MEF Standard
MEF 55.1

Lifecycle Service Orchestration (LSO): Reference Architecture and Framework

January 2021

Table of Contents

List of Figures

List of Tables

1 List of Contributing Members

The following members of the MEF participated in the development of this document and have requested to be included in this list.

- NEC/Netcracker
- Nokia
- Verizon

2 Abstract

LSO is an agile approach to streamlining and automating the service lifecycle in a sustainable fashion for coordinated management and control across all network domains responsible for delivering an end-to-end Service (e.g., Carrier Ethernet as defined in MEF 6.3[11], IP VPN as defined in MEF 61.1[23], Layer 1 as defined in MEF 63[24], SD-WAN as defined in MEF 70[25], Cloud Services, etc.). This document describes a Reference Architecture and Framework for orchestrating the service lifecycle. It includes a set of functional management entities that enable cooperative service lifecycle orchestration for Services. The framework also provides high level functional requirements and outlines high level operational threads describing orchestrated Service behavior as well as interactions among management and control entities. The Management Interface Reference Points that characterize interactions between LSO functional management entities are identified in the reference architecture. These Management Interface Reference Points are described such that they can be realized by Interface Profiles and further by APIs, which can be used to enable automated and orchestrated Services.

3 Terminology and Abbreviations

This section defines the terms used in this document. In many cases, the normative definitions to terms are found in other documents. In these cases, the third column is used to provide the reference that is controlling, in other MEF or external documents.

Term	Definition	Reference
Adagio (ICM:ECM)	The element Management Interface Reference Point needed to manage the network resources, including element view related management functions	This document
Agile	Relating to a Service Provider's ability to rapidly introduce new, on demand services using new technologies without disrupting their top-to-bottom operational environment. Agility can be achieved via proper product / service / resource abstractions using APIs and orchestration.	This document
Allegro (CUS:SOF)	The Management Interface Reference Point that allows Customer Application Coordinator supervision and control of dynamic service of the LSO service capabilities under its purview through interactions with the Service Orchestration Functionality.	This document
Application Program Interface (API)	In the context of LSO, API describes one of the Management Interface Reference Points based on the requirements specified in an Interface Profile, along with a data model, the protocol that defines operations on the data and the encoding format used to encode data according to the data model.	This document
Assured	Relating to the Customer's expectations that a Service will provide consistent performance and security assurances to meet their needs.	This document
Business Applications (BUS)	The Service Provider functionality supporting Business Management Layer functionality.	This document
BUS-partner	Business Applications in the Partner domain	This document
Business Process Flow	Graphically represents the behavior of Process Elements in an "end-to-end" or "through" Process view across the business (i.e., Enterprise).	TMF GB921P [33]
BUS-sp	Business Applications in the Service Provider domain	This document

Term	Definition	Reference
Cantata (CUS:BUS)	The Management Interface Reference Point that provides a Customer Application Coordinator (including enterprise Customers) with capabilities to support the operations interactions with the Service Provider's Business Applications for a portion of the Service Provider service capabilities related to the Customer's Products and Services.	This document
Connectivity Service	A service delivering network connectivity (i.e. traffic) among service access points described by a set of both static and/or dynamic service attributes.	This document
Customer	A Customer is the organization purchasing, managing, and/or using Services from a Service Provider[1]. This may be an end user business organization, mobile operator, cloud operator, or a partner network operator.	This document
Customer Application Coordinator (CUS)	A functional management entity in the Customer domain that is responsible for coordinating the management of the various service needs (e.g., compute, storage, network, etc.) of specific applications.	This document
Data Model	Models managed objects based on an Information Model at a more detailed level using a specific data modeling language. Data modeling languages include XSD, IDL, and YANG.	IETF RFC 3444 [6]
Element Control and Management (ECM)	The set of functionality supporting element management layer capabilities for individual network elements.	This document
Element Management System (EMS)	A management system used to manage the individual network elements as well as the networks that connect them. One or more EMSs may be deployed within a Service Provider management domain depending on the different supplier products and geographic distribution of the network elements in the network.	MEF 7.3 [10]
Ethernet Virtual Connection (EVC)	An association of EVC End Points	MEF 10.4 [12]
Extensible Markup Language (XML)	A markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable.	W3C XML[37]
External Network Network Interface (ENNI)	A reference point representing the boundary between two Operator CENs that are operated as separate administrative domains.	MEF 26.2[16]
Forwarding Construct (FC)	Enabled forwarding between two or more LTPs which supports any transport protocol including all circuit and packet forms.	ONF TR-512.1 [30]

[1] Note that in the MEF Service Standards, the term Subscriber is used.

Term	Definition	Reference
Forwarding Domain (FD)	The topological component which represents the opportunity to enable forwarding between points represented by LTPs.	ONF TR-512.1 [30]
Functional Management Entity	A set of specific management layer functionality within the LSO Reference Architecture.	This document
Hypertext Transfer Protocol (HTTP)	A stateless application-level protocol for distributed, collaborative, hypertext information systems.	IETF RFC 7230 [7]
Information Model	Models managed objects at a conceptual level, independent of any specific implementations or protocols used to transport the data. Information models may be described using UML class diagrams.	IETF RFC 3444 [5]
Infrastructure Control and Management (ICM)	The set of functionality providing domain specific connectivity, application and topology view resource management capabilities including configuration, control and supervision of the infrastructure.	This document
Interlude (SOF:SOF)	The Management Interface Reference Point that provides for the coordination of a portion of LSO services within the partner domain that are managed by a Service Provider's Service Orchestration Functionality within the bounds and policies defined for the service.	This document
Interface Profile	Defines the structure, behavior, and semantics supporting a specific Management Interface Reference Point identified in the LSO Reference Architecture. The Interface Profile specification contains all the necessary information to implement the related API, including objects, attributes, operations, notifications, and parameters.	This document
Internal Network Network Interface (INNI)	A reference point representing the boundary between two networks or network elements that are operated within the same administrative domain.	MEF 4 [9]
JavaScript Object Notation (JSON)	A text format that facilitates structured data interchange between all programming languages.	ECMA-404 [37]
Legato (BUS:SOF)	The Management Interface Reference Point between the Business Applications and the Service Orchestration Functionality needed to allow management and operations interactions supporting LSO Services.	This document
Lifecycle Service Orchestration (LSO)	Open and interoperable automation of management operations over the entire lifecycle of Services. This includes fulfillment, control, performance, assurance, usage, security, analytics and policy capabilities, over all the network domains that require coordinated management and control in order to deliver the Service.	This document

