An Introduction to the Industry's leading Open Source High Availability base platform middleware

What I hope to accomplish

- Warm you up to OpenSAF
  - What is it?
  - What does it contain?

- OpenSAF Details
  - High Level Generic Architecture of OpenSAF Services
  - OpenSAF Dependencies
    - OpenSAF Platform touch-points
  - OpenSAF Manageability

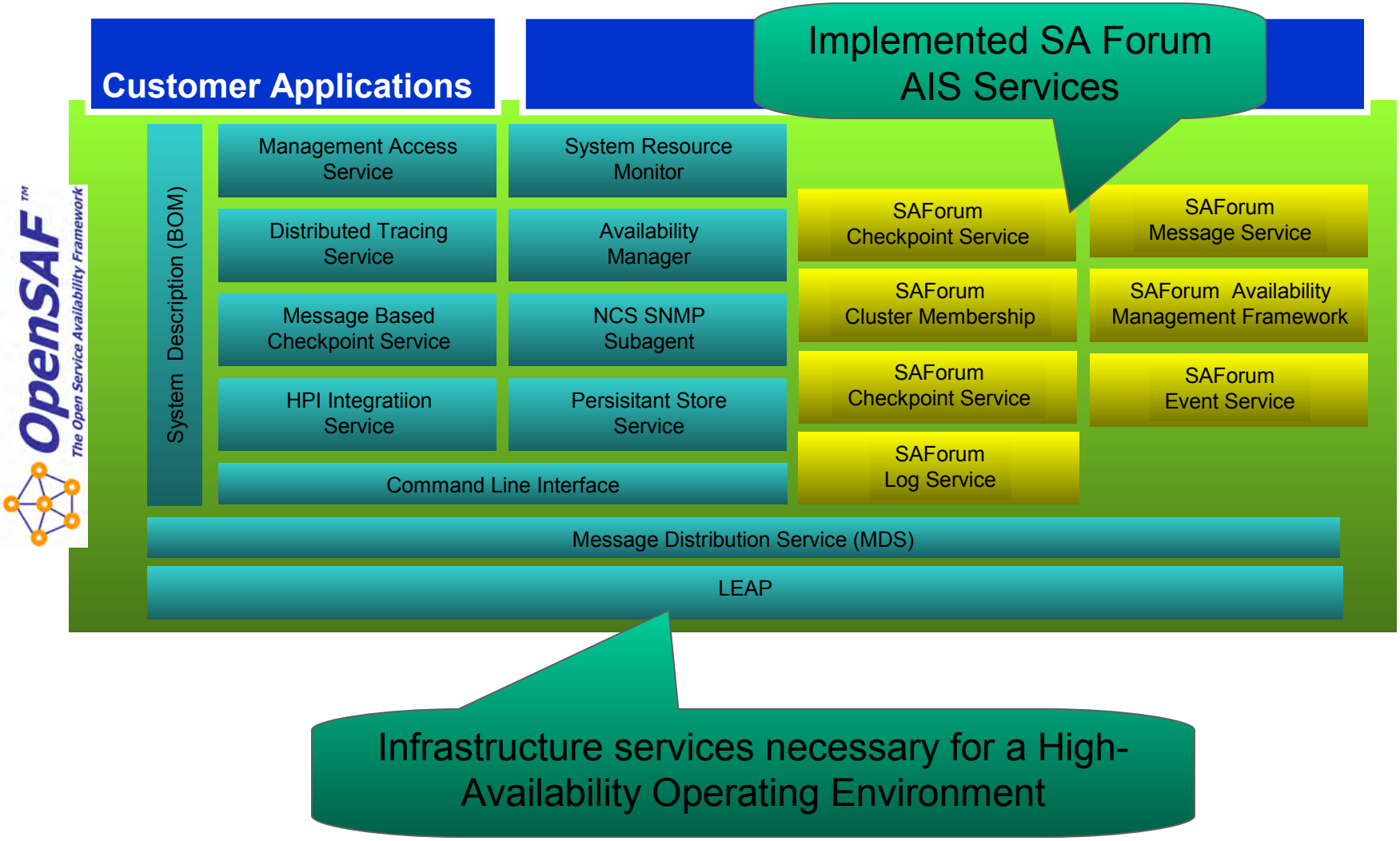
OpenSAF Goals

- *Create a true open source implementation of SAForum standards not tied to any commercial implementation.*

- *Accelerate the development of SAForum specifications by proposing enhancements implemented in the OpenSAF project*

- *Establish a broadly adopted high availability ecosystem that can be leveraged by computing technology companies, NEPs, and other industries requiring high availability, and ISVs.*

Functional Elements of OpenSAF



SAForum AIS Services

- Availability Management Framework
  - High Availability Management Framework
- Cluster Membership Service
  - Maintains and provides cluster membership information
- Message Service
  - Intra cluster message passing ensuring preservation
- Event Service
  - Publish-Subscribe mechanism
- Log Service
  - Log cluster significant events
- Checkpoint Service
  - Record & checkpoint application state data incrementally
- Lock Service
  - Provides distributed locking for shared resources

OpenSAF Infrastructure Services (1)

- Message Based Check-pointing Service
  - Enables low latency failovers and switchovers
- Message Distribution Service
  - High performance message distribution over TIPC
- System Resource Monitor
  - Overload Detection Service
- Availability Manager
  - Interconnects HPI with HA middleware (AIS layer).
- HPI Integration Service
  - Makes HPI events accessible to other OpenSAF services.

