

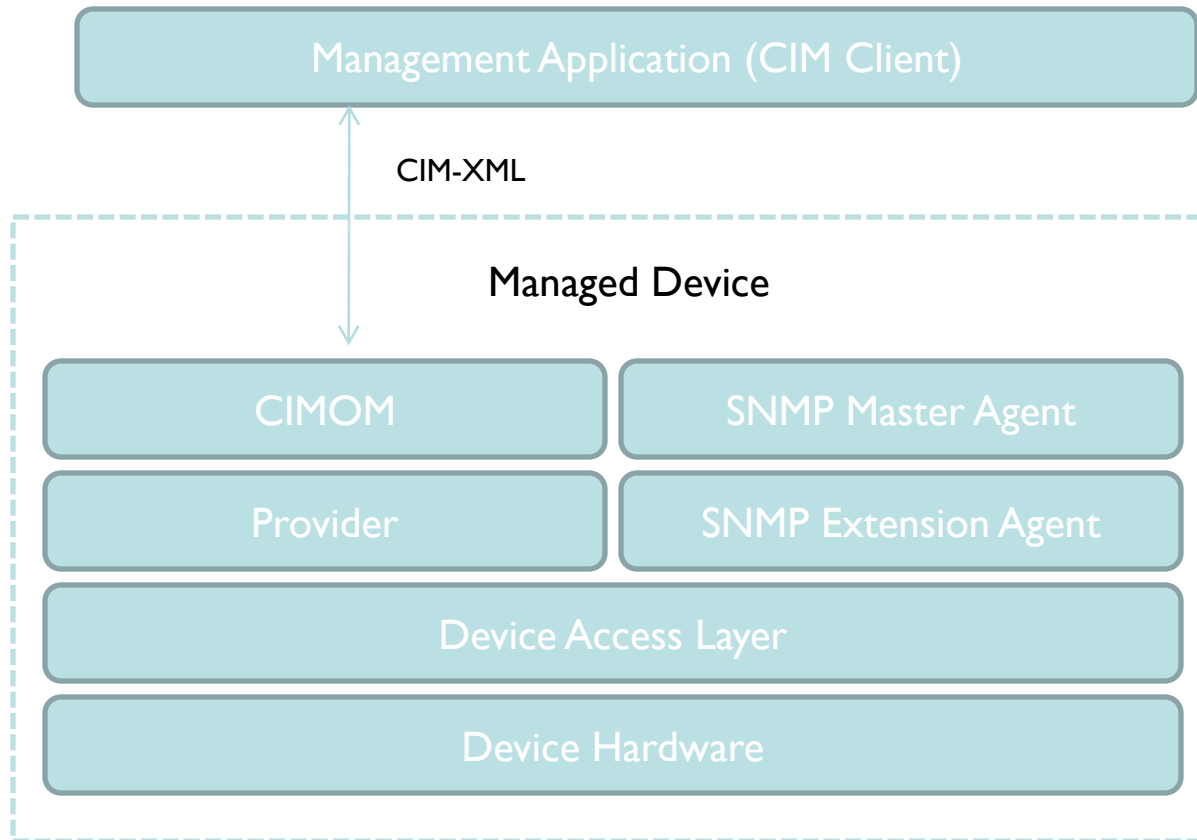
Proxy Providers versus Embedded Providers (SMI-S)