Term	Definition	Reference
LSO Reference Architecture	A layered abstraction architecture that characterizes the management and control domains and entities, and the interfaces among them, to enable cooperative orchestration of Services.	This document
Logical Termination Point (LTP)	Termination point that encapsulates the termination, adaptation and OAM functions of one or more transport layers.	ONF TR-512.1 [30]
Management Abstraction	A management view of information categories and high-level information classes that hides the details of the underlying complexity. LSO identifies Management Abstractions for the Product, Service, and Resource views.	This document
Management Interface Reference Point	The logical point of interaction between specific management entities	This document
Network Control Domain	Represents the scope of control that a particular network controller or WAN controller has with respect to a particular network	This document
Network Domain Controller	Manages the subnetwork boundary edge to subnetwork boundary edge aspects of the network connectivity along with the resources and infrastructure under its control within a specific subnetwork domain.	This document
Network Function Virtualisation (NFV)	The principle of separating network functions from the hardware they run on by using virtual hardware abstraction	ETSI GS NFV 003 [4]
NFV Orchestrator (NFVO)	The functionality that coordinates the management of the connectivity lifecycle, Virtualized Network Functions (VNF) lifecycle, and Network Functions Virtualization Infrastructure (NFVI) resources to ensure an optimized allocation of the necessary supporting resources and connectivity.	ETSI GS NFV-MAN 001 [2]
Object Class	Used to convey the representation of an entity, including behavior, properties and attributes. An instance of an Object Class may be referred to as an Object.	This document
Operational Thread	Describes the high-level Use Cases of LSO behavior as well as the series of interactions among management entities, helping to express the vision of the LSO capabilities. May be further described by a series of detailed use cases spanning a top down approach from Product to Service to Resource.	This document
Operator Virtual Connection (OVC)	An association of OVC End Points	MEF 26.2 [16]

Term	Definition	Reference
Orchestrated	Relating to automated service management across potentially multiple operator networks which includes fulfillment, control, performance, assurance, usage, security, analytics, and policy capabilities, which are achieved programmatically through APIs that provide abstraction from the particular network technology used to deliver the service.	This document
Partner	An organization providing Products and Services to the Service Provider in order to allow the Service Provider to instantiate and manage Service Components external to the Service Provider domain.	This document
PNF	Physical Network Function	ETSI GS NFV 002 [2]
Physical Network Function	A purpose-built network element providing specific network function (s) consisting of a set of software modules deployed on dedicated hardware.	This document
Presto (SOF:ICM)	The resource Management Interface Reference Point needed to manage the infrastructure, including connectivity, application and topology view related management functions.	This document
Process	A systematic, sequenced set of functional activities that deliver a specified result. In other words, a Process is a sequence of related activities or tasks required to deliver results or outputs.	TMF GB921P [33]
Process Element	The building blocks or components, which are used to ‘assemble’ end-to-end business Processes performed in an organization.	TMF GB921P [33]
Product Offering	An externally facing representation of a Service and/or Resource procurable by the Customer.	TMF GB922 [34]
Product Instance	Specific implementation of a Product Offering dedicated to the benefit of a party.	TMF GB922 [34]
Product Lifecycle	The sequence of phases in the life of a Product Offering, including definition, planning, design and implementation of new Product Offerings, changes for existing Product Offerings, and the withdrawal and retirement of Product Offerings.	MEF 50.1 [21]
Product Specification	The detailed description of product characteristics and behavior used in the definition of Product Offerings.	TMF GB922 [34]
Resource	A physical or non-physical component (or some combination of these) within a Service Provider’s infrastructure or inventory.	TMF GB922 [34]

Term	Definition	Reference
SDN Controller	Translates SDN applications' requirements and exerts more granular control over network elements, while providing relevant information up to SDN applications.	ONF TR-504 [30]
Service	Represents the Customer experience of a Product Instance that has been realized within the Service Provider's and / or Partners' infrastructure.	TMF GB922 [34]
Service Component	A segment or element of a Service that is managed independently by the Service Provider.	This document
Service Access Point	The endpoint of a specific Service at a Service Interface (e.g., UNI, ENNI).	This document
Service Interface	A service level demarcation point between administrative domains, including between a Customer and a Service Provider, between two Service Providers, or between internal administrative domains within a single Service Provider. A Service Interface (e.g., UNI, ENNI, INNI) may include a collection of Service Access Points, each representing an endpoint of a specific Service.	This document
Service Level Agreement (SLA)	A contract specifying the service level commitments and related business agreements for a service.	MEF 10.4 [13]
Service Level Specification (SLS)	Technical details of the service level, in terms of Performance Objectives, agreed between the Service Provider and the Customer as part of the Service Level Agreement.	This document; adapted from MEF 10.4 [13]
Service Operations, Administration, and Maintenance (SOAM)	Mechanisms for monitoring connectivity and performance of Services (e.g., Carrier Ethernet).	This document
Service Orchestration Functionality (SOF)	The set of service management layer functionality supporting an agile framework to streamline and automate the service lifecycle in a sustainable fashion for coordinated management supporting design, fulfillment, control, testing, problem management, quality management, usage measurements, security management, analytics, and policy-based management capabilities providing coordinated end-to-end management and control of Services.	This document
Service Specification	The detailed description of the characteristics and behavior of a Service.	TMF GB922 [34]
SOF-partner	Service Orchestration Functionality in the Partner domain	This document
SOF-sp	Service Orchestration Functionality in the Service Provider (SP) domain	This document