OpenSAF Infrastructure Services (2)

- Distributed Tracing Service
  - Tracing and Logging
- Management Access Service
  - System-wide access point for all management operation
- Persistent Store Service
  - Enables low latency failovers and switchovers
- Command Line Interface
  - Enables command line based configuration and administration
- OpenSAF SNMP Agent
  - Record & checkpoint application state data incrementally
- Software Portability using LEAP
  - Interconnects HPI with HA middleware (AIS layer).

Benefits of OpenSAF

- *Application Portability* – Adherence to Open SAForum Specifications provides you the freedom to move your applications around to other HA middleware implementations if you so choose.

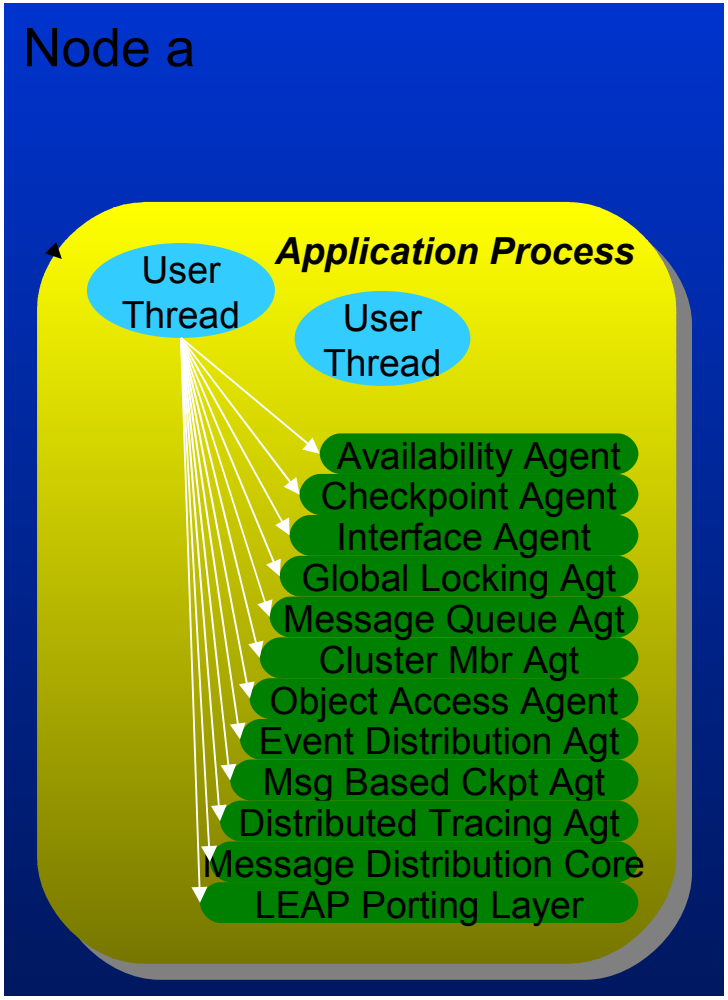
- *Avoid Vendor Locking* – A closed commercial solution locks you to a particular vendor as current SAF specifications only account for 60 – 70 % of what a real HA Telco platform needs. Usage of the other services (that SAF doesn't offer) from a commercial vendor locks you in. With an open solution you have a voice and control on what you lock into.

- Foster an Ecosystem that enables ISVs and 3rd party vendors to pre-integrate their solutions with OpenSAF which enables you to pick off the shelf solution offerings from Database, Protocol stack and other 3rd party software vendors.

