DEN-Based Object Modeling and Schema for Policy-Based Networking

Session 3010

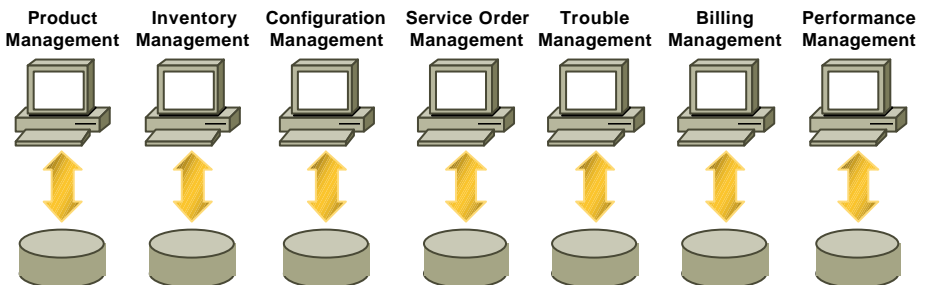## Policy Networking Scenario

- **Network configuration/management**

  Financial institution must cope with different operational scenarios

    Normal operation, high-volume, and emergency meltdown policies

    Configure the network, not just individual isolated device interfaces

    Map business rules and procedures to applications that use the network

Shortcomings of Today's Network Operations

Product Management	Inventory Management	Configuration Management	Service Order Management	Trouble Management	Billing Management	Performance Management

- **Architectural issues**

  Data redundancy

  Synchronization problems

  Application authorization issues

  Vendor and Application "lock in"

- **Integration issues**

  Isolated data silos

  Administrative nightmare

  Integration/customization nightmare

  Transition from legacy systems to a more flexible new operational architecture

Cisco's Internet OSS

Legacy Systems		ISV Apps and Systems

Partner Products

Performance and SLA Reporting	Order Entry and Workflow Mgmt	Customer Care and Self-Service	OSS Gateway	Billing and Mediation $$$

OPEN APIs ... **OPEN Object Models**

Cisco Products

Developer Toolkits	OEM Partner Extensions

Cisco Internal APIs

Cisco-Specific DEN Extensions

DEN Information Model

Application-Specific Bindings

OPEN APIs **Cisco APIs** ... **OPEN Object Models**

Intelligent Agents

Service Assurance	NetFlow Accounting	Generic Event Client	Expression MIB	Java Client	CORBA Client	LDAP Client

3010
1170_05_2000_c2 © 2000, Cisco Systems, Inc.

cisco.com

5

Agenda

- **Object-Oriented Modeling**

- **CIM and DEN**

- **Cisco's Extensions to DEN**

- **Policy Info Model and LDAP Realization**

- **Use of DEN in Cisco Products**

- **Futures for DEN and PBM**

3010
1170_05_2000_c2 © 2000, Cisco Systems, Inc.

cisco.com

6

Agenda

- **Object-Oriented Modeling**

  Terminology

  CIM, DEN, and Mapping

  Basics of OO Modeling

  Examples

Definition of a Data Model

"

A concrete representation of the characteristics of a set of objects in terms appropriate to a specific data storage and access technology

"

Definition of a Schema

"

A set of data models that describe a set of objects to be managed

"

Definition of an Information Model

"

A technology-independent specification of the characteristics of a set of objects, and their relationships to other objects in a managed environment, with no reference to either storage methods, access protocols, or technologies

"

Agenda

- **Object-Oriented Modeling**

  Terminology

  CIM, DEN, and Mapping

  Basics of OO Modeling

  Examples

Definition of CIM, the Common Information Model

"

CIM is an object-oriented information model that describes how a system and its components may be managed

"

Definition of DEN

" DEN is two things:

1) An extension of CIM

2) A mapping of information to a format that can be stored in a directory that uses (L)DAP as its access protocol "

DEN Mappings

- **Information model (repository-independent)**

  1:n

- **Data model (repository-specific)**

  1:n

- **Vendor implementation (vendor-specific)**

Directory-Enabled Networking

- **Confusion between the terms "DEN" and "directory-enabled networking"**

   DEN is a specification that defines an information model and schemata that serve as the foundation for policy-based network management

   The term directory-enabled networking is a design philosophy that enables applications to use directories to store information to take advantage of the network

- **End-goal is to manage the binding of network services to clients of the network**

   Allocates network resources based on business rules and network conditions

   Can be used just as effectively for Inventory applications as it can for QoS, Voice, and other "active" applications

---

Agenda

- **Object Oriented Modeling**

   Terminology

   CIM, DEN, and Mapping

   Basics of OO Modeling

   Examples

What Is Object-Oriented Modeling?

" **Object-oriented modeling is a formal way of representing systems and components of a system using object-oriented concepts, it draws from traditional classification theory** "

OO Modeling Approach

- **Models the world in terms of objects**

  An object is an abstraction of something that consists of a set of related data and behavior

  An object is treated by the system as a named entity that has a set of characteristics (properties and methods), behavior, and a unique identity

- **Objects are described by hierarchies**

  Hierarchies emphasize common attributes and behavior of a group of objects

  Inheritance hierarchies generalize/specialize these

  Aggregation hierarchies relate objects to each other

Associations and Aggregations

- **An association is a class representing the relationship between two or more objects**

- **An aggregation is a strong form of association denoting whole-part:**

  Transitivity (a -> b, b -> c, then a -> c)

  Anti-symmetricity (if a is part of b, then b is not part of a)

  Assembly properties propagate to its constituent components

Multiplicity

- **Multiplicity specifies the number of instances on each side of a relationship**

  DEN usually limits this to 0, 1, 0..1, or n

  However, this can in general be a range, a set of specific values, or even determined by a function

  This is an area that will be specialized by vendor-specific implementations

Associations Are Classes!

- **Associations and aggregations are classes in their own right**

    They can have properties and methods

    They can inherit properties and behavior

    They can respond to events (future release of CIM)

Hierarchy Differences

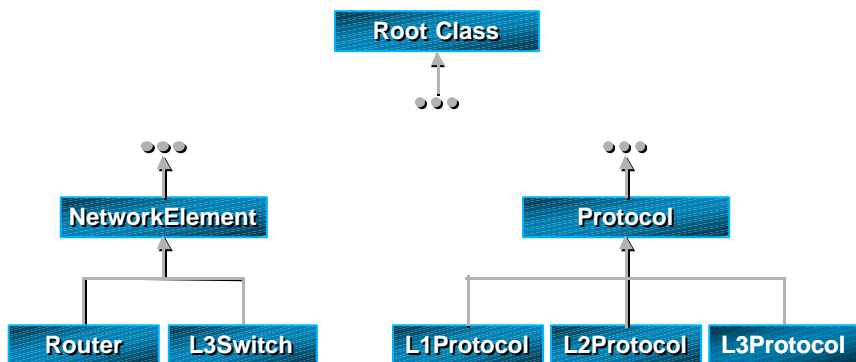- **Inheritance and association hierarchies are NOT the same**

    Inheritance hierarchies specify the refinement of the functionality and behavior of individual objects

    Association and aggregation hierarchies specify the refinement of functionality of relationships between objects

        Each aggregate serves as the root of its own class hierarchy

        Each aggregate has a different composition

Multiple Inheritance Hierarchies

Root Class

NetworkElement

Protocol

Router   L3Switch

L1Protocol   L2Protocol   L3Protocol

Implied Aggregation Hierarchies

Using Generalization

NetworkElement   NetworkElement   NetworkElement

L1Protocol   L2Protocol   L3Protocol

Using Specialization

Router   Router   Router   L3Switch   L3Switch   L3Switch

L1Protocol   L2Protocol   L3Protocol   L1Protocol   L2Protocol   L3Protocol
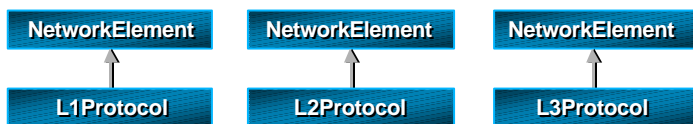
Conceptual Inheritance Hierarchy of a Switch Port
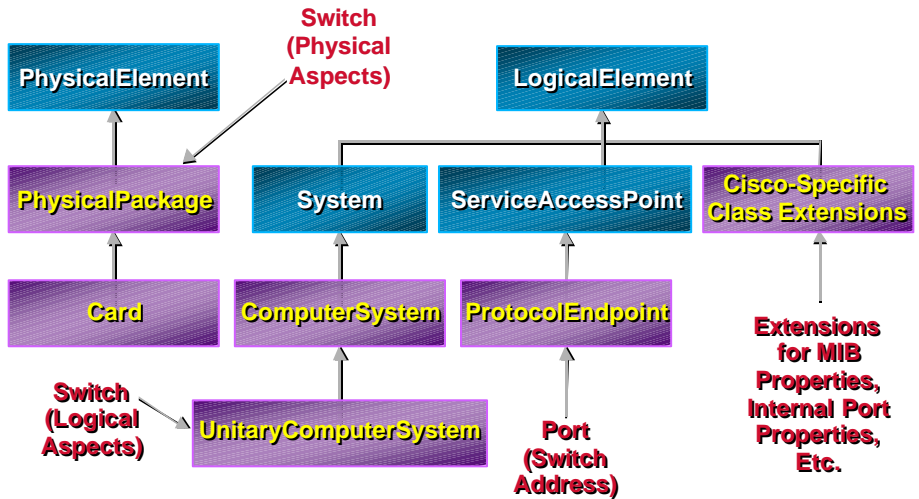
Switch (Physical Aspects)

PhysicalElement

LogicalElement

PhysicalPackage

System

ServiceAccessPoint

Cisco-Specific Class Extensions

Card

ComputerSystem

ProtocolEndpoint

Extensions for MIB Properties, Internal Port Properties, Etc.