Term	Definition	Reference
Software Defined Networking (SDN)	An architecture that provides open interfaces that enable the development of software that can control the connectivity provided by a set of network resources and the flow of network traffic though them, along with possible inspection and modification of traffic that may be performed in the network.	ONF TR-504 [30]
Sonata (BUS:BUS)	The Management Interface Reference Point supporting the management and operations interactions (e.g., ordering, billing, trouble management, etc.) between two Operators, e.g., Service Providers and Partners).	This document
Topology and Orchestration Specification for Cloud Applications (TOSCA)	A specification defining the structure, properties and behavior expressed by TOSCA Service Templates	TOSCA OASIS BPEL2.0 [26]
TOSCA Service Template	The combination of a TOSCA Topology Template and TOSCA Plans (or Orchestration processes). In this document, a TOSCA Service Template can be used for Products or Services or Resources in MEF LSO	TOSCA OASIS BPEL2.0 [26]
TOSCA Topology Template	A TOSCA Topology Template (also referred to as the topology model of a service) defines the structure of a service. It consists of a set of TOSCA Node Templates and TOSCA Relationship Templates that together define the topology model of a service as a (not necessarily connected) directed graph.	TOSCA OASIS BPEL2.0 [26]
TOSCA Node Type	A TOSCA Node Type defines the properties and the operations available to manipulate a component of a service.	TOSCA OASIS BPEL2.0 [26]
TOSCA Node Template	A node in a topology graph is represented by a TOSCA Node Template. A TOSCA Node Template specifies the occurrence of a TOSCA Node Type as a component of a service.	TOSCA OASIS BPEL2.0 [26]
TOSCA Relationship Type	A TOSCA Relationship Type defines the semantics and any properties of the relationship between TOSCA Nodes.	TOSCA OASIS BPEL2.0 [26]
TOSCA Relationship Template	A TOSCA Relationship Template specifies the occurrence of a relationship between nodes in a TOSCA Topology Template.	TOSCA OASIS BPEL2.0 [26]

Term	Definition	Reference
TOSCA Plans	TOSCA Plans define the (Orchestration) process models that are used to create and terminate a service as well as to manage a service during its whole lifetime. In this document, TOSCA Plans are similar to the term Business Process Flows in some entities, e.g., SOF or ICM, of LSO RA.	TOSCA OASIS BPEL2.0 [26]
Unified Modeling Language (UML)	A general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.	OMG UML[29]
Use Case	In UML, a Use Case represents one particular type of a system's behavior based on stimuli from an external source (i.e., an actor). A system may have several Use Cases that define all its behavior.	OMG UML[29]
User Network Interface (UNI)	The demarcation point between the responsibility of the Service Provider and the responsibility of the Customer.	This document; Adapted from MEF 11[14]
Virtual Network Element (VNE)	An abstraction representing a set of network functions providing network element capabilities implemented in a virtualized environment.	This document
VNF	Virtual Network Function	ETSI GS NFV 002[3]
Virtual Network Function	A network function that is provided through software virtualization techniques.	This document

Table 1 – Terminology and Abbreviations

4 Scope

The purpose of this document is to define a reference architecture that describes the functional management entities needed to support LSO, and the Management Interface Reference Points between them. LSO provides open and interoperable automation of management operations over the entire lifecycle of Services. This includes design, fulfillment, control, testing, problem management, quality management, billing and usage, security, analytics and policy capabilities, over all domains that require coordinated management and control in order to deliver the service. The reference architecture characterizes the management and control domains and entities that enable cooperative LSO capabilities for Connectivity and Cloud Services. The LSO architecture and framework enables automated management and control of Connectivity and Cloud Services that span multiple operator domains. For example, a Service Provider may extend its footprint by using LSO to interact with potentially several Operators to manage and control components of the end-to-end services including access portions.

The framework also outlines high level operational threads providing business rationale and describing orchestrated service behavior as well as interactions among management and control entities. This document describes the essential LSO capabilities for Services that need to be supported by the common product, service, and resource abstractions and constructs. Such constructs will drive the information and data models that enable the definition of open and interoperable APIs supporting LSO functionality. The reference architecture work will also be cross referenced with the efforts of other Standards Development Organizations (SDOs) and open-source projects (e.g., ONF, ETSI, IETF, TMF, OPNFV, ODL, etc.).

This framework also describes the engineering approach being followed to generate re-usable engineering specifications and artifacts capturing the LSO requirements, capabilities, functionality, behavior, processes, information, interfaces and APIs supporting management and control of Services.

5 Compliance Levels

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 (RFC 2119 [8], RFC 8174 [9]) when, and only when, they appear in all capitals, as shown here. All key words must be in bold text.

Items that are **REQUIRED** (contain the words **MUST** or **MUST NOT**) are labeled as **[Rx]** for required. Items that are **RECOMMENDED** (contain the words **SHOULD** or **SHOULD NOT**) are labeled as **[Dx]** for desirable. Items that are **OPTIONAL** (contain the words **MAY** or **OPTIONAL**) are labeled as **[Ox]** for optional**.**

6 Introduction

LSO provides orchestration capabilities for the open and interoperable management and control of Services such as Connectivity Services or Cloud Services. The LSO Reference Architecture characterizes the management and control domains and entities that enable cooperative LSO capabilities. This architecture also outlines high level operational threads describing orchestrated Service behavior as well as interactions among management entities. LSO overcomes existing complexity by defining product, service, and resource abstractions that hide the underlying complexity from users of the services.