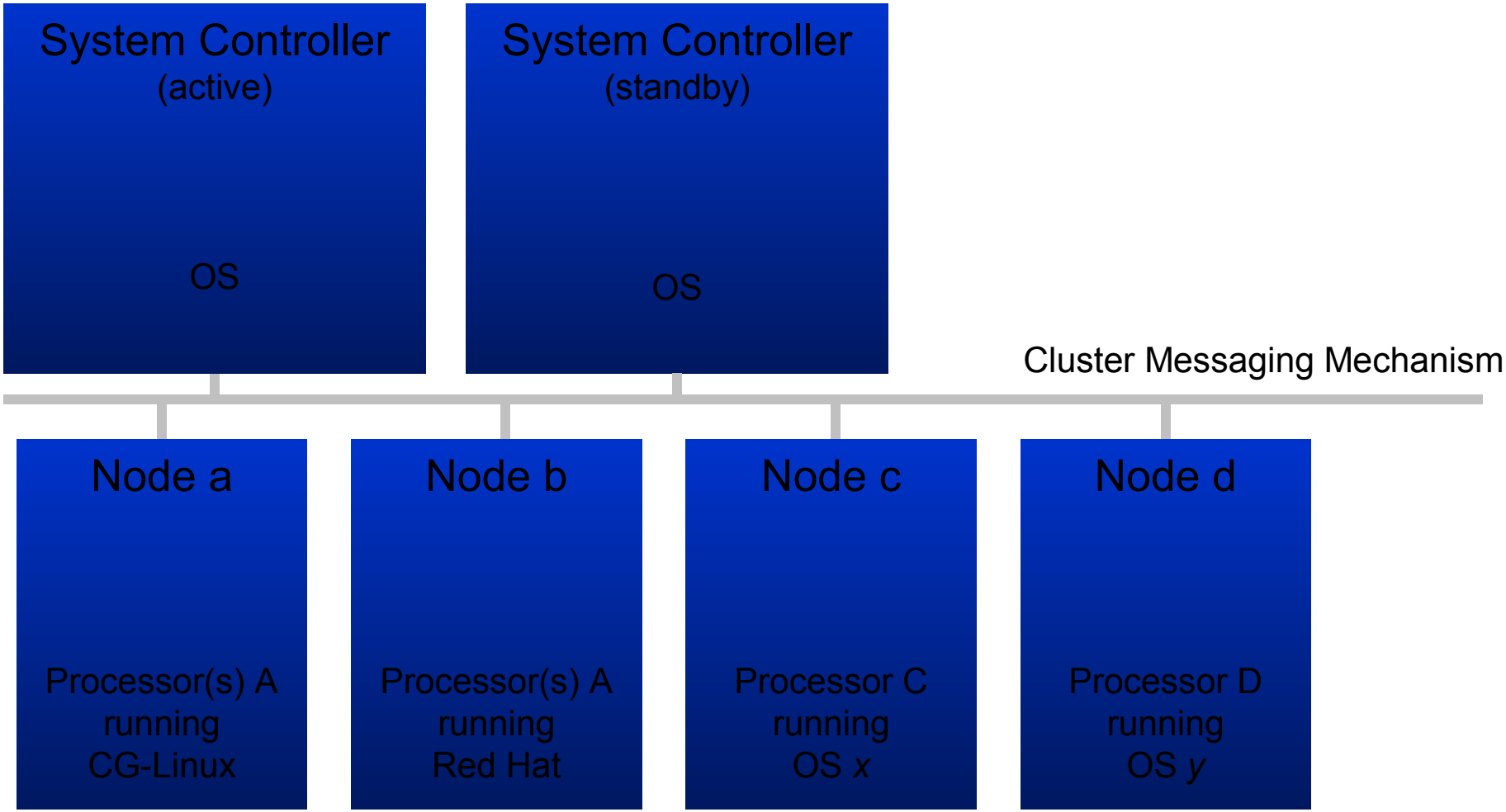
OpenSAF Architecture

Architecture
Application Process

- Applications access OpenSAF services via shared linked libraries called *Agents*
- Use only services desired
- Use LEAP portability or directly access OS