Srinivasa Reddy Gandlaparthi

NetApp

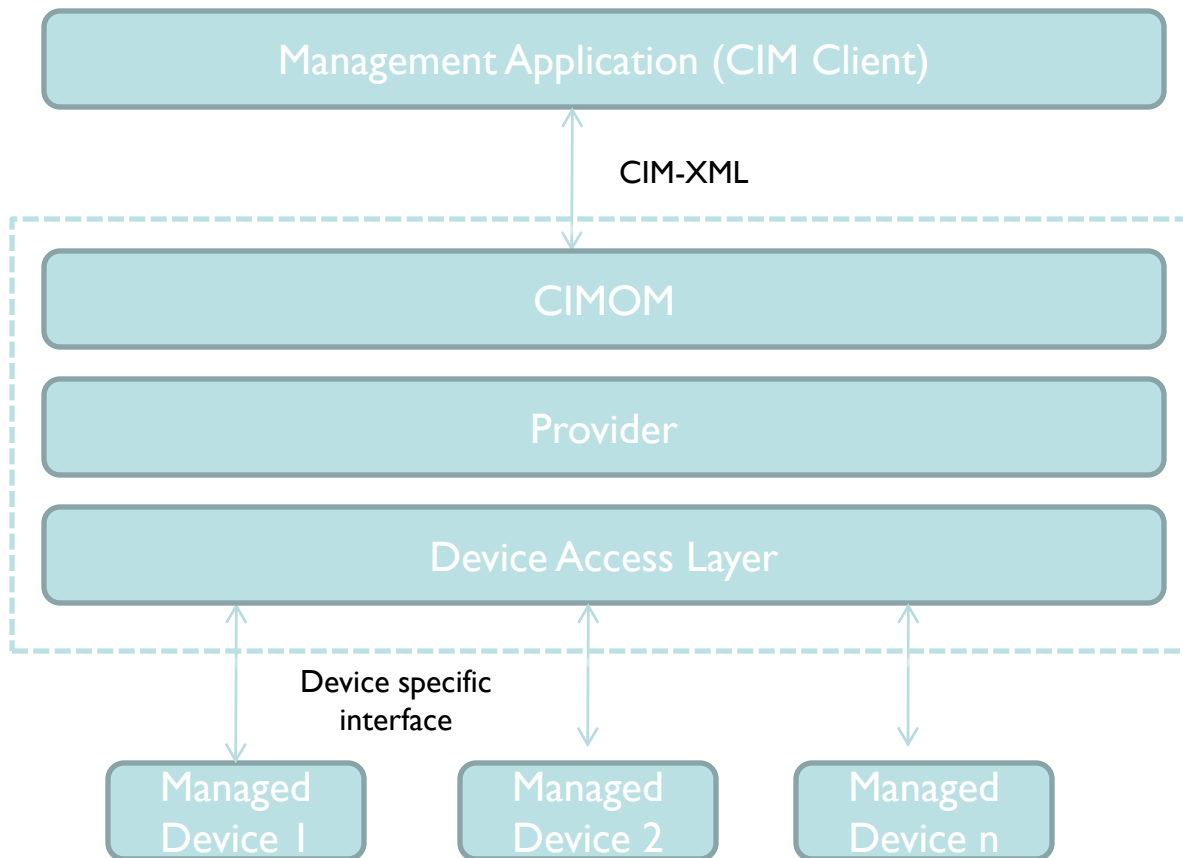
- ❑ Embedded Providers
- ❑ Proxy Providers
- ❑ Differences between Embedded and Proxy providers
- ❑ Design Considerations
- ❑ Considerations for Management applications
- ❑ Common issues and resolution for proxy providers

Embedded Providers



- ❑ Embedded providers Implementation Scenarios
 - ❑ Dedicated VM for CIMOM, Providers and Management Agents
 - ❑ Service processor / dedicated hardware for Management Agents and Provider implementation
 - ❑ As part of the managed device

Proxy Providers



- ❑ All managed devices represented in single namespace
 - ❑ For example root/vendorXXX namespace representing all the nodes discovered and managed by the proxy

- ❑ Each managed device represented in different namespace
 - ❑ For example root/vendorXXXDevice1, root/vendorXXXDevice2 etc

Design Considerations - Basics

- ❑ Memory, CPU usage
- ❑ Disk/flash space for binaries
- ❑ Initialization time of the provider to discover/add all managed nodes
- ❑ Response time for CIM Operations
- ❑ Network utilization – between proxy and managed nodes
- ❑ Simulators to test scalability of providers
- ❑ Number of Managed device
- ❑ Number of objects within each manage device

- ❑ Plan for Future Needs (may require additional runtime and flash/disk space or other resources)
 - ❑ WWS-Man support requirement by management application
 - ❑ Additional SMI-S profiles implementation in future