In this document, Section 7 discusses the LSO engineering methodology. The high-level functional requirements for LSO functional management entities are provided in Section 8. Section 9 provides the LSO Reference Architecture that characterizes the management and control domains and functional management entities that enable cooperative LSO capabilities. High level Operational Threads describing the use cases for LSO behavior are identified in Section 10. LSO Management Abstractions and constructs are described in Section 11. References may be found in Section 12. Appendix A provides an informative appendix with examples of high-level interactions per LSO management interface reference point. Appendix B describes the relationship of LSO functional areas to MEF 50 [20] processes. Finally, Appendix C describes TOSCA service templates defining the relationship between Customers, Service Providers (SPs) and developers of IT services.

6.1 MEF 3.0 Vision

The communications industry is in the midst of a multi-year transformation to dynamic and assured services across a global ecosystem of automated networks, as envisioned in the MEF 3.0 framework (MEF 3.0 Overview [9]),

There is a growing consensus that Service Providers must become more cloud-like and automated to deliver powerful networking solutions that provide unprecedented user- and application-directed control over network resources and service capabilities. However, it is not enough for providers to offer dynamic connectivity and virtualized services over just their own networks. Service Providers must transition from operating as independent islands to being integral players in a worldwide business federation supporting dynamic services across multiple operators. To realize this vision, MEF has advanced standardization of a full family of MEF 3.0 services – including Carrier Ethernet, SD-WAN, Optical Transport, IP, Cloud Services, and Application Security for SD-WAN – and progressed standardization of LSO APIs that are foundational to orchestrating these services across federated networks in addition to multiple technology domains within provider networks.

This approach overcomes existing constraints by defining service abstractions that hide the complexity of underlying technologies and applications from users of the services, while providing sufficient management and control capabilities.

6.2 Lifecycle Service Orchestration

Since Connectivity and Cloud Services are agile, assured, and orchestrated, they rely on coordinated orchestration of distributed capabilities across potentially multiple internal networks, multiple Connectivity Operators and multiple Cloud Operators to enable end-to-end management. Such orchestration is executed for the entire Service lifecycle where each functional area of the lifecycle is further streamlined and automated, from Product Offering definition through service fulfillment, control, assurance, and billing MEF 50.1 [21]. For example, the fulfillment phases of the service lifecycle are focused on automating the inter-provider business interactions and interfaces for the buyer-seller process, including the product catalog, order, service location, and service qualification. Each of these phases is based on the Product Offering defined by the selling operator. Since the Product Offering is fully defined in the product catalog, a model-driven approach is used to execute the subsequent stages of the service lifecycle, including pre-order, order, and service orchestration. By using a model-driven approach along with abstractions representing products, services, and resources, LSO ensures an agile approach to streamlining and automating the entire service lifecycle in a sustainable fashion.

In LSO, Connectivity and Cloud Services are orchestrated by Service Providers across all internal and external domains from one or more operators. These domains may be operated by Connectivity Operators (e.g., wireless network operators, IP/MPLS network operators) or Cloud Operators (e.g., content providers, data center operators). LSO encompasses all domains that require coordinated end-to-end management and control to deliver Services. LSO capabilities not only dramatically decrease the time to establish and modify the characteristics of the Service, but also assure the overall service quality and security for these services.

7 LSO Engineering Methodology

The primary goal of the LSO engineering methodology being followed by MEF is the generation of re-usable engineering specifications and artifacts capturing the LSO requirements, capabilities, functionality, behavior, processes, information, interfaces and APIs supporting management and control of Services. These engineering artifacts are valuable resources in enabling the transformation of LSO capabilities into interoperable, specific, consistent, and verifiable designs and implementations. Each of these stages of the agile LSO Engineering Methodology illustrated in Figure 1 is discussed in more detail the subsequent sub-sections. In the figure, the solid arrows describe that the artifacts produced during one stage are consumed during subsequent stages, while the dotted back arrowed lines represent relevant feedback to the prior stages.