Switch (Logical Aspects)

UnitaryComputerSystem

Port (Switch Address)

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

25

Agenda

- **Object-Oriented Modeling**

   Terminology

   CIM, DEN, and Mapping

   Basics of OO Modeling

   Examples

3010
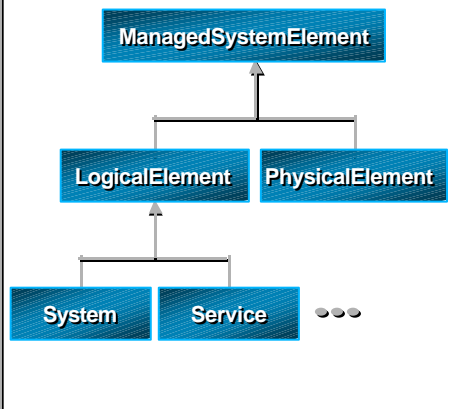1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

26

Modeling Example

- **Let's trace the development of part of the Physical Schema to see how one might design a relatively simple model**

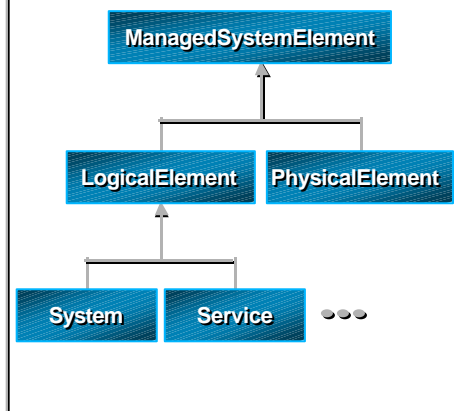# Physical Schema—1

Existing Class Hierarchy

ManagedSystemElement

LogicalElement   **PhysicalElement**

System   **Service**   ●●●

New Classes

PhysicalFrame

Chip

Card

Chassis

Rack

Physical Schema—2

Existing Class Hierarchy

```
            ManagedSystemElement
                    ▲
         ┌──────────┴──────────┐
   LogicalElement         PhysicalElement
         ▲
    ┌────┴────┐
  System   Service   ● ● ●
```

New Classes

```
            PhysicalFrame
                 ▲
         ┌───────┴───────┐
      Chassis          Rack

       Chip             Card

          PhysicalConnector
```
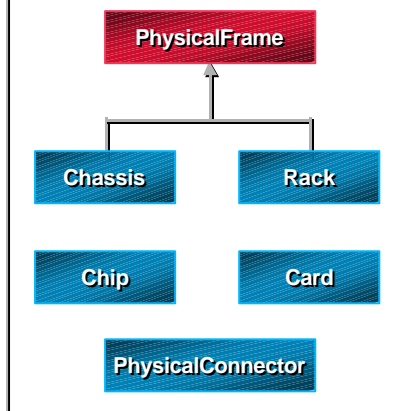
Physical Schema—3

New Class Hierarchy

```
                    ManagedSystemElement
                            ▲
            ┌───────────────┴───────────────┐
      LogicalElement                   PhysicalElement
            ▲                                ▲
       ┌────┴────┐                    ┌───────┴────────┐
   System    Service          PhysicalPackage    PhysicalConnector
                                     ▲
                               ┌─────┴─────┐
                         PhysicalFrame   Card
```
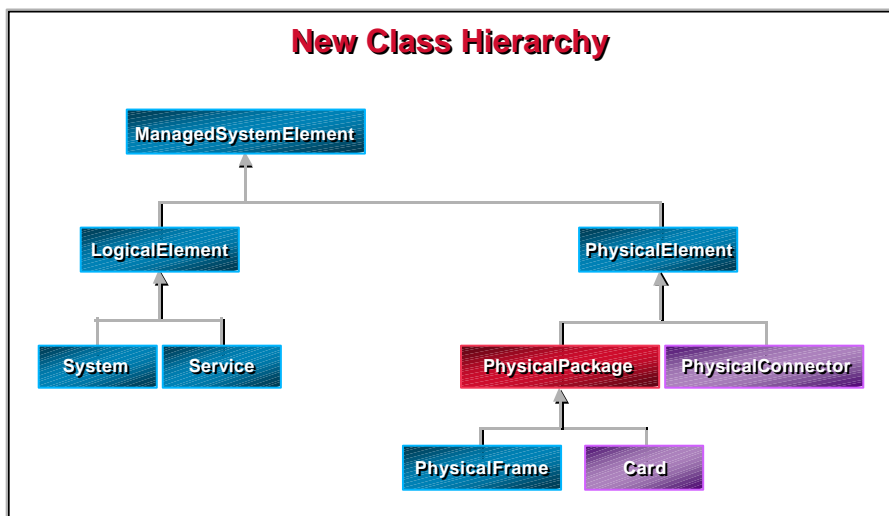
Physical Schema—4

New Class Hierarchy

```
                    ManagedSystemElement
                    /                  \
          LogicalElement              PhysicalElement
          /        \                 /       |       \
    System      Service    PhysicalComponent  PhysicalPackage  PhysicalConnector
                                |              /        \              |
                              Chip      PhysicalFrame   Card         Slot
                                        /        \
                                  Chassis        Rack
```

Using the Building Blocks of DEN

Physical Realization

Information Model

Logical Representation

Let's See How Physical Chassis and Racks Can Be Represented Using DEN

Using the Building Blocks of DEN

Physical Realization

Information Model

Logical Representation

ManagedSystemElement

PhysicalElement

PhysicalFrame

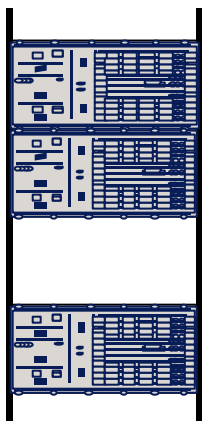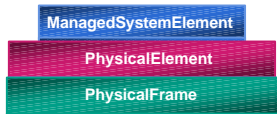**Instance**

Name: "RouterRack"
Description: "Rack used to store routers"
Caption: ""
Status: "In Use"
InstallDate: 8/19/1994 10:53 AM
CreationClassName: "Rack"
Tag: "A23498372"
Manufacturer: "Rack-Em-Up Inc."
Model: "R4-10x15"
SKU: "123291990"
SerialNumber: "REU152-3312012"
PartNumber: "REU152"
Removable: False
Replaceable: False
HotSwappable: False
Height: "8 ft"
Depth: "18 in"
Width: "18 in"
Weight: "92 lbs"
CableManagementStrategy: "RatsNest"
LockPresent: True
IsLocked: True
TypeOfRack: 17
CountryDesignation: "USA"

Rack
TypeOfRack: uint16
CountryDesignation: string

NumberOfP
CurrentReq
HeatGenera
ChassisTypes: uint16 [ ]
TypeDescriptions: string [ ]

0..1

ChassisInRack

Using the Building Blocks of DEN

Physical Realization

Information Model

Logical Representation

ManagedSystemElement

PhysicalElement

PhysicalFrame

Rack:
Name="Rack 1"

ChassisInRack

Chassis:
Name="Chas A"

Chassis:
Name="Chas B"

ChassisInRack

ChassisInRack

Chassis:
Name="Chas D"

Rack
TypeOfRack: uint16
CountryDesignation: string

Chassis
NumberOfPowerCords: uint16
CurrentRequiredOrProduced: sint16
HeatGeneration: uint16
ChassisTypes: uint16 [ ]
TypeDescriptions: string [ ]

0..1

ChassisInRack

Using the Building Blocks of DEN

Physical Realization

Information Model

Logical Representation

ManagedSystemElement

PhysicalElement

PhysicalFrame

Rack: Name="Rack 1"

ChassisInRack

Chassis: Name="Chas A"

Chassis: Name="Chas B"

Chassis: Name="Chas D"

Rack: Name="Rack 2"

ChassisInRack

Chassis: Name="Chas A"

Chassis: Name="Chas C"
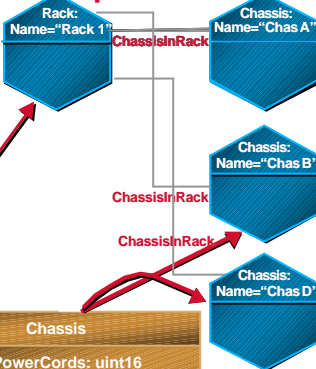
Chassis: Name="Chas D"

Rack

TypeOfRack: uint16
CountryDesignation: string

0..1

Chassis

NumberOfPowerCords: uint16
CurrentRequiredOrProduced: sint16
HeatGeneration: uint16
ChassisTypes: uint16 [ ]
TypeDescriptions: string [ ]

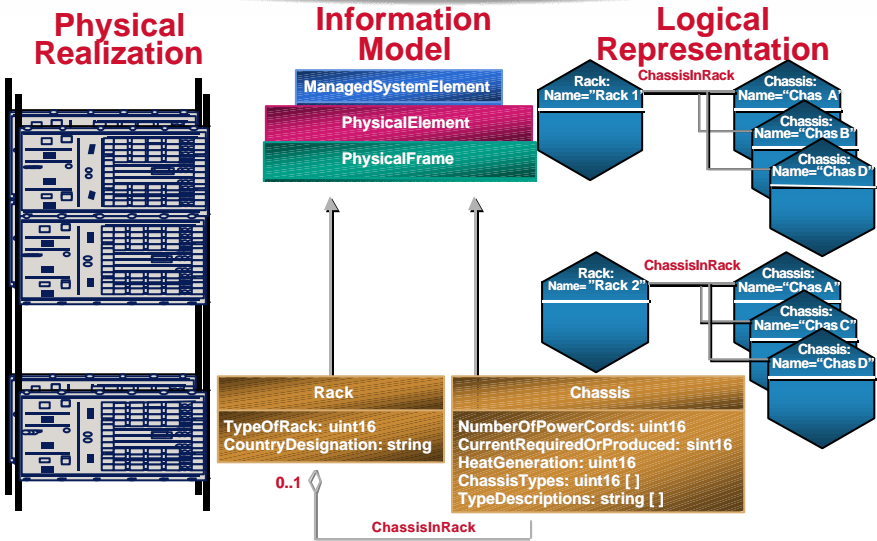ChassisInRack

---

Agenda

- **Object-Oriented Modeling**

- **CIM and DEN**

- **Cisco's Extensions to DEN**

- **Policy Info Model and LDAP Realization**

- **Use of DEN in Cisco Products**

- **Futures for DEN and PBM**

Agenda

- **CIM**

  Overview

  Core, System and Device Common Models

  Physical Model in Detail

  Network Model in Detail

- **DEN**

CIM Components

CIM Specification v2.2   **DEN LDAP Mappings**

Meta Model

System

QoS (DEN)