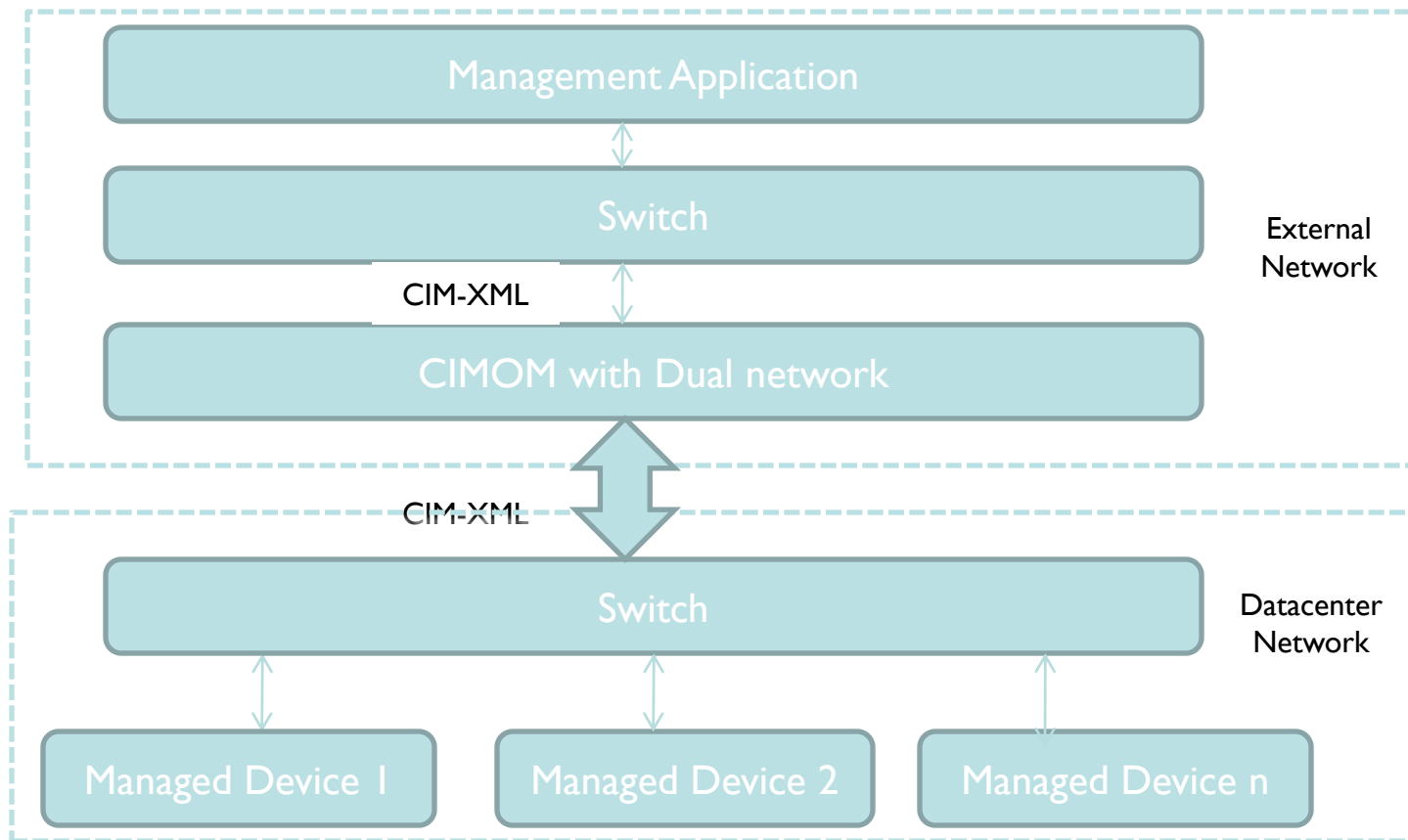
- ❑ Support for multiple Firmware versions and Managed devices configurations
 - ❑ Embedded: Providers can be upgraded to support latest Firmware at the time of firmware upgrade
 - ❑ Proxy: Design mechanism to handle different versions of Firmware dynamically

- ❑ Coexistence of different providers in proxy environment
 - ❑ Usage of CIMOMs with different configuration
 - ❑ Duplicate Open SLP installations
 - ❑ Usage of default CIMOMs available with Operating Systems (especially X86_64 versions)
 - ❑ Other providers failure causing unavailability

- ❑ Single point of failure

Design Considerations – Isolated network for managed devices

- Proxy: Isolated network for managed devices
 - Proxy with dual network access
 - Data center internal network isolated from rest of Network



Design Considerations – Multi-client communication with proxy

- ❑ Simultaneous access by multiple management applications
 - ❑ Management and monitoring
 - ❑ Event logs and Indication analyzers
 - ❑ Performance statistics collectors

- ❑ Number of connections supported by managed device simultaneously

- ❑ Proxy implementation often require multiple OS support:
 - ❑ CIMOM agnostic design - Business logic as SDK layer
 - ❑ CMPI interface implementation or Adapters
 - ❑ OS agnostics libraries (for example BOOST, xerces)

Design Considerations - User Management

- Proxy:
 - Access through single proxy user authentication
 - Managed devices added with passwords individually.