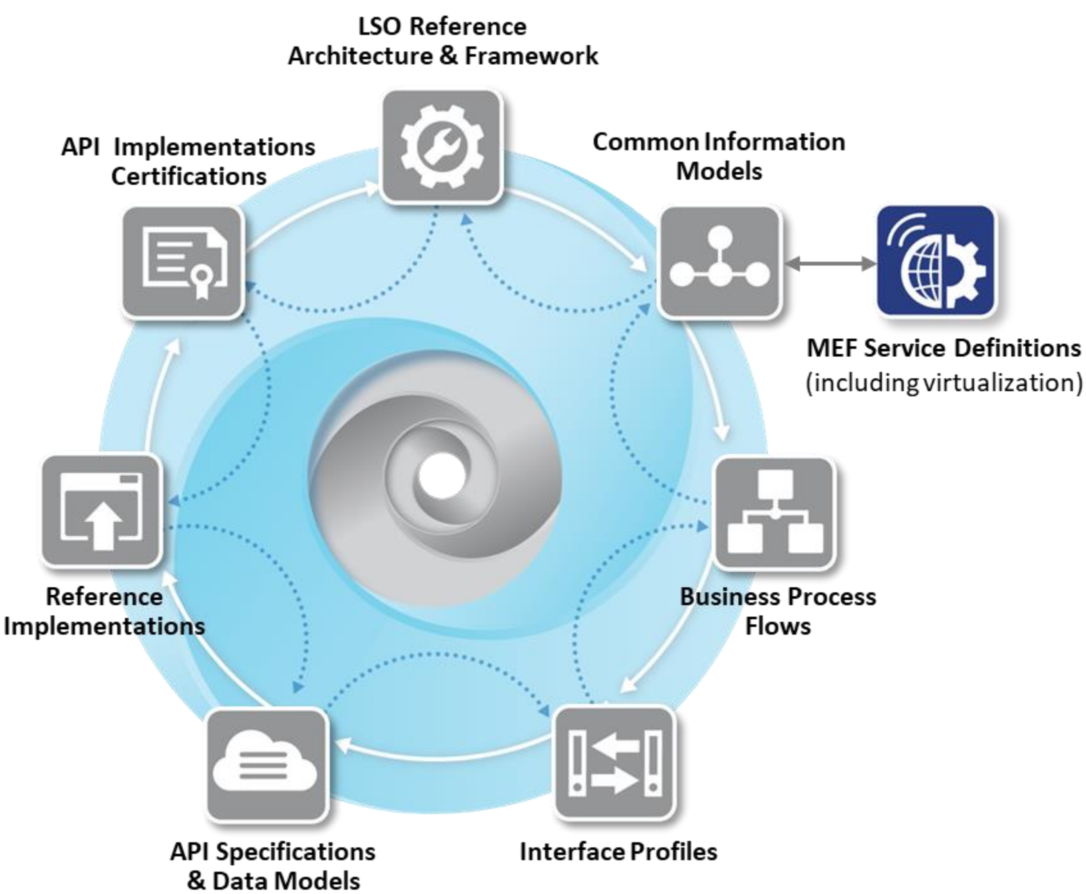


Figure 1 – LSO Engineering Methodology

7.1 LSO Reference Architecture and Framework

The LSO Reference Architecture and Framework, specified in this document, provides a layered architecture that characterizes the management and control domains and entities that enable cooperative LSO capabilities for Services. The framework also describes the high-level management requirements and outlines high level operational threads. Operational threads describe orchestrated service behavior as well as interactions among management entities spanning the Customer, Service Provider, and Partner management domains, expressing the vision of MEF LSO capabilities. Within the LSO Reference Architecture, a Management Interface Reference Point is the logical point of interaction between specific management entities. The Management Interface Reference Points that characterize interactions between LSO functional management entities are identified in the reference architecture. These Management Interface Reference Points are further defined by Interface Profiles and implemented by APIs and Reference Implementations that realize automated and orchestrated Services. An Interface Reference Point may be described by a number of Interface Profiles, each addressing a specific functional scope. Artifacts from the LSO Reference Architecture and Framework are used by the subsequent stages in the LSO Engineering Methodology. Lessons learned from API certification may be used to update the LSO Reference Architecture.

As a specification the LSO Reference Architecture and Framework:

- Describes the LSO Engineering Methodology (Section 7);
- Provides high level requirements associated with LSO functional areas (Section 8);
- Defines the LSO Reference Architecture (Section 9);
- Outlines operational threads for LSO (Section 10); and
- Identifies the LSO Management Abstractions and constructs (Section 11).

7.2 Information Models

Information Models define managed objects at a conceptual level, independent of any specific implementations or protocols used to transport the data. The shared common information models for LSO supporting Services include the service attributes defined in MEF Standards defining a common set of consistent managed object definitions for managing the service lifecycle. These common management and control Information Models support management of Products, Services, and Resources.

Common information models help ensure that management and control functionality, and information shared among different functions and elements, are provided in a logically consistent fashion, allowing network operators to readily integrate such capabilities into their Service management and control environment. These functions and elements could include Customer relationship management, service management, resource management, and supplier / partner management functions, as well as orchestrators, infrastructure managers, controllers (e.g., Network Domain Controllers, SDN controllers, VNF Managers, etc.), and Network Elements (NEs).

MEF defines management Information Models supporting LSO, that describe the information associated with the generalized management interactions using protocol neutral Unified Modeling Language (UML) and TOSCA Service Templates. Artifacts from the Information Models are used by the subsequent stages in the LSO Engineering Methodology, including the definition of Interface Profiles. Lessons learned from Information Models may be used to update the Management Abstractions in LSO Reference Architecture.

7.3 Business Process Flows

The details of the high-level Operational Threads outlined in the LSO Reference Architecture and Framework are further expanded into more detailed Business Process Flows or TOSCA Plans. Business Process Flows describe the functional activity flows among and within organizations along with information exchanges between management and control entities based on the information models. A process describes a systematic, sequenced set of functional activities that deliver a specified result. MEF LSO enables automation of the related Business Processes that operationalize Services. In this model driven approach, the business models (including process models and associated information exchanges with external entities) help drive the Interface Profile design. Artifacts from the Business Process Flows are used by the subsequent stages in the methodology, including the definition of Interface Profiles to support process interactions. Lessons learned from Business Process Flows may be used to update the Information Models.

7.4 Interface Profiles

An Interface Profile is the protocol neutral functional description that defines the structure, behavior, and semantics supporting a specific Management Interface Reference Point identified in the LSO Reference Architecture. A single Interface Reference Point may be described by a number of Interface Profiles, each addressing a specific functional scope. An Interface Profile describes information views and interactions by identifying a subset of object classes, properties, and capabilities (e.g., write, read, etc.) necessary to support each interface view, or information model fragment, relevant to a Management Interface Reference Point based on the Information Models as well as other relevant standards. The Interface Profiles provide a step in MEF LSO engineering methodology that will supply the logical requirements for language specific (e.g., YANG, XSD, etc.) management data models and APIs. The selection of specific data modeling languages and encodings may be restricted by the Interface Profile. Also, an Interface Profile may identify and constrain the application of specific API definitions developed by other SDOs. Artifacts from the Interface Profiles are used by the subsequent stages in the LSO Engineering Methodology, including the definition of API Specifications. Lessons learned from Interface Profiles may be used to update the Information Models and Process Flows. Bottom up feedback to realign Information Model and Interface Profiles provides alignment and consistency.