OpenSAF Cluster and Nodes

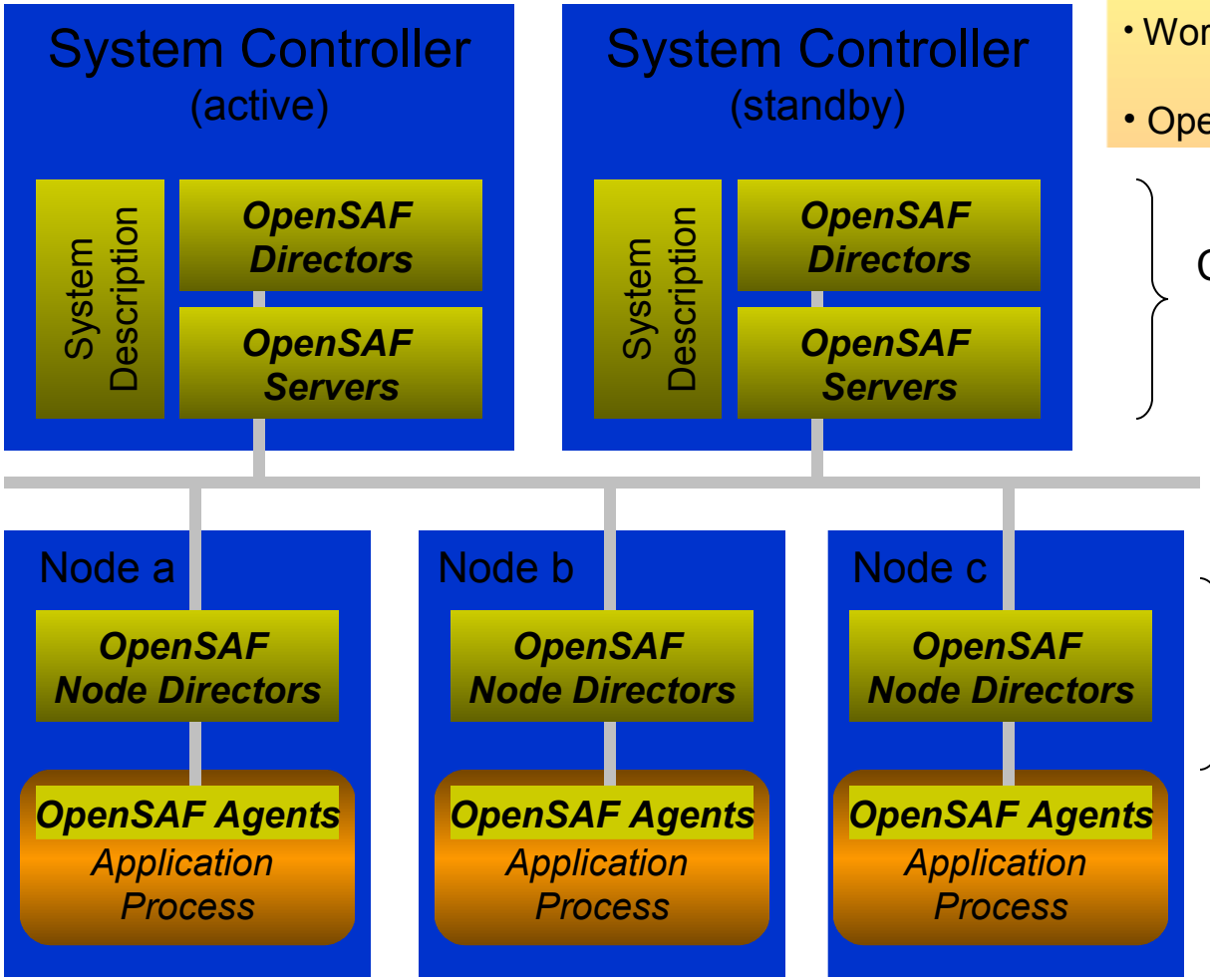


OpenSAF Basic Architectural Paradigms

2-Tier Client Server	➢ SAF EVT Service (EDSv) ➢ OpenSAF Distributed Trace Service (DTSv)
3-Tier Director, Node-Director and Agent	➢SAF AMF ➢ SAF CLM ➢ SAF CKPT ➢ SAF MSG ➢ SAF LCK ➢ OpenSAF Interface Service
Headless peer-to-peer	➢OpenSAF System Resource Monitor Service ➢OpenSAF Message Based Checkpoint Service

3/2-Tier OpenSAF Architecture

Node and System Control



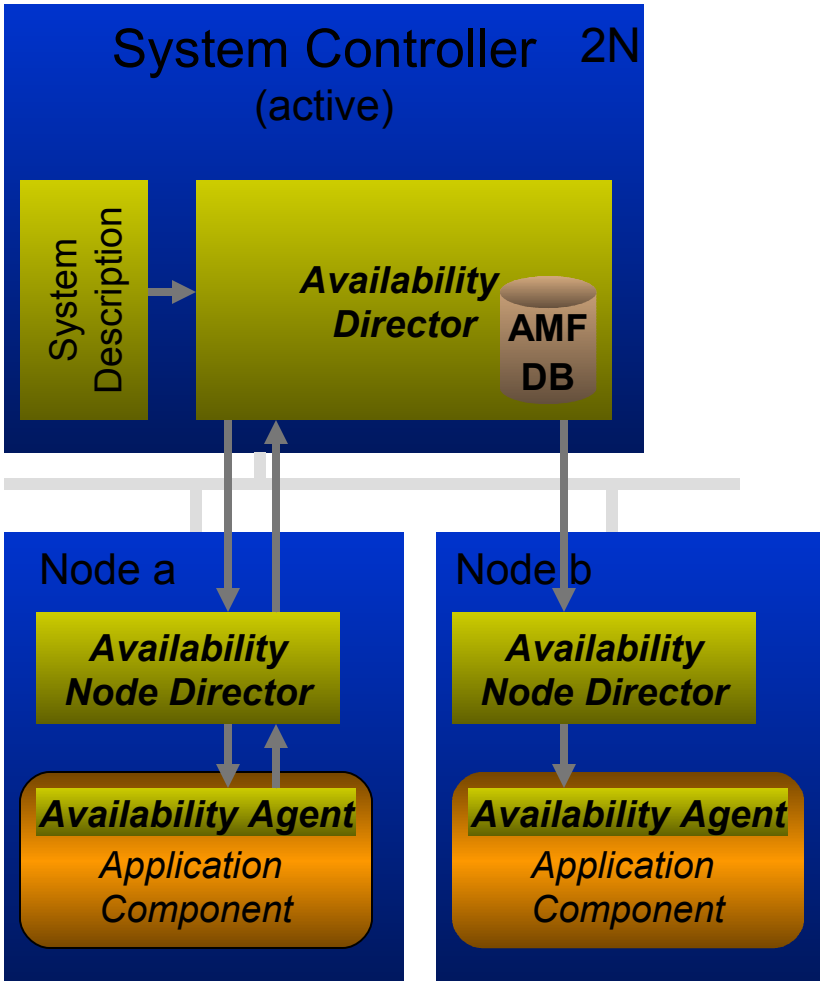
- Directors/servers have cluster wide view
- Work in conjunction with node directors
- OpenSAF configuration is stored here

Centralized System Control

Node Control

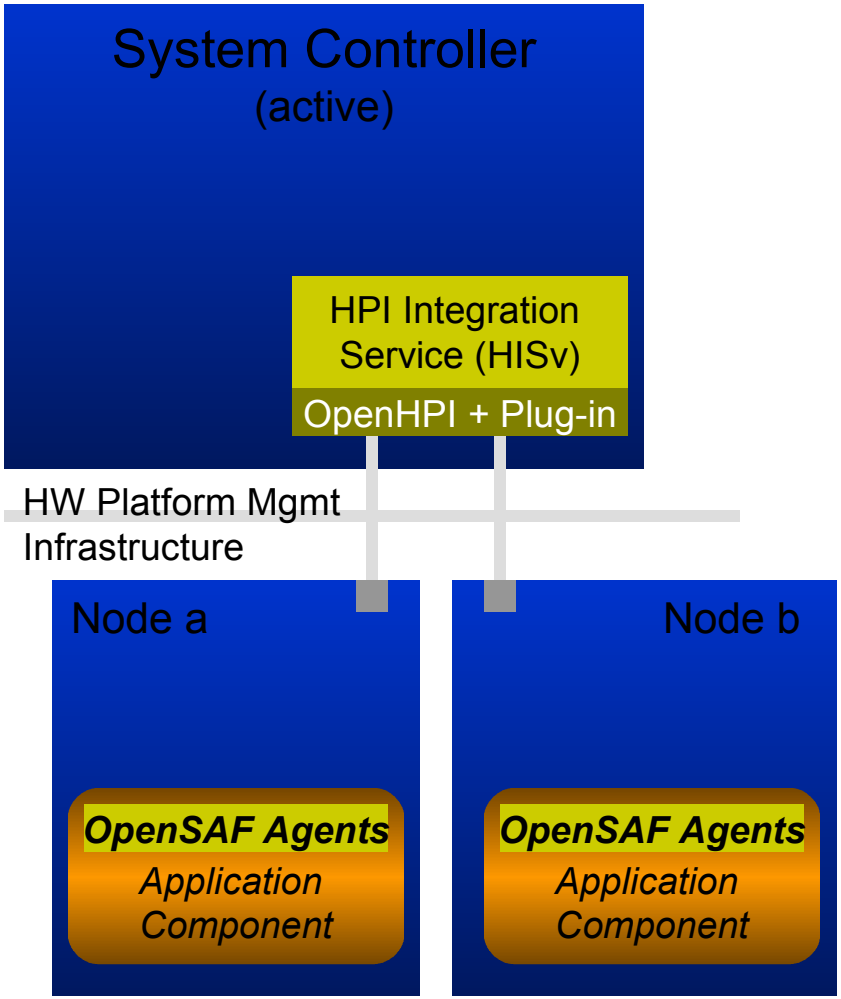
- **Node directors process all events that can be managed at node scope**