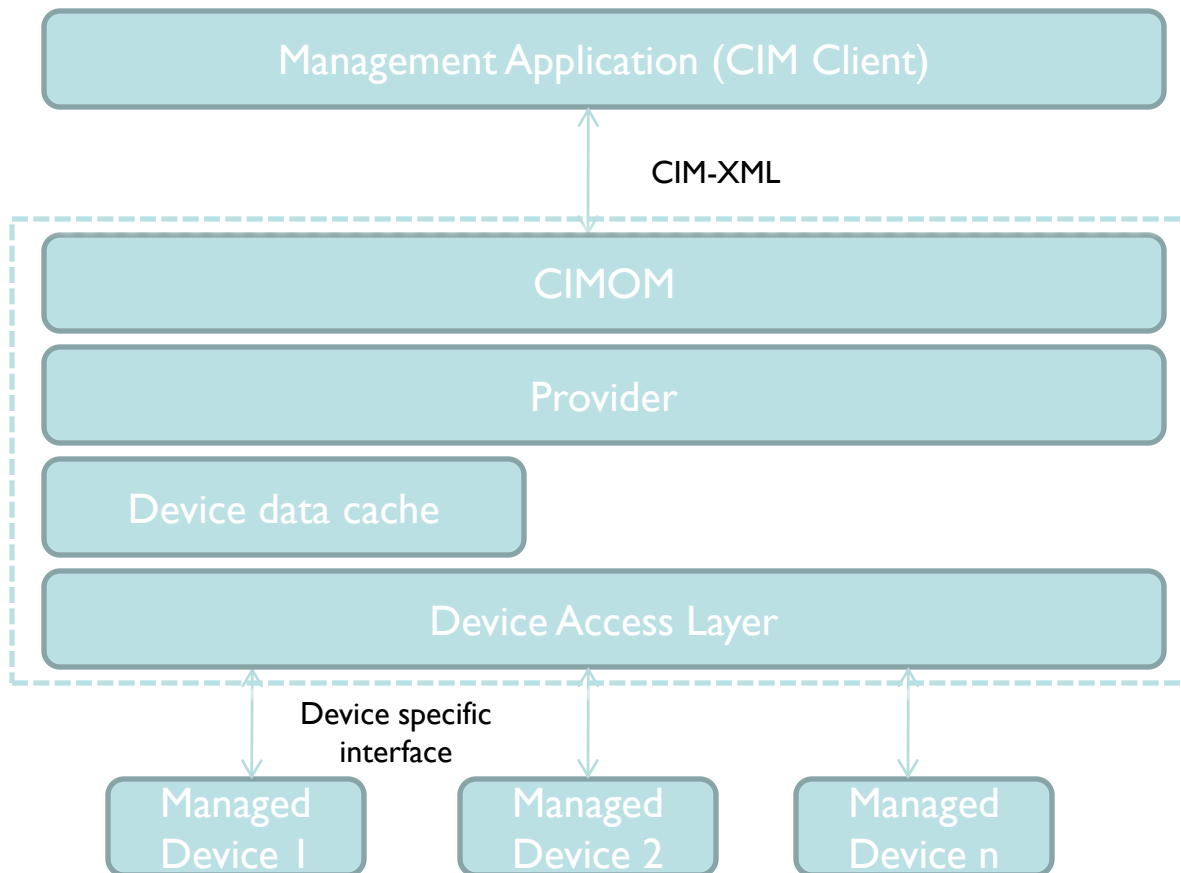
- Embedded:
 - Integration of CIMOM user accounts with Device user accounts.
 - Management of read-only users and Admin users

Management Applications using Associations

- ❑ Management Applications:
 - ❑ Focus on using association traversal even if the provider is embedded for extensibility to support Proxy providers
 - ❑ Traversal through instance names of source class, association class name to narrow down the search of resultant class instances
- ❑ Proxy Providers:
 - ❑ Retrieve information from cache or specific device only by using the input from the CIM Operations effectively

Proxy Providers – Improving Response time

- Caching data – In memory or File System or lightweight database with event based updates to the cache



Proxy Providers – Improving Response time

- ❑ Implementation of Block Storage Views Profile (Experimental in SMI-S 1.4.0) – Simplifies the retrieval of commonly used data with minimal number of instances retrieval and association traversals
 - ❑ Disk Drive View
 - ❑ Masking Mapping View
 - ❑ Volume View
 - ❑ Exposed View

Testing & Simulation with close to real time scenarios

- ❑ Testing using third party applications
- ❑ Frequently of object properties change to update cache
- ❑ Addition and removal of managed devices
- ❑ Indication registration and delivery
- ❑ Concurrent Jobs running and tracked by the provider
- ❑ Performance statistics polling
- ❑ Simultaneous client applications

- ❑ Planning Threads for:
 - ❑ Provider initialization
 - ❑ Data collection, polling and caching data
 - ❑ Performance Statistics collection
 - ❑ Tracking Job – Job Control Profile implementation
 - ❑ Events tracking and Indication delivery

Proxy Providers – Minimizing live data collection

- ❑ Performance Statistics (Block Server Performance Profile):
 - ❑ Collect statistics for all objects of same type (for example Storage Volume) within a Managed device at the same time.
 - ❑ Provide configurable time delay between two statistics collections to avoid continuous polling for statistics.
 - ❑ Enhance managed device capability to support Bulk data retrieval (Similar to SNMP Get Bulk).

- ❑ Minimize inclusion of live data from devices and cached information in the same CIM object instance by providers as large configurations can lead to delay in response time (especially the vendor specific data)

Queries @
gandlaps@NetApp.com

References

- ❑ SNIA (www.snia.org)
- ❑ SNIA Storage Management Initiative (www.snia.org/forums/smi)
- ❑ DMTF (www.dmtf.org)
- ❑ Xerces Libraries (xerces.apache.org)
- ❑ Boost Libraries(www.boost.org)