7.5 API Specifications and Data Models

This section describes the LSO engineering phase where API specifications and their associated data models are defined based on the requirements in the Interface Profiles. Interface Profiles provide requirements for the API which may be implemented using a variety of data models and encodings while retaining semantic consistency. More than one API may be defined to instantiate a management interface described in an Interface Profile. Knowledge garnered from this phase will feed back into the Interface Profiles as well as the reference architecture and framework.

An API specification defines how software components should interact with each other. In the context of LSO, an API is the realization of an Interface Profile for a specific Management Interface Reference Point. The information exchanged across an API is described within a data model that is specified in a data modeling language, for example YANG or XSD. Such a data model defines the structure of data that is conveyed between the two management entities that bound the Management Interface Reference Point.

An API also defines the encoding format (e.g. JSON or XML) that is used to encode data into a representation and format that can be exchanged across the interface according to the structure described by the data model, and the protocol that is used to carry the encoded interface data (e.g. NETCONF, RESTCONF or REST/HTTP). The protocol, along with the data model, also defines the operations that are supported - for example, creating and deleting persistent managed objects, reading and writing attributes of those objects, etc.

Note that in the context of LSO, an API does not constrain the implementation of either management entity to a particular programming language; it simply describes the format and semantics of messages passed between them.

7.6 API Reference Implementations

This section discusses the development of reference implementations based on the API specifications. API Reference Implementations are MEF developed management protocol specific implementations of interfaces providing the functions and information exchanges that implement Management Interface Reference Points in the LSO reference model. Such Reference Implementations help to accelerate the development of open and interoperable APIs that will bring about the realization of LSO. API Reference Implementations are based on the functional requirements described in an Interface Profile and defined in the associated API Specification. MEF API Reference Implementations may apply MEF standards as well as specifications of partnering SDOs. To help accelerate the development of LSO API Reference Implementations, MEF sponsors events such as LSO Hackathons.

7.7 API Implementation Certification

MEF has unique positioning in the industry with MEF service-oriented certification, and continue to do so to support the LSO Vision of MEF 3.0. API Implementations are essential for the realization of LSO APIs and may be incorporated in MEF certification programs that verify the LSO related API Implementations including data exchange formats and behavior. Also, experience gained during certification may be used to improve or extend the LSO Reference Architecture.

8 High Level Management Requirements

This section describes the high-level functional requirements for LSO functional management entities (see Section 9.1), including the Service Orchestration Functionality (SOF). Interface reference point specific application of the functionality described in this section will be addressed in subsequent documents. The service lifecycle addresses each functional area from Product Offering definition through service fulfillment, control, assurance, and billing MEF 50.1 [21].

8.1 Agile Product / Service Design

Product and Service development lifecycle management agility is supported by LSO with its abilities to rapidly model or import modular model specifications from different layers of abstractions such as Product Offering, Product, Service, Service Component, and Resource. The static and dynamic relationships among layers of model abstractions need to be represented, along with their behaviors (such as design and assign policies) and actions (such as create, modify, test, etc.).

Requirements:

[R-LSO-RA-1]:LSO **SHALL** support the product lifecycle management process (i.e., as defined in MEF 50.1[21])

[R-LSO-RA-2]:LSO **SHALL** maintain catalog capabilities in support of:

- Product Specifications (from which Product Offerings are defined and exposed in a product catalog)
- Service Specifications (for the Service and each Service Component)
- Product Instance to Service mapping rules for each Product Offering
- Service design and policy assignment

8.2 Order Fulfillment and Service Control

Order Fulfillment and Service Control support the orchestration of provisioning related activities involved in the fulfillment of a Customer order or a service control request, including the tracking and reporting of the provisioning progress. This breaks down into multiple functional orchestration areas:

- <u>Order Fulfillment Orchestration:</u> deals with decomposing a customer order into one or multiple service provisioning activities and orchestrating of all customer order-related fulfillment activities;
- <u>Service Configuration and Activation Orchestration</u>: responsible for the design, assignment, and activation activities for the end-to-end service and/or some or all Service Components;
- <u>Service Control Orchestration:</u> permits the service to be dynamically changed within specific bounds described in policies that are established in advance;

- Service Delivery Orchestration: responsible for the service delivery via network implementation delegation of each Service Component to their respective delivery system or mechanism; and
- Service Activation Testing Orchestration (see Section 8.3): coordinates all service activation testing activities, for parts and/or the complete end-to-end service.

8.2.1 Order Fulfillment Orchestration

Order Fulfillment Orchestration is triggered from a Customer order, generally originating from a business application such as a customer relationship management system or order entry system. This set of functionality will deliver an order initiated rapid on-demand customer experience provided all activities are automated. Its responsibilities include, but are not limited to:

- Scheduling, assigning and coordinating customer provisioning related activities;
- Generating the respective service creation / modification / move / deletion request(s) based on specific Customer orders;
- Undertaking necessary tracking of the execution process;
- Adding additional information to an existing customer order under execution;
- Modifying information in an existing customer order under execution;
- Canceling a Customer order when the initiating sales request is cancelled;
- Monitoring the jeopardy status of customer orders (e.g., order is in danger of not meeting its expected completion date), and escalating customer order status as necessary in accordance with local policy;
- Instantiating, when appropriate, an event for the billing system; and
- Indicating completion of a customer order by modifying the Customer order status.

Requirements:

[R-LSO-RA-3]:LSO **SHALL** be able to decompose a Customer order into one or multiple service provisioning activities (such as multiple service requests), and orchestrate these provisioning activities.

[R-LSO-RA-4]:LSO **SHALL** ensure Customer order related provisioning activities are assigned, managed and tracked efficiently to meet the agreed or estimated committed availability time or date.