User   Apps

IPSec (DEN)   Policy (DEN)   Core

MOF Parser and Editor

Logical Network (DEN)   Physical (DEN)

QoS (DEN)   Device
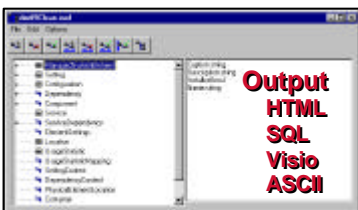
Output
HTML
SQL
Visio
ASCII
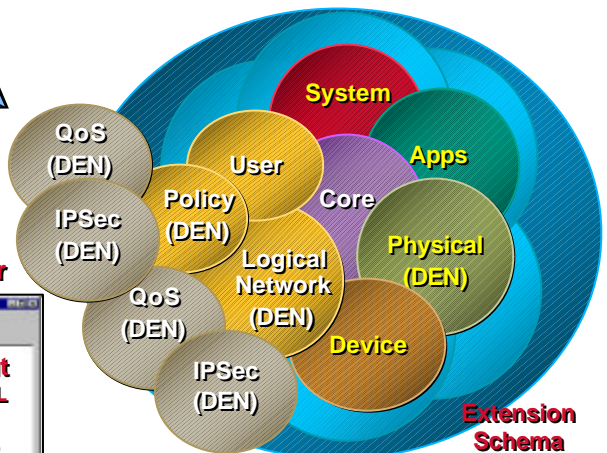
IPSec (DEN)   Extension Schema

Agenda

- **CIM**

  Overview

  Core, System and Device Common Models

  Physical Model in Detail

  Network Model in Detail

- **DEN**

Core and Common Models

- **Core Model contains classes and associations applicable to all management domains**

- **Common models address specific domains—systems, networks, …**

  Subclassed from Core

  Models overlap and cross reference each other

Overview of the Core Model
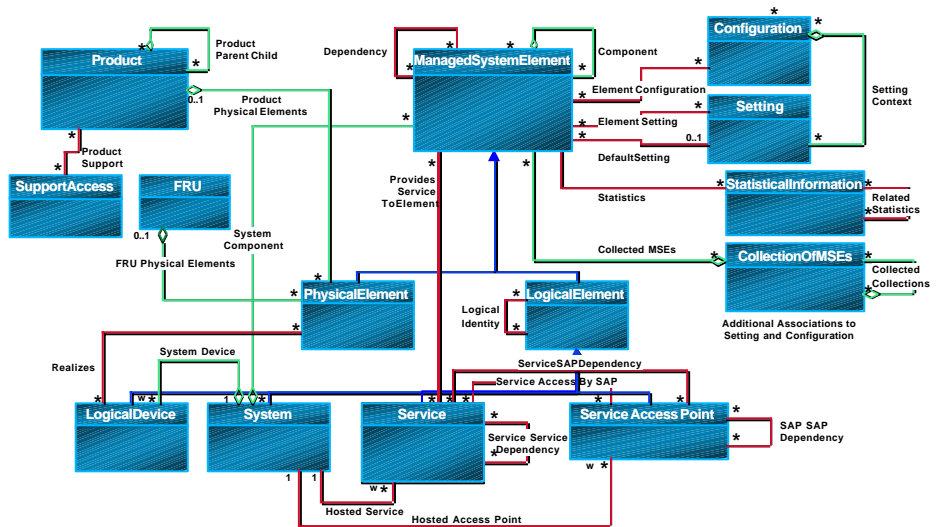
Core Model—Service and SAP

- **Service and ServiceAccessPoint**

   Service is a general-purpose object to configure and manage "functionality" (provided by a Device and/or SoftwareFeature)

   ServiceAccessPoints manage the utilization or invocation of a Service

   Both are "hosted" on Systems

ManagedSystemElement

LogicalElement

Service SAP Dependency

Service Access By SAP

Service

System Creation Class Name: string [key] †
System Name: string [key] †
Creation Class Name: string [key]
Name: string [key]
StartMode: string
Started: boolean

Start Service(): uint32
Stop Service(): uint32

ServiceAccessPoint

System Creation Class Name: string [key] †
System Name: string [key] †
Creation ClassName: string [key]
Name: string [key]

System

Service Service Dependency

SAP SAP Dependency

Hosted Service

HostedAccessPoint
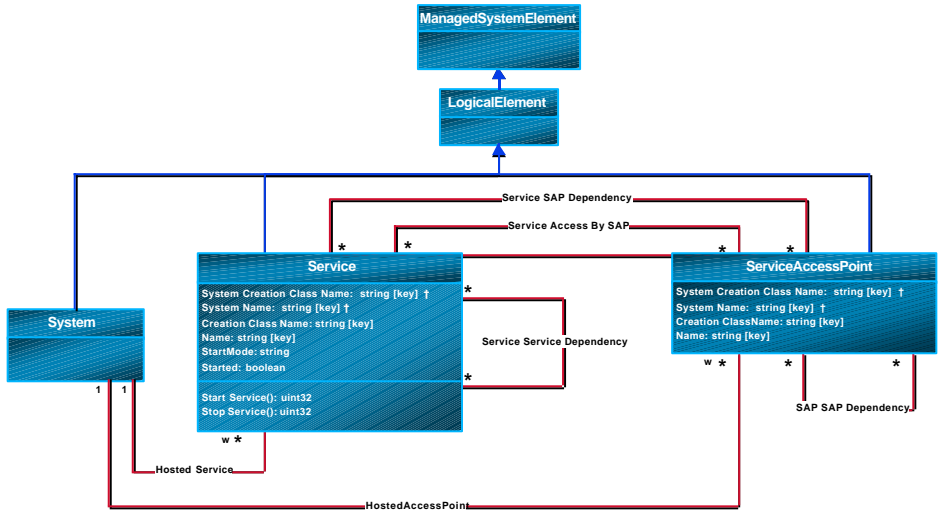
† **Indicates a Propogated Key**

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

43

Core Model—System

- ## System

  Aggregates a well-defined set of MSEs, operates as a functional whole and is viewed as a single entity

  Individual functions of each MSE are retained, but new functions are created by aggregating the MSEs (into a system)

  A top level object providing "scope" to aggregated components

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

44

System
CreationClassName: string [key]
NameFormat: string
Name: string [key]
PrimaryOwnerName: string
PrimaryOwnerContact: string
Roles: string [ ]
OtherIdentifyingInfo: string[ ]
IdentifyingDescriptions: string[ ]
Dedicated: uint16[ ]

LogicalDevice

StorageExtent

Processor

Computer System

ApplicationSystem

StorageLibrary

VirtualComputerSystem

Memory

OperatingSystem

UnitaryComputerSystem

Cluster

System Device

ComponentCS

Installed OS

Hosting CS

Running OS

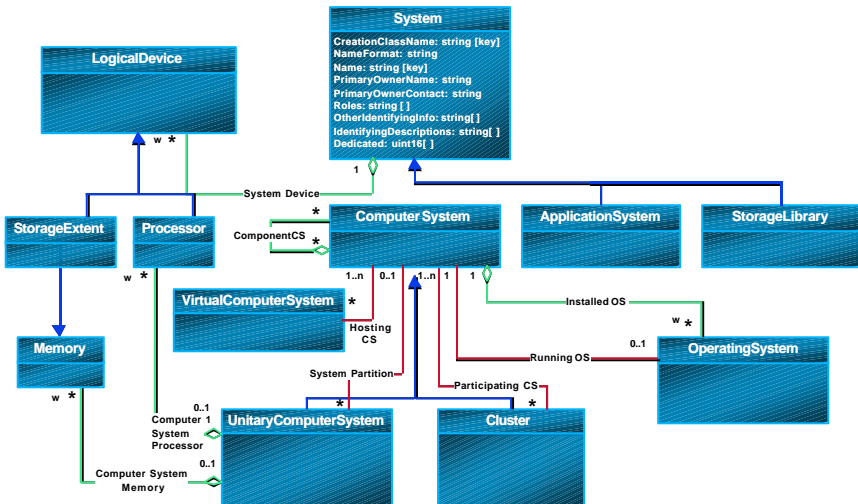System Partition

Participating CS

Computer System Processor

Computer System Memory

1..n 0..1 1..n 1 1

0..1

0..1

0..1

w *

w *

w *

w *

*

*

*

*

*

3010
1170_05_2000_c2 © 2000, Cisco Systems, Inc.

cisco.com

45

Core Model—LogicalDevice

- **LogicalDevice**

  Abstraction or emulation of the hardware aspects of a system

  Defined as "weak" to system

  Management of a device's operation and configuration

  Looking at the SystemDevice association:

  The system class has a cardinality of 1, so when a LogicalDevice is created, a scoping system and the SystemDevice association must also be instantiated

3010
1170_05_2000_c2 © 2000, Cisco Systems, Inc.