AMF Design



1. The Availability Director reads configuration from System Description
2. The Availability Director passes on the node related configuration to the node directors.
3. The node directors manage the life-cycle and health check or the application components.
4. Faults in application are detected by node director and passed on to the director if a failover is needed.
5. The Availability Director orchestrates the failover by sending commands in a sequence to node directors.

HPI Integration Service
Design



1. **OpenSAF assumes the existence of a HW platform management Infrastructure like IPMI used in ATCA platforms.**

3. **One HPI Integration Service is responsible for one HPI domain.**

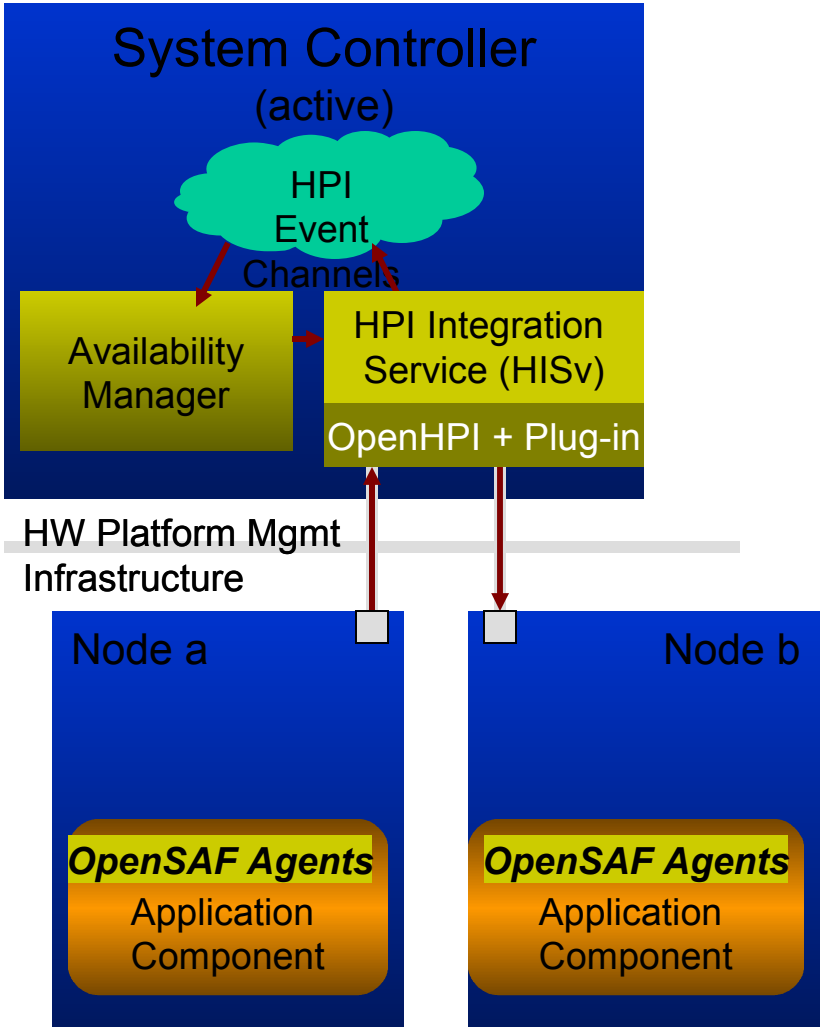
5. **It is linked with OpenHPI and plug-in libraries.**

7. **The plug-in sitting below OpenHPI is responsible for translation to IPMI.**

9. **The HPI Integration Service publishes HPI events on well known event channels.**

11. **Users can access HPI API through HISv Agent.**

HPI Integration Service
How it is used….