Note that LSO should enable staggered billing per site, for example, in cases where one or more sites, in a multi-site customer order, were to get into exception/fall-out stages (e.g., not meeting expected completion date) for a long duration or require longer duration manual activities.

[R-LSO-RA-5]:LSO **SHALL** be able to receive a completed Customer order, with content based on a Product Offering and definition within a product catalog.

[D-LSO-RA-1]:LSO **SHOULD** be able to orchestrate diverse product-related activities, based on an incoming Customer order (which may comprise many dependent and independent order lines), such as initiating the billing process, coordinating supply chain management for delivery of a purchased CPE, coordinating with other service fulfillment systems within the Service Provider's domain, etc.

[D-LSO-RA-2]:LSO **SHOULD** support customer order revisions (add or modify order elements, such as adding a new site to the Customer order, or modifying a site bandwidth) in case they are submitted against an order which is still in progress.

[D-LSO-RA-3]:LSO **SHOULD** support customer order cancellation, including rollback, intercepting the order fulfillment execution.

[R-LSO-RA-6]:LSO **SHALL** be capable of orchestrating business and operations support systems (e.g., billing and revenue management, customer relationship management, fault management, and performance / SLA management).

[R-LSO-RA-7]:LSO **SHALL** undertake necessary tracking of the execution process, dynamically modify and report the Customer order status, and monitor the jeopardy status of Customer orders, escalating Customer orders as necessary.

8.2.2 Service Configuration and Activation Orchestration

At a high level, the Service Configuration and Activation Orchestration is responsible for the design of the end-to-end Service, including the selection and routing of the Service over the involved domains (e.g., Forwarding Domains) and the Service Component parameters, as well as the calculation of the list of technical actions (i.e., "delivery orchestration plan" or plan of tasks necessary to instantiate the Service) that must get executed for the implementation of the Service. Specifically, Service Configuration and Activation Orchestration encompasses allocation, design, and configuration of specific Services or Service Components in support of Product Instances to meet Customer requirements, or in response to requests from other processes to alleviate specific service capacity shortfalls, availability concerns or failure conditions. In support of Service Configuration and Activation Orchestration, LSO applies details from the Product Offering and the Customer Order to design the end-to-end Service, and identifies the Service Components composing the Service. Network and Application Domain Controllers design and configure each Service Component within their domain.

Responsibilities of the Service Configuration and Activation Orchestration include, but are not limited to:

- Verifying whether specific Service Request sought by Customers are feasible;
- Decomposition of the Service into Service Components;
- Allocating the appropriate specific Service parameters within each Service Component to support Service requests, control requests, or requests from other processes;
- Reserving specific Service-related resources (if needed) for a given period of time until the initiating Customer order is confirmed, or until the reservation period expires (if applicable);
- Configuring the specific Service, as appropriate;
- Recovery of the specific Service;
- Updating of the Service state information to reflect that the specific Service has been allocated, modified or recovered;
- Assigning and tracking Service Component provisioning activities;

- Managing Service provisioning jeopardy conditions (e.g., conditions that add to the risk of missing a confirmed due date or activity required to continue processing the Service Request, such as: capacity is not available, capability is not supported, etc.); and
- Tracking progress on Service configurations and activations.

Requirements:

[R-LSO-RA-8]:LSO **SHALL** be able to determine the necessary Service Components and configurations needed to support a Service.

[R-LSO-RA-9]:LSO **SHALL** be able to dynamically design and assign connectivity and application resources to Services based on its understanding of the underlying network topology (across one or multiple internal and/or external networks) and application platform in order to manage the fulfillment and assurance of Services.

[D-LSO-RA-4]:LSO **SHOULD** be able to retrieve topology information from Network Domain Controllers.

[D-LSO-RA-5]:LSO **SHOULD** be able to retrieve application information from Application Domain Controllers.

[R-LSO-RA-10]: LSO **SHALL** own and manage a stateful inventory of services, network topologies (forwarding domains bounded by external and internal interfaces on edge network elements or network functions) and, optionally, resources, or have direct access to such external sources (e.g., domain managers).

[D-LSO-RA-6]: LSO **SHOULD** support the service view, application view, network view, and topology view abstractions (as described in Section 11 of this document).

[R-LSO-RA-11]: LSO **SHALL** be able to dynamically compute the list of technical actions to be supplied to the Service Delivery Orchestration process (described in Section 8.2.4) as an orchestration delivery plan (including the designed service layout, infrastructure resource requirements, and associated schedule) resulting from service topology and/or configuration changes to the stateful inventory in relation to part or all of one or more Customer orders or Service Control requests.

- This includes any Customer or system requests such as create, modify, move, delete, rollback, change administration status, etc. against any or all parts of the Service and/or its constructs. (Note that technical actions may be related to one or multiple internal networks managed by the Service Provider, but also targeted to external networks managed by Partners.)

[D-LSO-RA-7]: Technical Actions in LSO **SHOULD** include validation, feasibility checks, provisioning of network connectivity (e.g., forwarding constructs, and logical termination points as described in Section 11), provisioning of applications, requests to spin up new network functions (e.g., firewall function, monitoring function, etc.), requests to deliver a physical network function, and requests to order relevant Products from Partner (s) (e.g., an Access E-Line type product, VNF, etc.).

[R-LSO-RA-12]:LSO **SHALL** identify manual service configuration and activation activities which were not or could not be automated and orchestrate tracking of them, for delivery of the End-to-End Service.

8.2.3 Service Control Orchestration

While Order Fulfillment Orchestration deals with establishing or modifying a service through the ordering process, Service Control permits the service to be dynamically changed within specific bounds described in policies that are established at the time of ordering. After a Service is provisioned and established, LSO may enable Service Control to Customers, such as the ability to modify attributes subject to schedule policies and service constraint policies with for example specified ranges of valid values. Such dynamic behavior and associated constraints are defined based on the Product Offering and Product implemented by the Service. Service Control relates to capabilities such as activating or deactivating connections, throttling bandwidth or other QoS characteristics, activating or deactivating applications, etc.