cisco.com

46

Overview of the Device Model
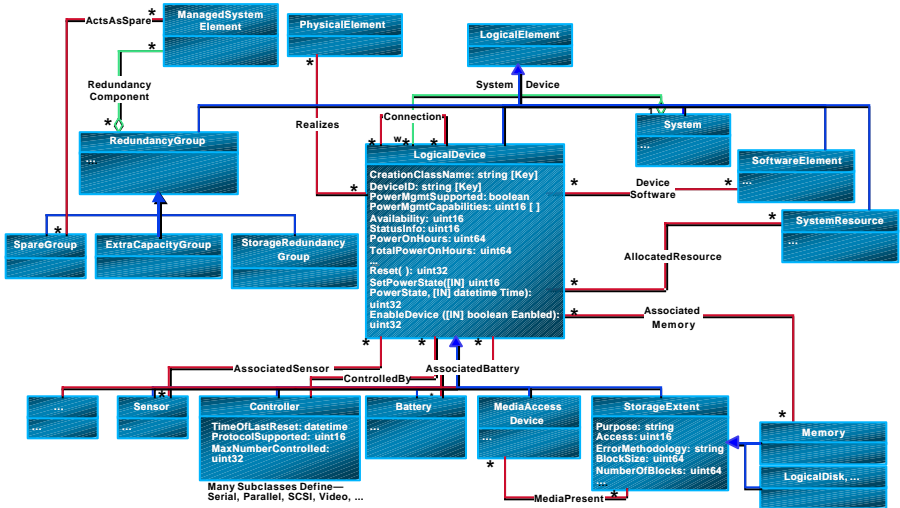
Agenda

- ## CIM

  Overview

  Core, System and Device Common Models

  Physical Model in Detail

  Network Model in Detail

- ## DEN

Overview of the Physical Model
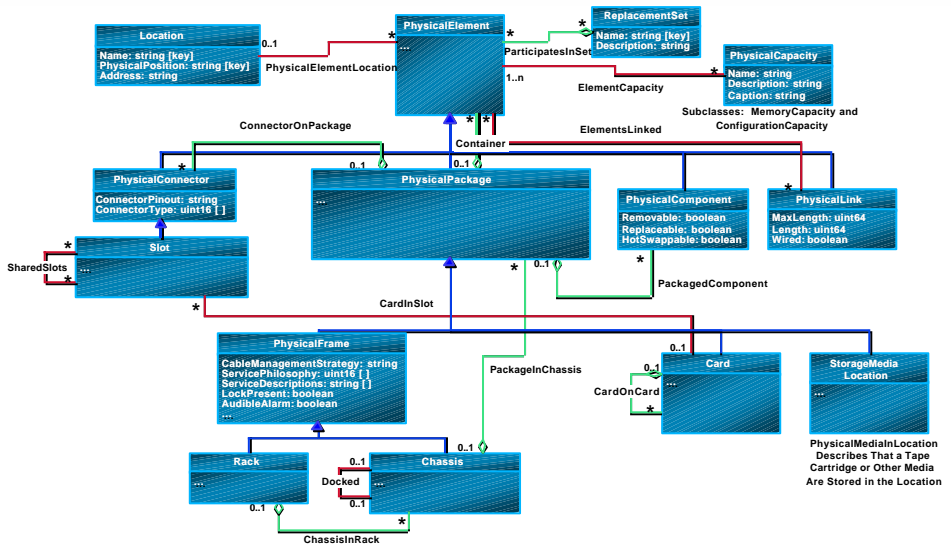
Agenda

- **CIM**

  Overview

  Core, System and Device Common Models

  Physical Model in Detail

  Network Model in Detail

- **DEN**

Network Model—Approach

- **Purpose**

  Contains classes and associations that model and manage network elements and services

- **Philosophy**

  Uses and builds upon general concepts from the core model

  References concepts in other common models

  Accordingly

  No new top-level classes

  Minimum number of top-level associations

Overview of the Network Model Infrastructure

ManagedSystemElement

LogicalElement

System Filters ServiceAccessPoint Service

Computer System Admin Domain Protocol Endpoint Network Service

CollectionOfMSEs

LogicalNetwork
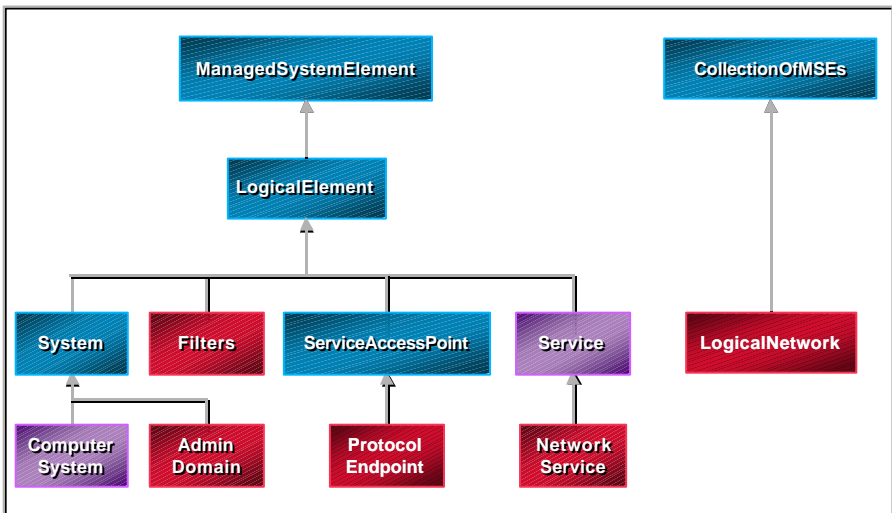
Network Model Highlights

- **Feature highlights**

  Network model infrastructure

  BGP sub-model

  Multiprotocol bridge sub-model

  VLAN sub-model

- **Salient features are overviewed in the following seven charts**

ProtocolEndpoint Relationships

ServiceAccessPoint

InLogicalNetwork

ActiveConnection

BindsTo

EndpointIdentity

0..1

LogicalNetwork

ProtocolEndpoint

Name: string
NameFormat: string
Type: uint16 (enum)
OtherTypeDescription: string
Speed: uint64
MaxSpeed: uint64

ProvidesEndpoint

ForwardsAmong

CalculatesAmong

BindsToLANEndpoint

0..1

NetworkService

ForwardingService

RouteCalculationService

LANEndpoint

NetworkService Subclasses

NetworkService

CalculatesAmong

ProtocolEndpoint

ForwardingService
ProtocolType: uint16 (enum)
OtherProtocolType: string

IPRoute

RouteCalculationService
RouterID: string
Type: uint16 (enum)

EGPRouteCalcDependency

ForwardsAmong

HostedForwardingService

ForwardedRoutes

CalculatedRoutes

ComputerSystem

HostedRoutingService

BGP Operation

BGPCluster

ReflectorService

BGPAdminDistance

BGPService
BGPVersion: uint8
LastErrorCode: uint8 (enum)
LastMessageErrorSubCode: uint16 (enum)
LastOpenErrorSubCode: uint16 (enum)
LastUpdateErrorSubCode: uint16 (enum)

AdministrativeDistance

ReflectorClientService

ReflectorNonClientService

BGPServiceStatistics

BGPStatistics

RoutingPolicy

BGPServiceAttributes

BGPRoutingPolicy

BGPPeerGroupService

BGPPathAttributes

BGPPeerGroupService

Routing Policy Objects

BGPRoutingPolicy

BGPService

* **RoutingPolicy**

CreationClassName: string [key]
Name: string [key]
Description: string
Action: uint16 (enum)
BGPAction: uint16 (enum)
BGPValue: string
RemarkAction: uint16 (enum)
RemarkValue: string
ConditioningAction: uint16 (enum)
OtherConditioningAction: string
ConditioningValue: string
AttributeAction: uint16 (enum)

BGPAttributes

FilteredBGPAttributes

ListsInRoutingPolicy

BGPRouteMapsInRoutingPolicy

FilterEntry

CreationClassName: string [key]
Name: string [key]
TrafficType: uint16 (enum)
MatchConditionType: uint16 (enum)
OtherMatchCondition: string
MatchConditionValue: string
Action: uint16 (enum)

* **FilterList**

CreationClassName: string [key]
Name: string [key]
Direction: uint8 (enum)

1

0..1

BGPRouteMap

EntriesInFilterList

3010
1170_05_2000_c2   © **2000, Cisco Systems, Inc.**

cisco.com

57

SwitchPort

SwitchesAmong

Switchable

SwitchPortStaticForwarding

0..1

SwitchService

0..1

LANEndpoint

0..1   *

SwitchPort

PortNumber: uint16

1..n

1..n   **SwitchPortPair**
1..n

1

StaticForwardingEntry

SwitchPortTransparentBridgingStatistics

InFrames: uint32
OutFrames: uint32
InDiscards: uint32

0..1

SAPStatistics

*   *   1   1

DynamicForwardingEntry

SpanningTreeService

*   **SwitchPortSpanningTree**

*

SwitchPortDynamicForwarding

SourceRoutingService

0..1   **SwitchPortSourceRouting**

0..1   **SwitchPortStatistics**

SAPStatistics

3010
1170_05_2000_c2   © **2000, Cisco Systems, Inc.**

cisco.com

58

SwitchesAmong

SwitchServiceTransparentBridging   SwitchServiceSpanningTree

SwitchPort

TransparentBridgingService
AgingTime: uint32
FID: uint32

1..n   0..1   1..n

SwitchService
BridgeAddress: string
BridgeType: uint16 (enum)
NumPorts: uint16

SpanningTreeService
ProtocolSpec: uint16 (enum)
Priority: uint16
DesignatedRoot: string
RootCost: uint16
RootPort: uint16
MaxAge: uint32
HelloTime: uint32
HoldTime: uint32
ForwardDelay: uint32
BridgeMaxAge: uint32
BridgeHelloTime: uint32
BridgeForwardDelay: uint32
SpanningTreeBridgeAddress: string

StaticForwardingEntry

1 1
TransparentBridging
StaticForwarding
w

1..n

DynamicForwardingEntry

TramsparentBridging
w DynamicForwarding

SourceRoutingService
BridgeLfMode: uint8 (enum)

