnabledLogicalElement
(See Core Model)
Inheritance
System
(See System Page (System)
FileSystem
See System Model page (File Systems))
OperatingSystem
CreationClassName : string {key}
Name : string {override, key}
OSType : uint16 {enum}
OtherTypeDescription : string
Version : string
LastBootUpTime : datetime
LocalDateTime : datetime
CurrentTimeZone : sint16
NumberOfLicensedUsers : uint32
NumberOfUsers : uint32
NumberOfProcesses : uint32
MaxNumberOfProcesses : uint32
TotalSwapSpaceSize : uint64 {units}
TotalVirtualMemorySize : uint64 {units}
FreeVirtualMemory : unit64 {units}
FreePhysicalMemory : uint64 {units}
TotalVisibleMemorySize : uint64 {units}
SizeStoredInPagingFiles : uint64 {units}
FreeSpaceInPagingFiles : uint64 {units}
MaxProcessMemorySize : uint64 {units}
Distributed : boolean
MaxProcessesPerUser : uint32
Reboot() : uint32
Shutdown() : uint32
ComputerSystem
(See System Page (System)
Property
BootOSFromFS
1
0..1
Class
Association
InstalledOS
RunningOS
0..1
w
Method
Aggregation

[1] Details in this column vary by implementation.

Provider Registration

Key Fact: CIM Provider Modules are implemented as shared libraries.

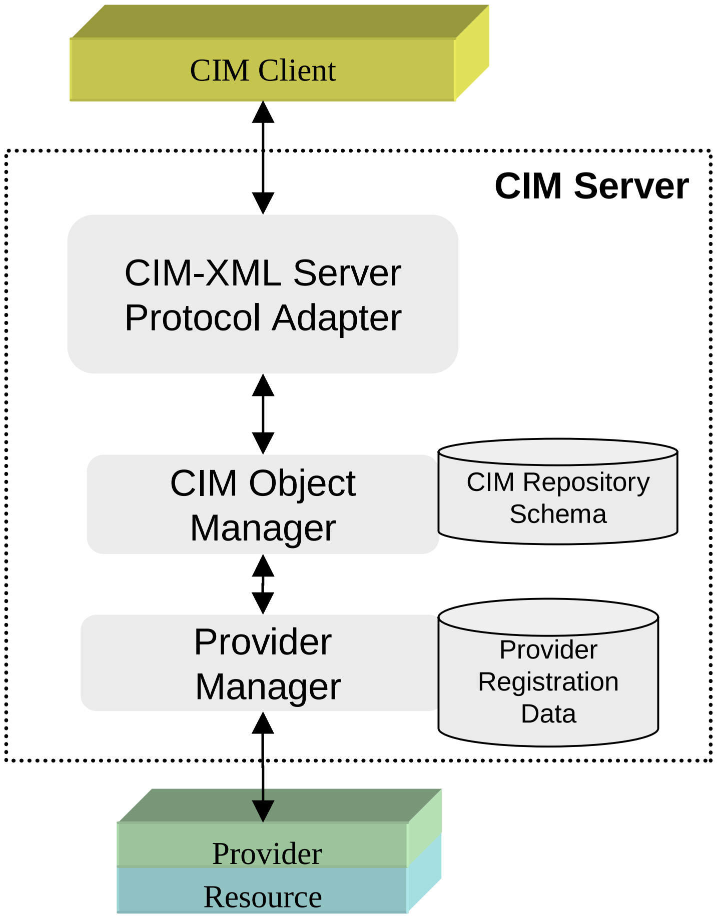


Registering with the CIM Server is a key part of Provider installation.

Provider registration includes:

- Defining the Schema (e.g., classes and properties) supported by the Providers in this Module.
- Informing the CIM Server of the supported capabilities
- Placing links to the Provider Module binary (i.e., shared library) in the /opt/wbem/providers/lib directory.

Provider Manager



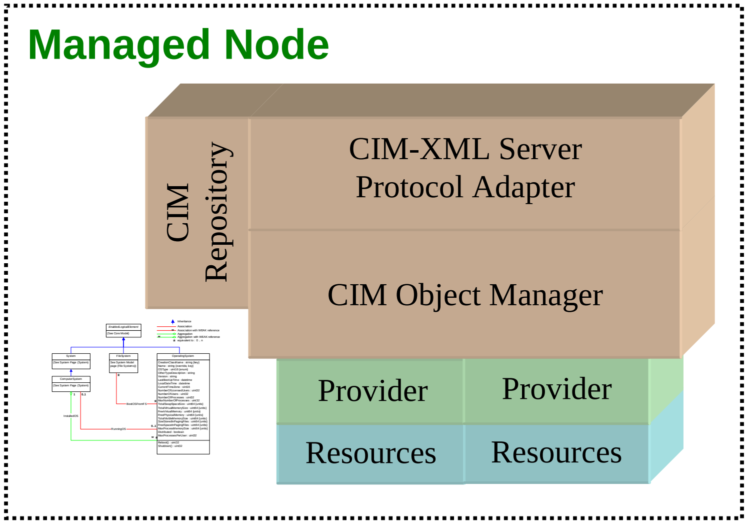
The **Provider Manager** is responsible for maintaining the Provider registration, Provider loading and unloading, and the routing of requests and responses between the CIM Object Manager and the CIM Providers.

Module Content

OpenPegasus Architecture Overview

- CIM Client
- CIM Provider
- **Repository**

CIM Repository



The **CIM Repository** is a persistent store managed by the CIM Server. It contains the definition of Schema grouped by Namespace.

In some implementations, the CIM Repository may also contain instance data. However, as a general rule, management data (i.e., the information contained in CIM Instances) is owned and managed by the resource-specific management instrumentation.

Pre-Defined Namespaces

root/PG_InterOp
INTEROP NAMESPACE

- The root/PG_InterOp namespace is owned by the CIM Server. Schema modification operations are restricted to CIM Server.
- Instance data is accessible using standard CIM operations (e.g., loading an instance of the provider registration schema)

root/cimv2
MANAGED SYSTEM NAMESPACE

- This namespace is “owned” by the managed system vendor.

root
ROOT NAMESPACE

- The root namespace is empty.
- It exists to enable interoperability with older CIM Client implementations

root/PG_Internal
PEGASUSINTERNALNAMESPACE

•This namespace is “for internal use only” and subject to change