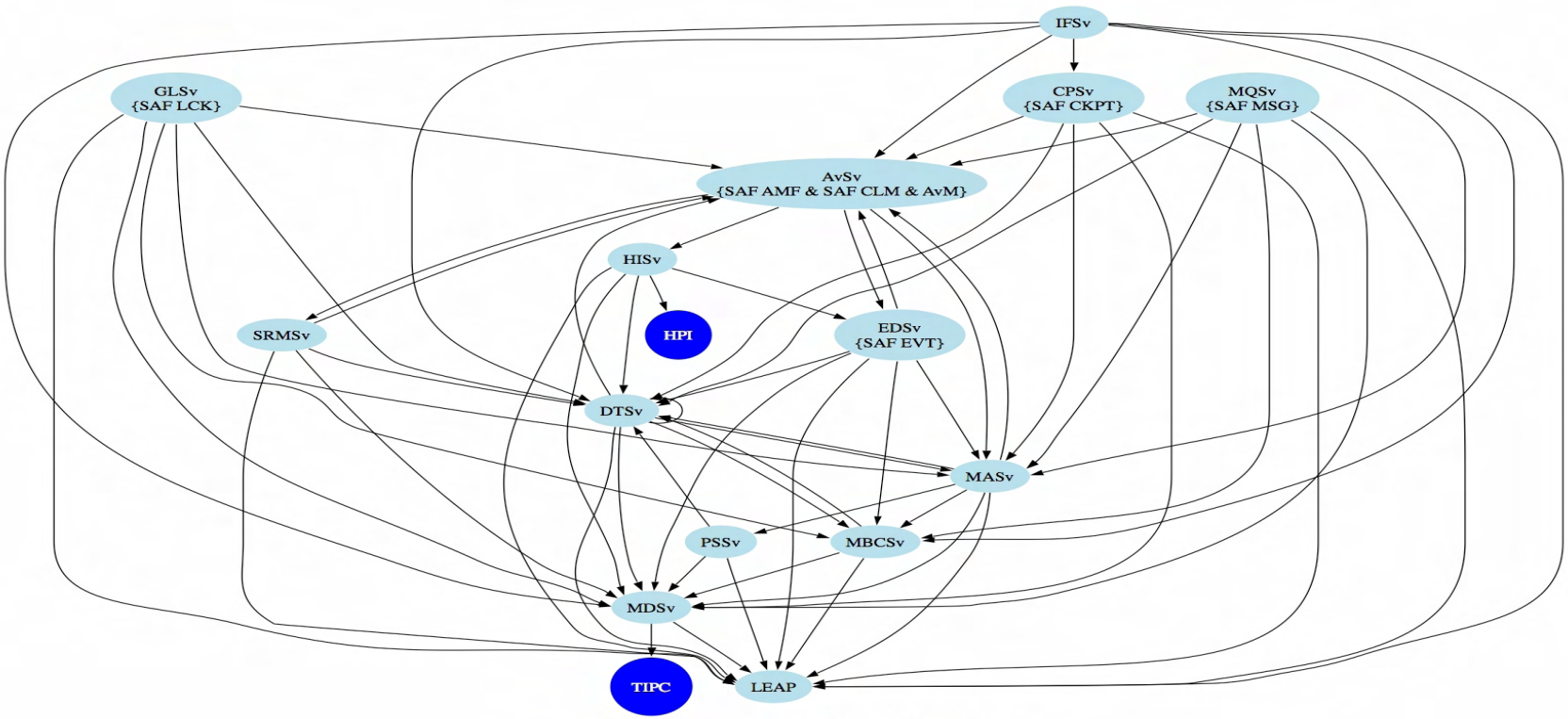
- **HISv receives events through HPI API from the underlying HW platform management infrastructure.**
- **HISv posts the mapped HPI events to certain EVT event channels.**
- **Availability Manager having subscribed for these events, receives them.**
- **Availability Manager being a user of HISv directs HISv to invoke HPI API, say to restart the blade with an entity path (that maps to node b)**
- **The HISv uses OpenHPI library to further execute HPI API to restart Node b.**

HPI Events actually used

HPI Event	OpenSAF	Use-case Context
Hot Swap: Insertion	AvM Service	Validation to ensure that the FRU is admissible
Hot Swap: Pending Extraction	AvM	Fault Domain Analyses to determine fail-over scope and proper shutdown
Hot Swap: Surprise Extraction	AvM	Fault Domain Analyses to determine fail-over scope and orchestrate failovers
Sensor Event: Assertion	AvM	There may be lots of HPI sensors, all capable of issuing an assertion sensor event. If CRITICAL or MAJOR, Do fault domain analysis to determine scope of failure domain; initiate proper failover activity
Sensor Event: Platform	AvM	An asserted Sensor Event may be 'cleared' and reported as a de-asserted Sensor Event. Enable operational state on all effected nodes.

OpenSAF Dependencies

OpenSAF Dependencies



SAF Availability Service (AvSv) which consists of SAF AMF
and SAF CLM and SAF Availability Manager (AvM)
- -
SAF Checkpoint Service (CPSv)
SAF Event Distribution Service (EDSv)
SAF Lock Service (GLSv)
SAF Message Service (MQSv)
Hardware Integration Service (HISv)
Message Based Checkpointing Service (MBCSv)
Interface Service (IFSv)
Management Access Service (MASv)
Persistent Store Service (PSSv)
System Resource Monitoring Service (SRMSv)
Distributed Trace Service (DTSv)
Message Distribution Service (MDSv)