0..1

SwitchPortSourceRouting

SwitchServiceSourceRouting

SwitchPortSpanningTree

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

59

RelatedSpanningTree

SwitchServiceSpanningTree

TransparentBridgingService
AgingTime:uint32
FID:uint16

1..n   *   0..1

SwitchService
BridgeAddress:string
NumPorts:uint16
BridgeType:uint16 (enum)

SpanningTreeService
ProtocolSpec:uint16 (enum)
Priority:uint16
DesignatedRoot:string
RootCost:uint16
RootPort:uint16
MaxAge:uint16
HelloTime:uint16
HoldTime:uint16
ForwardDelay:uint16
BridgeMaxAge:uint16
BridgeHelloTime:uint16
BridgeForwardDelay:uint16

1

1   *

SwitchService
TransparentBridging

1..n 1

TransparentBridging
DynamicForwarding

RelatedSpanningTree

SwitchServiceVLAN

DynamicForwardingEntry

VLAN
VLANNumber:uint32

VLANFor 1..n **VLANService**

InboundVLAN   OutboundVLAN

SwitchPort

SwitchPort
DynamicForwarding 1

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

60

Agenda

- **CIM**

    Overview

    Core, System and Device Common Models

    Physical Model in Detail

    Network Model in Detail

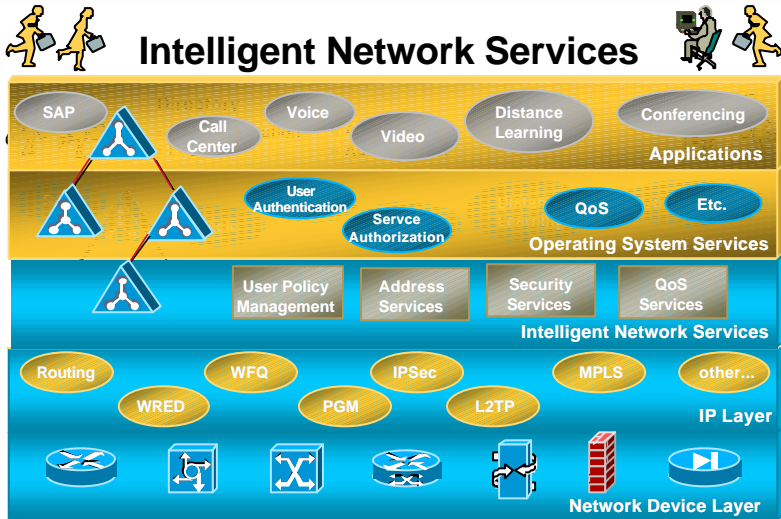- **DEN**

Agenda

- **DEN Overview**

    Motivation and Problem Statement
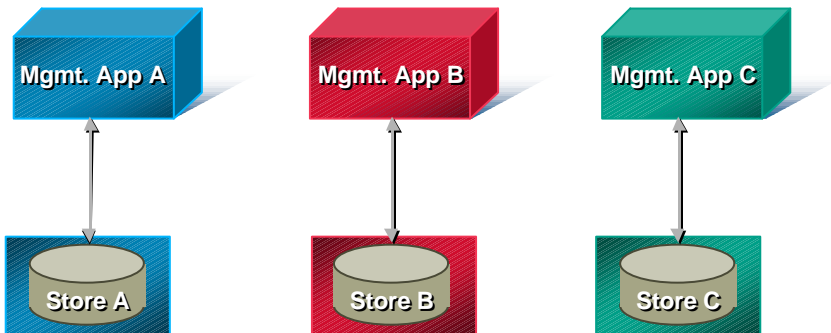
    Solution and Enhancements for Networking

    DEN and Standards

Intelligent Network Services

Intelligent Network Services

Applications
- SAP
- Call Center
- Voice
- Video
- Distance Learning
- Conferencing

Operating System Services
- User Authentication
- Servce Authorization
- QoS
- Etc.

Intelligent Network Services
- User Policy Management
- Address Services
- Security Services
- QoS Services

IP Layer
- Routing
- WFQ
- IPSec
- MPLS
- other...
- WRED
- PGM
- L2TP

Network Device Layer

3010
1170_05_2000_c2 © 2000, Cisco Systems, Inc.

cisco.com

63

The Need for Data Interoperability

Application-Specific Data Stores Limit Interoperability

- Mgmt. App A
- Mgmt. App B
- Mgmt. App C

- Store A
- Store B
- Store C

Integration Is Complicated and Costly

3010
1170_05_2000_c2 © 2000, Cisco Systems, Inc.

cisco.com

64

Agenda

- ## DEN Overview

  Motivation and Problem Statement

  **Solution and Enhancements
  for Networking**

  DEN and Standards

What Is the Solution?

- **An information model that defines management abstractions of**

  Profiles and policies

  Devices, protocols, and services

- **This provides**

  A unified model for integrating users, applications, and networking services

  An extensible service-oriented framework

Overall Goals

- **Topology and device discovery**

  Requires use of extensions to DEN model to ensure that best-of-breed apps can share and reuse data

- **Collect current configurations and convert them into policy statements**

  Achieved through combination of DEN policy model extensions and mechanism models

- **Wizards for domain-specific policy creation**

  Use of DEN model plus extensions to bind policy to Cisco-specific devices and mechanisms

Strategic Importance of DEN

- **Configuration is not the same as management**

  Customers need integrated management tools

  Management, ease of use, and product integration is frequently more important than performance

- **DEN is an integral part of current and future Cisco products**

  Common info model enables product-specific information to be shared and reused with other management products

  VPNSC: integrates VPN provisioning and management; brings together MPLS and IPSec under one product

  CW2K: integrates different modules; enables CW2K to talk to other applications

Agenda

- ## DEN Overview

  Motivation and Problem Statement

  **Solution and Enhancements
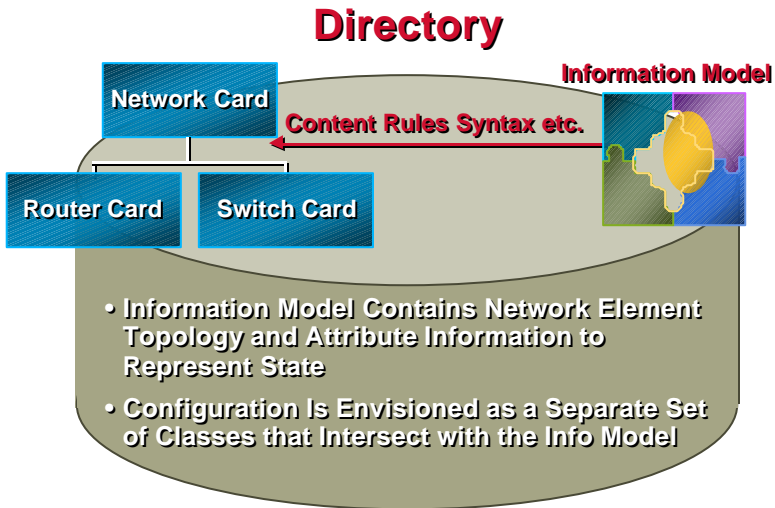 for Networking**

  DEN and Standards

DEN Specification

- **Defines a set of CIM classes plus new classes (i.e., extensions) to form an information model describing network elements and services**

  Extensions are specific to networking, whereas CIM network model is still general purpose

- **Also defines a mapping to a form suitable for implementation in a directory**

Information Model Content

Directory

- **Information Model Contains Network Element Topology and Attribute Information to Represent State**
- **Configuration Is Envisioned as a Separate Set of Classes that Intersect with the Info Model**

Information Model Behavior

- **Defines object relationships**

- **Specifies realization of business logic (usually in middleware)**

- **Enables sharing and exchange of business rules, SLAs, and policies**

DEN Implementation

Vendor-Specific Extensions

DEN Schema

Native Directory Schema
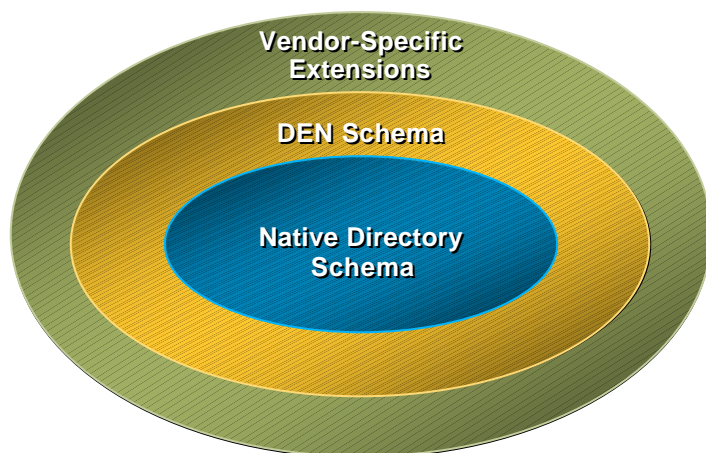
Key Standards Affected by DEN

- **DMTF**

    Network Model, LDAP mappings WGs

    Also Core, User, Security, and SysDev WGs

- **IETF**

    Policy (Policy Framework, IPSP)

    Technology Mechanisms (DiffServ, RAP, DHC)

    Directory Technology (LDAPEXT and LDUP)

    SNMPConf (and Polterm, CfgMgmt)

CIM/DEN Update

- **CIM 2.3**
  - Adds the user-security model
  - Numerous updates to system and devices
  - Roots the model

- **CIM 2.4**
  - Adds the QoS and IPSec sub-models to the network model
  - Adds the IETF policy core model

- **DEN**
  - New version will be published later this year
  - Also includes other models, including the IETF policy QoS and device QoS info models, and the IPSec policy model

---

Agenda

- **Object-Oriented Modeling**

- **CIM and DEN**

- **Cisco's Extensions to DEN**

- **Policy Info Model and LDAP Realization**

- **Use of DEN in Cisco Products**
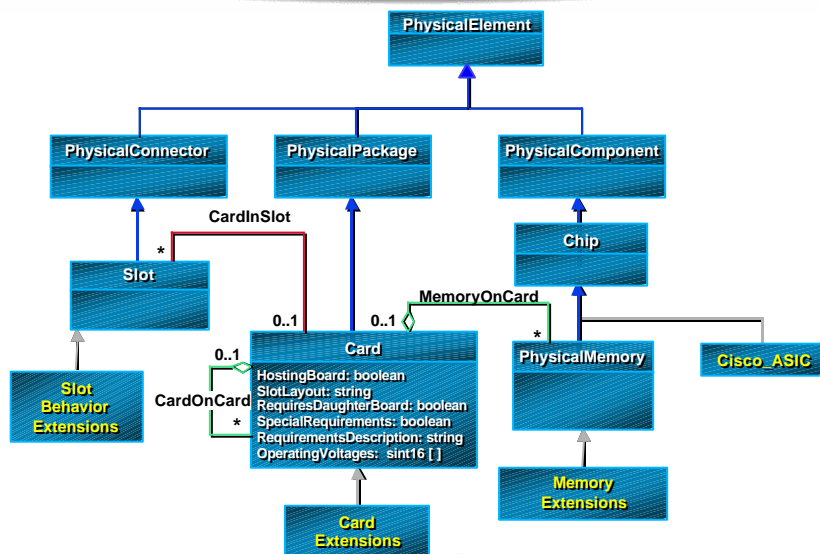
- **Futures for DEN and PBM**

CIM-XC

- **Joint EMBU-NSMBU DEN-based effort creating a unified information model for Cisco devices and services**

- **Cisco apps can store information and make it available…**

   Through directory/LDAP schema mappings and operations

   Through databases and xmlCIM operations

What Do the Cisco Extensions Look Like?

PhysicalElement

PhysicalConnector   PhysicalPackage   PhysicalComponent

CardInSlot

Slot

Chip

MemoryOnCard

0..1   0..1

Card
HostingBoard: boolean
SlotLayout: string
RequiresDaughterBoard: boolean
SpecialRequirements: boolean
RequirementsDescription: string
OperatingVoltages: sint16 [ ]

PhysicalMemory

Cisco_ASIC

0..1

Slot Behavior Extensions

CardOnCard

Card Extensions

Memory Extensions

The Cisco Schema Police

- **Group of engineers and consultants from various business units to:**

  Develop information model and schema extensions that ALL Cisco products can use

  Mentor people in becoming information modeling and schema experts (similar to the MIB police)
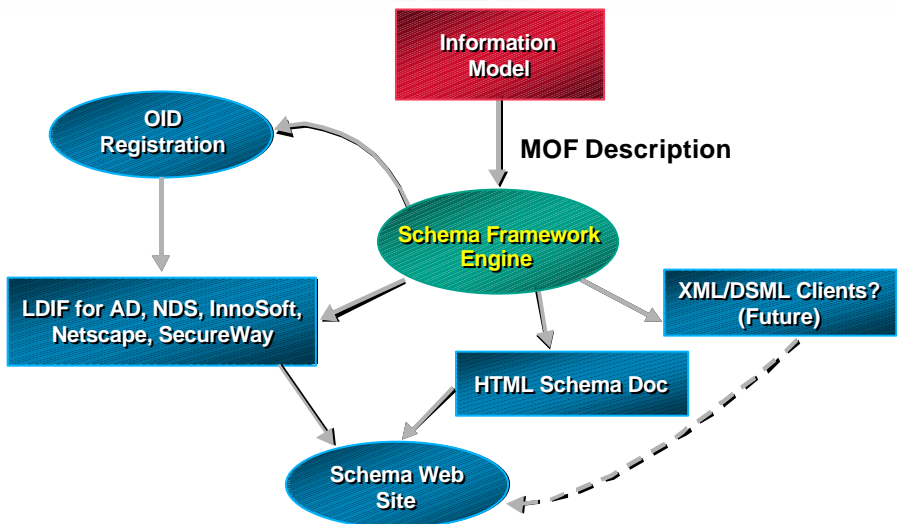
How to Extend the Schema?

- **How to maintain compatibility with emerging standards**

  Must work in the standards bodies

- **How to make it directory-independent**

  Must know strengths and weaknesses of various directory vendors and of LDAP

- **How to extend the schema without breaking existing applications**

  Cisco process described in next slides

How to Extend the Schema?

- **Information model analysis, produce UML diagrams and describe using MOF**

- **Map data in information model to LDAP schema capabilities**

- **Submit the proposed schema to schema police for review**

- **Revise the schema (iterate as necessary)**

- **Official OID assignment and publish schema at internal schema web site**

The Schema Process

Information Model

MOF Description

OID Registration

Schema Framework Engine

XML/DSML Clients? (Future)

LDIF for AD, NDS, InnoSoft, Netscape, SecureWay

HTML Schema Doc

Schema Web Site

Cisco-Specific Extensions

- **Common schema**

  DEN foundation (CIM 2.2/2.3/2.4 model)

  DEN extensions (Policy, QoS, IPSec models)

- **New bridging classes**

  Bridge physical and logical sub-models

  Bridge different types of logical concepts

  Associate services with a network device

  Associate paths with network elements and services

  Associate users/applications with network elements and services

Cisco-Specific Extensions (2)

- **Cisco extensions**

  Physical (inventory for Cisco devices)

  System (Cisco network devices)

  Network (Cisco network services)

  User (privileges, credentials, AAA)

  Application (e.g., ASP applications)

  Policy (DiffServ and RSVP; DNS, DHCP, RADIUS; IPSec and VPN provisioning…)

  Services (DiffServ and RSVP; DNS, DHCP, RADIUS; IPSec, VPNs, VoXXX,…)

Agenda

- **Object-Oriented Modeling**

- **CIM and DEN**

- **Cisco's Extensions to DEN**

- **Policy Info Model and LDAP Realization**

- **Use of DEN in Cisco Products**

- **Futures for DEN and PBM**

Agenda

- **Policy Info Model and LDAP Realization**

    Theory

    LDAP Realization

    QoS Extensions

How Do We Use the QoS Model?

- **Problem: How do we link business rules and policies to network configuration and provisioning?**

Solution

- **Solution: Realize that policy is actually a continuum of policies expressed at different domains**

  Business rule

  Device-independent specification

  Set of device-specific specifications

- **DEN policy information model is mapped to each set of policies**

  QoS model layered on (layered) network model

  QoS model covers policy and network devices

Policy Layers

- **Administrator-defined: device- and technology-independent**
  IF user is subscribed to Gold Service, THEN allow use of
  NetMeeting and provide premium data services

- **Device-independent policy rules**
  If SourceIPAddress == 172.3.128.0/15,
  THEN mark voice with EF and Data with AF11

- **Device-independent, mechanism-dependent policy**
  configuration take three forms…

  Configure component so it can be used to condition forwarded traffic

  Configure component so it can act on traffic directly

  Trigger action based on network or system event (e.g., link failure)

- **…And perform a set of device-independent actions:**

  Configure classifier, configure filter and bind to classifier,…

Policy Meta Definition

- **What is policy?**

  In our case, a declarative system primarily
  for controlling device configuration

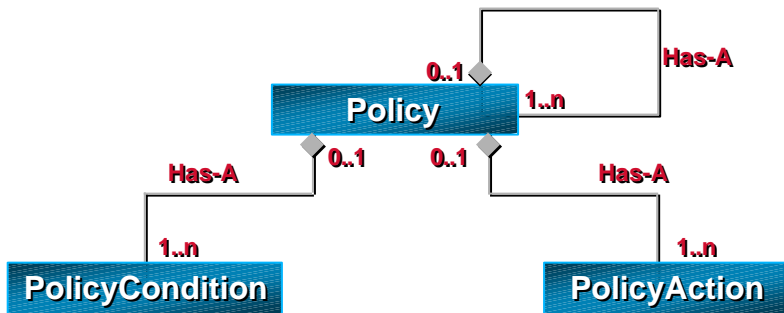 **Describes desired state, not changes**
  to get there

  Monitors system to verify conformance
  and adjusts as necessary

  In general, controls configuration of objects
  to regulate access and use of resources

Policy Meta Model

Goal: Related to Business Goal or Rule   **SLA**

Disabled => Disable All Strategies;
Enabled => Manage Individual Strategies

Strategy: How to Accomplish a Goal   **SLO(s)**

Disabled => Disable All Policies;
Enabled => Manage Individual Policies

Policy: Describes Overall Business Function to Be Accomplished

Disabled => Disable Specific Policies;
Enabled => Manage Individual Mechanisms   **PDL**

Conditions and Actions: Defines When Policy Is Active, Specific Mechanisms to Invoke, and Supplies Parameters to Those Mechanisms

---

Policy Model

Has-A

0..1   1..n

Policy

Has-A   0..1   0..1   Has-A

1..n   1..n

PolicyCondition   **PolicyAction**

Approach

Define generic policy information model

Draft-ietf-policy-core-info-model-06.txt

Refine representation of policy to suit the application-specific needs of DiffServ and RSVP

Draft-ietf-policy-qos-info-model-01.txt

Refine network model to suit the application-specific needs of DiffServ and RSVP
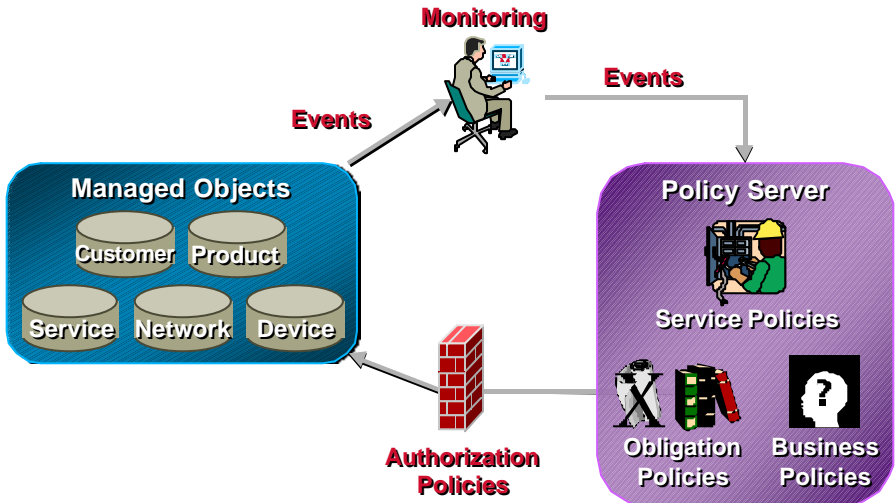
Draft-ietf-policy-qos-device-info-model-01.txt