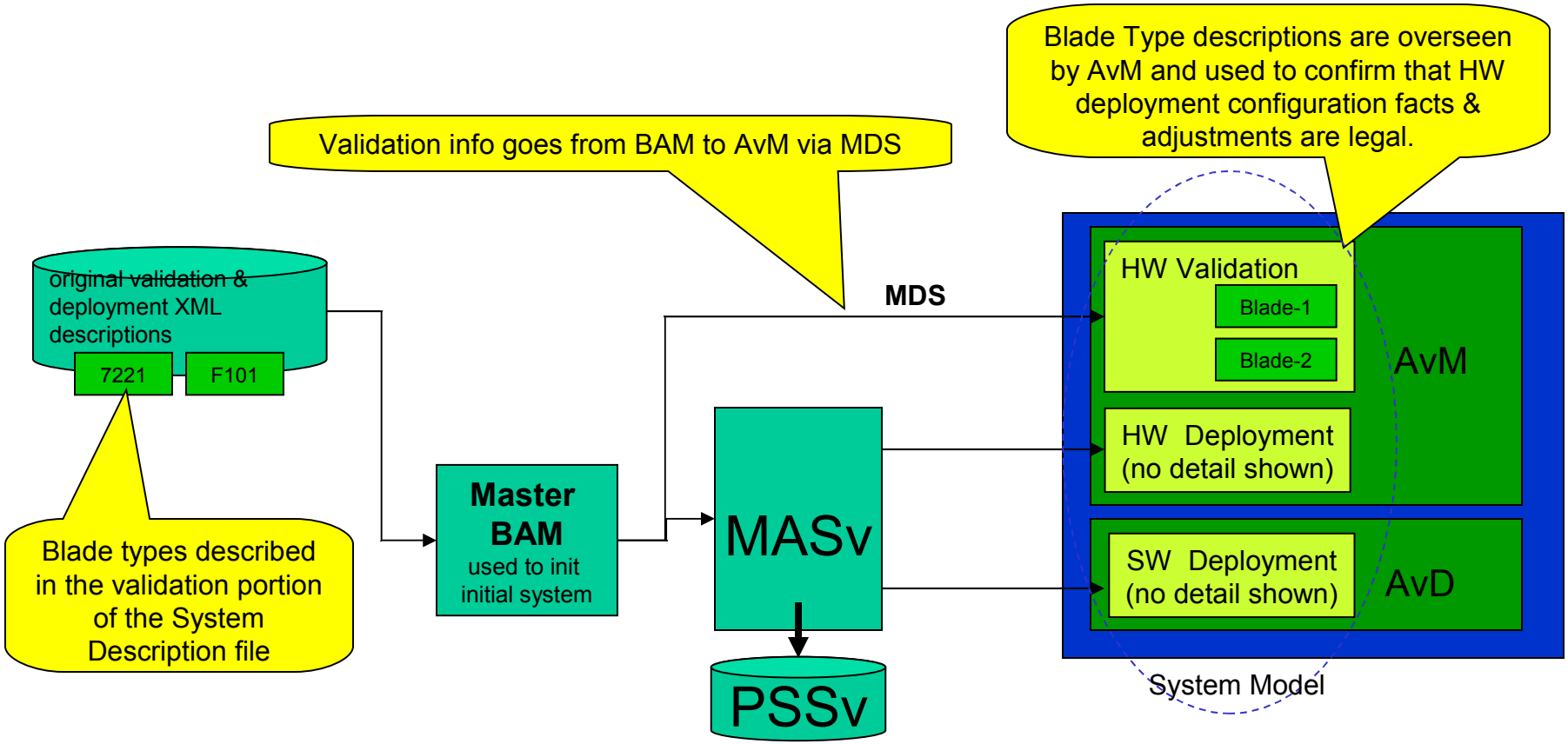
OpenSAF Platform Dependencies

- Role Determination
- Node Location Information (chassis_id, slot_id, node_id)
- File Replication