Ensure that each can be mapped into a directory implementation

Draft-ietf-policy-qos-device-schema-00.txt

Approach (2)

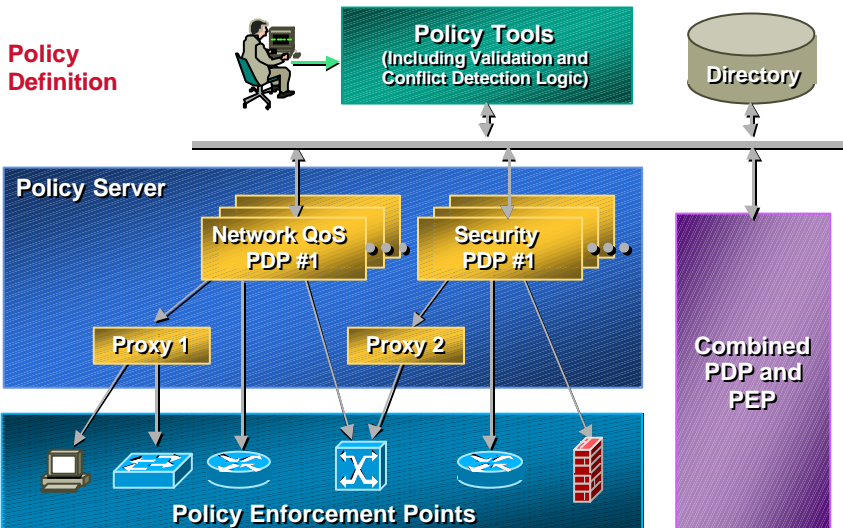- **Separates modeling of policies from modeling of device mechanisms**

- **Ensures that the information model can accommodate repository-specific requirements without biasing the information model to any one repository**

- **Policy application is standardized on**

    Representation of policies

    Representation of device mechanisms

- **Result is that policy implementation is based on standards**

Policy Networking

Monitoring

Events

Events

Managed Objects
- Customer
- Product
- Service
- Network
- Device

Authorization Policies

Policy Server

Service Policies

Obligation Policies

Business Policies

Conceptual Policy Architecture

Policy Definition

Policy Tools
(Including Validation and Conflict Detection Logic)

Directory

Policy Server

Network QoS PDP #1

Security PDP #1

Proxy 1

Proxy 2

Combined PDP and PEP

Policy Enforcement Points

Agenda

- **Policy Info Model and LDAP Realization**

    Theory

    LDAP Realization

    QoS Extensions
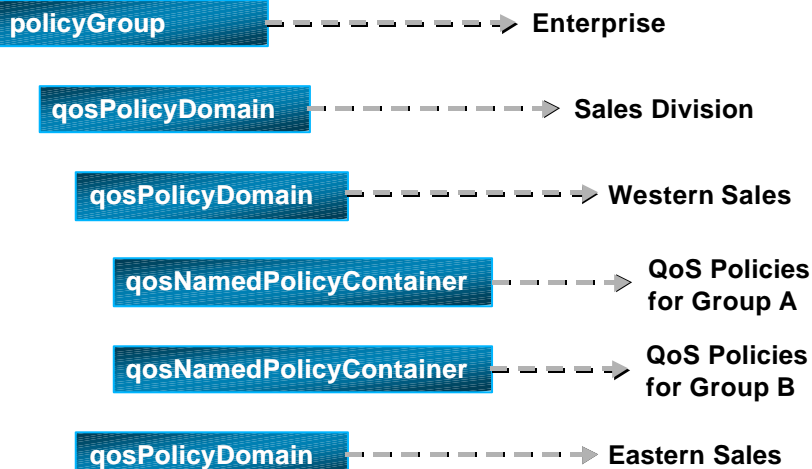
Details of QoS Policy Approach

- **Keep structure of policy information**

    But specialize to add concepts particular to DiffServ and RSVP

- **Divide the repository into two areas**

    One for definition of policy objects

    One for reusable policy objects

- **Use associations to implement containment**

    Enables rule-specific and reusable objects to be built

Expressing Containment

policyGroup	- - - - - - - - - -▷ **Enterprise**
qosPolicyDomain	- - - - - - - - - -▷ **Sales Division**
qosPolicyDomain	- - - - - - - - - -▷ **Western Sales**
qosNamedPolicyContainer	- - - - - -▷ **QoS Policies for Group A**
qosNamedPolicyContainer	- - - - - -▷ **QoS Policies for Group B**
qosPolicyDomain	- - - - - - - - - -▷ **Eastern Sales**

Example: Use of Association Classes in QoS Policy Rules

IF (cond1 or cond2) THEN <prAct1>;

OC: PolicyRule
policyRulePriority = 1
PolicyRuleConditionListType = 1

OC: 'policyRuleConditionAssociation'
policyConditionGroupNumber = 1
policyConditionNegated = FALSE
policyConditionName = 'Cond1'
 ←- - - **Attachment** - - - **OC: qosPolicySimpleCondition (See next slide)**

OC: 'policyRuleConditionAssociation'
policyConditionGroupNumber = 2
policyConditionNegated = FALSE
policyConditionName = 'Cond2'
policyConditionDN = DN:Cond2
 —— **DN** ——▶ **Reusable Object: policyConditionInstance + qosPolicySimpleCondition**

OC: 'policyRuleActionAssociation'
policyActionOrder = 1
policyActionName = 'prAct1'
 ←- - - **Attachment** - - - **OC: qosPolicyPRAction qpSetDSCPvalue = 5**

Example: Simple Condition Referencing a Reusable Value

'SourcePost == MyWEBServerPort'

OC: qosPolicySimpleCondition
qpValueAtom = DN:MyWEBServer

Reusable Objects Repository

DN

The condition may be attached to an instance of policyRuleConditionAssociation (ad-hoc) or to an instance of policyConditionInstance (reusable)

OC: PolicyInstance
PolicyInstanceName = 'MyWEBServerPort'

OC: qosPolicyVariable
qpVaribleName = 'SourcePort'

OC: qosPolicyIntegerValue
qpIntegerList = 8080

Attachment →

Simple Condition Examples—Reference

'SourcePost == MyWEBServerPort'

Object Class:qosPolicySimpleCondition
Type: Integer OID
Operator: "=="
PolicyVariableAtom: DN
PolicyConstantAtom: DN

Simple Condition Composed by Referencing a Variable and a Constant

PolicyInstance

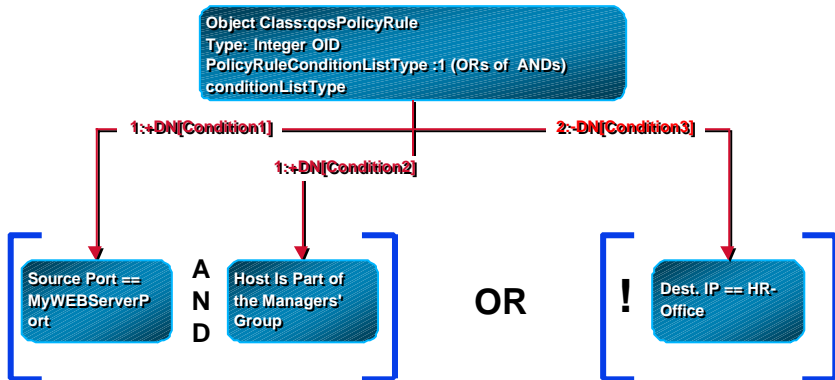**Obj. Class: qosPolicyConstant**
Name: SourcePort
Type: Integer OID

Obj. Class: qosPolicyNumberValue
Type: Integer OID
PortValue: 80

PolicyInstance

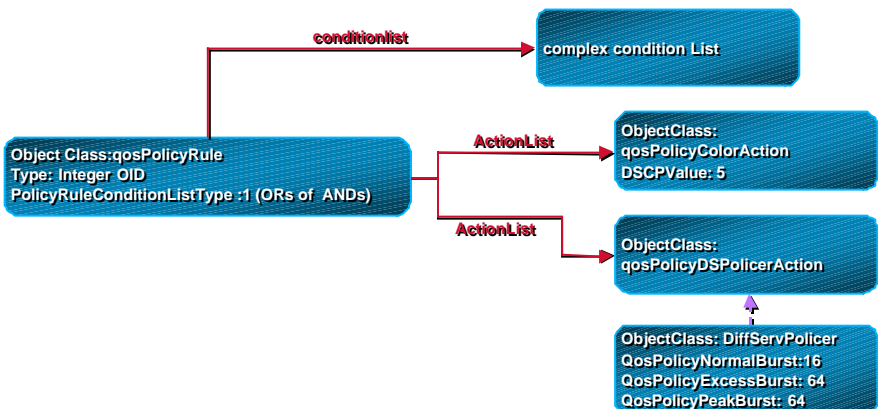**Obj. Class:qosPolicyVariable**
Name: SourcePort
Type: Integer OID

Complex Condition Example

[SourcePort == MyWEBServerPort & Host Is Manager] OR [Destination IP != HR-Office]

Object Class:qosPolicyRule
Type: Integer OID
PolicyRuleConditionListType :1 (ORs of ANDs)
conditionListType

1:+DN[Condition1]

1:+DN[Condition2]

2:-DN[Condition3]

Source Port == MyWEBServerPort

A N D

Host Is Part of the Managers' Group

OR

!