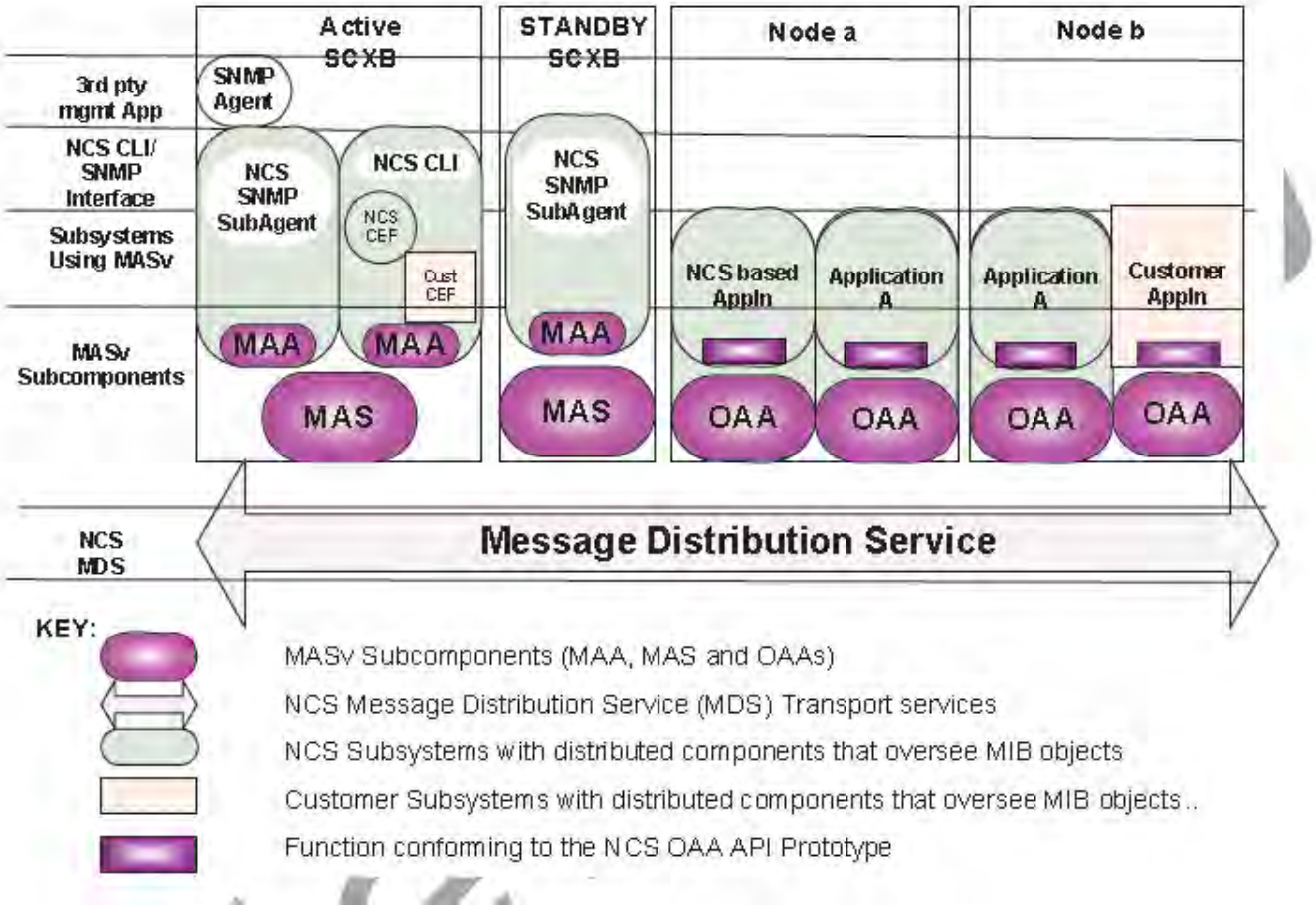
OpenSAF Manageability

System Model consumed at init time….

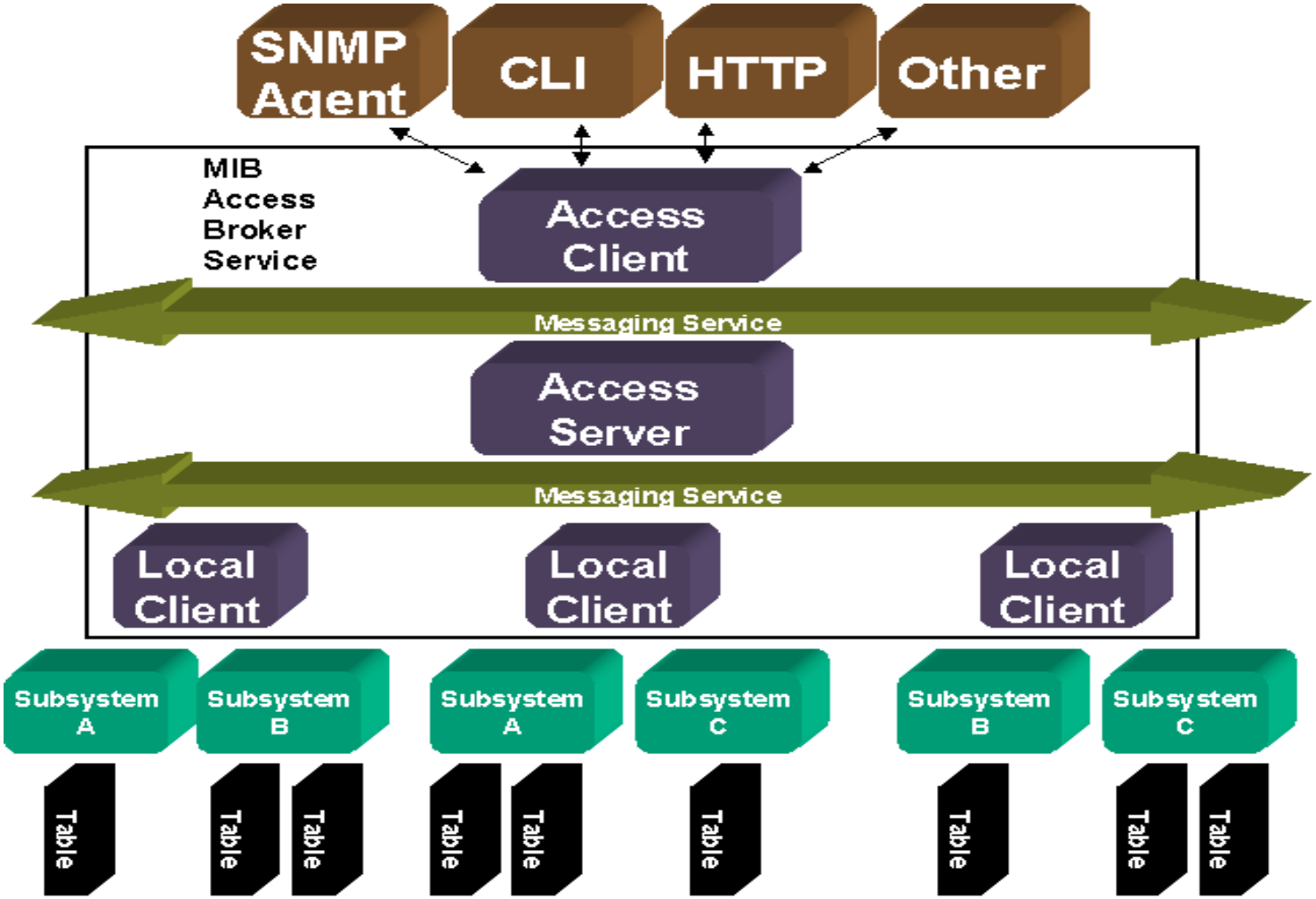
Original system description introduced



OpenSAF Management Access



Management Access Service Logical View



OpenSAF™
The Open Service Availability Framework

Questions ????