Dest. IP == HR-Office

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

103

Complex QoS Policy Rule

IF [SourcePort == MyWEBServerPort & Host Is Manager] OR [Destination IP != HR-Office] THEN Color DSCP=5, Policer=<16, 64,64>

conditionlist

complex condition List

Object Class:qosPolicyRule
Type: Integer OID
PolicyRuleConditionListType :1 (ORs of ANDs)

ActionList

ObjectClass:
qosPolicyColorAction
DSCPValue: 5

ActionList

ObjectClass:
qosPolicyDSPolicerAction

ObjectClass: DiffServPolicer
QosPolicyNormalBurst:16
QosPolicyExcessBurst: 64
QosPolicyPeakBurst: 64

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

104

Agenda

- **Policy Info Model and LDAP Realization**

  Theory

  LDAP Realization

  QoS Extensions

Overview of the QoS Model

LogicalElement

- Filters
- Statistics
- Service
  - NetworkService
    - BufferManagementService
    - ForwardingService
      - PacketSchedulingService
      - TCB
        - ClassifierService
        - MeterService
        - MarkerService
        - DropperService
        - QueuingService
    - QoSService
      - DiffServService, RSVPService, PrecedenceService classes
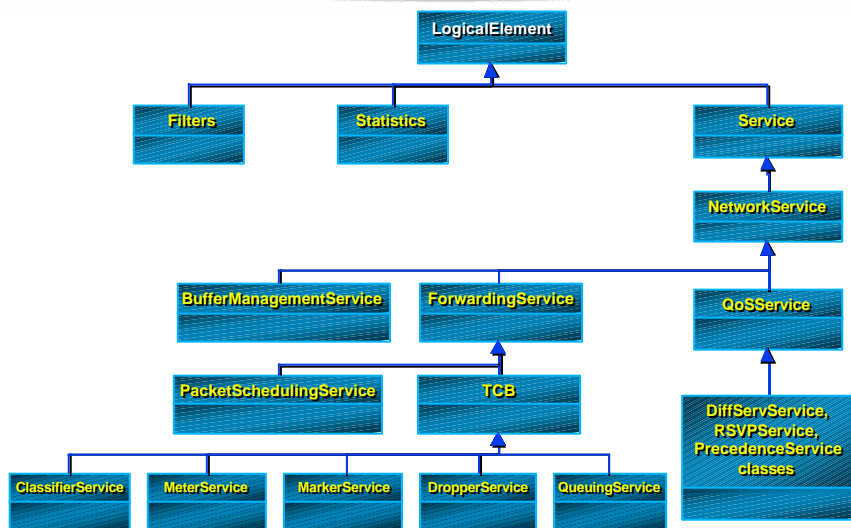
QoS Capabilities

- **Describes the QoS capabilities of a device**

    Identify available QoS services

    Filters the classifier can use

    Profiles the metering service can enforce

    Actions the QoS service can support

    Describe logical filter/meter/action combinations per service

    Describe available algorithms (e.g., queuing disciplines, how to drop packets, etc.)

QoS Actions—Provisioned

- **Actions applied based on how a packet is classified**

    Packet can be tagged (802.1p, ToS, DSCP)

    Matching filter used to classify flows

    Traffic profile parameters applied to filter using one or more meters

    Dispositions can be Drop, Remark, Shape

    Queues can be selected based on filters

    Can map tags to queues

QoS Usage Model—Signaled

- **Can be used to signal QoS for legacy apps**
    - **RSVP proxy can perform RSVP signaling**
    - **If filter match for outgoing packet**
        - **Determine QoS required (based on filter)**
        - **Send path message for specified TB params**
        - **Corresponding Resv messages accepted**
    - **If filter match for incoming packet**
        - **Proxy checks for corresponding path state**
        - **Determine QoS required (based on filter and path)**
        - **Send Resv message to data source**

Agenda

- **Object-Oriented Modeling**

- **CIM and DEN**

- **Cisco's Extensions to DEN**

- **Policy Info Model and LDAP Realization**

- **Use of DEN in Cisco Products**

- **Futures for DEN and PBM**

Building Around a Directory (Developer Perspective)

- **Strengths**

  Highly distributed, replicable, scalable

  Optimized for read access

  Open, standard API (LDAP)

- **Weaknesses**

  No support for transactions

  No support for change notification

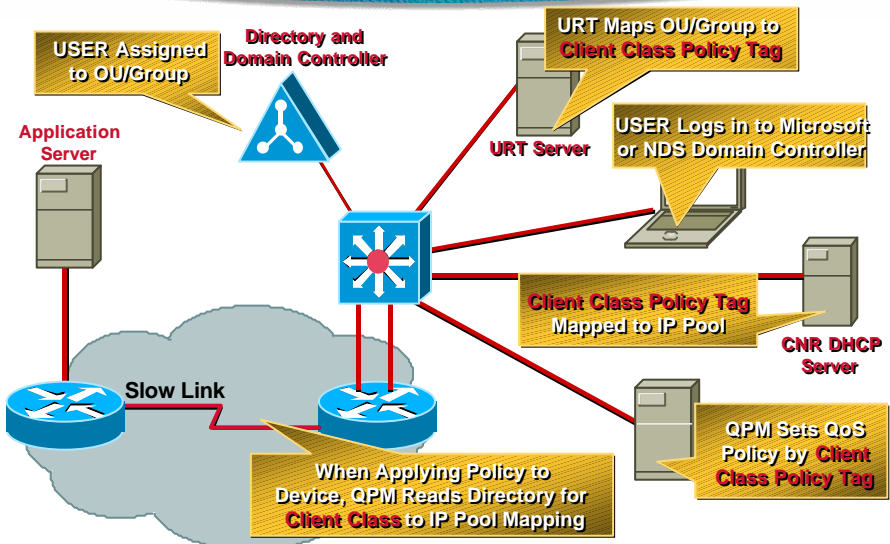# What Are We Delivering in CY 2000?

- **Using directories to authenticate users and bind them to groups, policies, services, and network entities**

- **Rolling out (app-by-app) integration with the leading enterprise directory services**

- **Forging a unified, CIM-based information model to be used by a variety of Cisco applications**

Commitment to Interoperability

- **Cisco applications will support major enterprise directory services:**
  - **Active directory**
  - **NDS**
  - **iPlanet directory service**
  - **SecureWay directory service**
- **Cisco applications are built to standards:**
  - **LDAPv3, COPS, DiffServ, etc.**
  - **CIM and DEN-compliant**
  - **IETF policy framework standards**

Directory Enabled User Policy Using QPM-CNR-URT

USER Assigned to OU/Group

Directory and Domain Controller

URT Maps OU/Group to Client Class Policy Tag

URT Server

USER Logs in to Microsoft or NDS Domain Controller

Application Server

Client Class Policy Tag Mapped to IP Pool

CNR DHCP Server

Slow Link

QPM Sets QoS Policy by Client Class Policy Tag

When Applying Policy to Device, QPM Reads Directory for Client Class to IP Pool Mapping

Microsoft Win2000 Keynote Launch Demo Using QPM-URT

2. Entire Sales Executive OU Is Promoted with One Mouse-Click

Microsoft Win2000 Server:
Active Directory
+ PDC
+ MS-DHCP

Red Indicates What Will be Shown
Green Indicates What Happens Behind the Scenes

Microsoft Win2000 App Server:
Windows Streaming Media Server

6. DHCP Server Assigns New IP Based on the Gateway from Which It Received the Request

Client PC

1. Client Is Logged in (Prior to Promotion); Streaming Video Is Slow and Choppy; Client Logs Out

3. Client Logs Back in

8. Instantly After Login, the Streaming Video of John Chambers Is Perfectly Smooth

Default VLAN Gateway

Executive VLAN Gateway

Slow Link

Cisco Server: URT + QPM

4. URT Intercepts the Login and Instructs the Cat6k to Dynamically Reassign the Port to the Executive VLAN and Also Instructs Client PC to Release+ Renew IP Address

7. Router Has QoS Policies (Deployed via QPM) for Executive IP Address Pool

5. Router Forwards the DHCP Renew Request to DHCP Server (Which Is in the Default VLAN)

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

115

DEN Roadmap

Phase	I Building Foundation	II Integrated User Identity	III Policy Manage-ment	IV Unified Security	V Ongoing Integration
Start	1997	1999	1H00	2H00	2001
Key Features	Standards (DEN/CIM)	App Hooks to Identity, Device and Profiles	User and App Policies— e.g., for QoS	Common AAA Services	Policy-based Admin and Control
Apps	N/A (Standards Work)	CNR 3.0 URT 1.2 CNS	CNR URT QPM AVVID CNS VPNSC EMSes	CW2K Common Mgmt Foundation CNS VPNSC\ EMSes	Apps Previously Referenced, and Additions

3010
1170_05_2000_c2   © 2000, Cisco Systems, Inc.

cisco.com

116

Agenda

- **Object-Oriented Modeling**

- **CIM and DEN**

- **Cisco's Extensions to DEN**

- **Policy Info Model and LDAP Realization**

- **Use of DEN in Cisco Products**

- **Futures for DEN and PBM**

Inter-PDP Communication

- **Previous slide pointed out the need for PDP–PDP communication**

  Required to coordinate policies among multiple domains

  Enables different PDPs to share data, avoiding the building of complicated APIs

- **Two main directions**

    Modeling policy validation and distribution

    More rule and mechanism models

    IPSec, MPLS, DHCP, and others

- **Work will be done in IETF and DMTF**

    IETF concentrates mostly on protocol

    DMTF concentrates on integration into a larger management environment

DEN-Based Object Modeling and Schema for Policy-Based Networking

Session 3010

Please Complete Your Evaluation Form

Session 3010