Service Control Orchestration is triggered from a service configuration change request, a Customer initiated service control request, a scheduled service change event, or any other automated control means. This function allows Customers and/or systems to actively control the dynamic behavior of the Services (including connections, applications and interfaces), constrained by Customer and service policies in terms of service status or service configuration change actions allowed or not, and with approved characteristics value ranges or sets. As examples, LSO may support the throttling up or down the bandwidth associated with specific connections (including on a per CoS basis) within defined constraints (e.g., bounds or ranges), and turning on and off specific service access points within established service interfaces in accordance with their specified service policies .

Service Control Orchestration responsibilities include, but are not limited to:

- Scheduling, assigning and coordinating service control related activities;
- Undertaking necessary tracking of the execution process of service control requests;
- Adding additional information to an existing service control request under execution;
- Modifying information in an existing service control request under execution;
- Modifying the service control request status, and indicating completion of a service control request;
- Canceling a service control request;
- Monitoring the jeopardy status of service control requests, and escalating service control requests as necessary; and
- Instantiating, when appropriate, an event for the billing system to capture the policy-constrained change.

Requirements:

[R-LSO-RA-13]:LSO **SHALL** be able to receive a service control request, with policy-constrained content based on subsets of service specifications, defined within a technical catalog, or based on service administration status change.

[R-LSO-RA-14]:LSO **SHALL** be able to decompose a service control request into one or multiple Service configuration and activation activities, and orchestrate these configuration and activation activities.

[R-LSO-RA-15]:LSO **SHALL** be able to determine the necessary Service Components and configurations needed to support a Service instance.

[R-LSO-RA-16]:LSO **SHALL** ensure Service configuration and activation activities are assigned, managed and tracked efficiently to meet the agreed or estimated committed availability time or date.

[R-LSO-RA-17]:LSO **SHALL** support changing the administrative state (e.g., enabled or disabled) of a Service and each of its Service Components.

[D-LSO-RA-8]:LSO **SHOULD** support service control request revisions (add or modify request elements, such as modifying bandwidth) in case they are submitted against a request which is still in progress.

[D-LSO-RA-9]:LSO **SHOULD** support service control request cancellation, including rollback, intercepting the service control request execution.

[R-LSO-RA-18]:LSO **SHALL** be capable of orchestrating service control requests with operations support systems that need to be aware of changes to Service attributes, such as Fault Management and Performance / SLA Management.

[R-LSO-RA-19]:LSO **SHALL** undertake necessary tracking of the execution process, dynamically modify and report the Customer service control request status, and monitor the jeopardy status of service control requests, escalating service control requests as necessary.

[R-LSO-RA-20]: Upon completion of any billing-impacting changes due to Service Control Orchestration, LSO **SHALL**, where applicable, generate a service control change event targeted at the billing system.

8.2.4 Service Delivery Orchestration

Service Delivery Orchestration is responsible for coordinated execution of the service delivery orchestration plan, considering dependencies and such, generated by Service Configuration and Activation Orchestration, delegating and tracking the actual Service Components implementation to various delivery or implementation systems or methods, such as:

- One or multiple Network and Application Domain Controllers (e.g., SDN Controllers);
- An NFV Orchestrator (NFVO) (e.g., VNF delivery in virtual CPE);
- A request for a Partner product order for off-net Service Components (e.g., E-Access, Firewall);
- Any other system, such as a workforce management system (e.g., last mile fiber installation with human resources) or Supply Chain Management (e.g., delivery of a physical CPE).

Requirements:

[R-LSO-RA-21]:LSO **SHALL** support service delivery orchestration, based on a dynamically generated delivery plan (including the designed service layout, infrastructure resource requirements, and associated schedule), against one or multiple delivery systems, methods, or Partners, to fulfill a portion or the entirety of a Customer order or service control request:

- Delivery systems may include: WAN Controllers, SDN Controllers, service-capable EMSs, VNF Managers (VNFMs), Virtualization Infrastructure Managers, NFV Orchestrators, etc.
- Delivery methods may include orchestration of automated and manual methods, the latter being either explicitly managed by LSO or delegated to an external system (e.g., a manual provisioning system, a workforce management system, a supply chain management system, a project management system, and so forth).
- Delivery via partners may include orchestration of requests to Partners (via direct order or via internal request for order) to create, modify, move, delete, or rollback Service Components provided by Partners.

[R-LSO-RA-22]:LSO Delivery Orchestration **SHALL** undertake necessary tracking of the execution process of technical actions, dynamically report the delivery status, and monitor the jeopardy status of technical actions, initiating fall-out management as necessary.

8.3 Service Testing Orchestration

Service Testing Orchestration plays a critical role within LSO by automating the test (including Service Activation Testing and In-Service Testing) and verification of Services, seamlessly, across multiple operators.

LSO may be used to orchestrate and control the different systems capable of conducting tests and reporting on Services. These systems may be implemented within the network infrastructure, the element control managers or can be deployed on demand, in the form of virtual machines.
As the different locations and network elements involved in the fulfillment of end-to-end Services may not all be available at the same time, the Service Testing Orchestration flexibility allows for real-time staggered testing, from simple unit level connectivity tests, to end-to-end comprehensive Service Activation Testing.

Customer acceptance is received from the Customer. The Customer may view their particular services test results, or under special agreement with their Service Provider, be able to perform a set of predefined service acceptance tests.
Requirements:

[R-LSO-RA-23]:: LSO **SHALL** orchestrate end-to-end network connectivity and application testing, including flexibility for staggered testing. (e.g., testing two different OVCs in the operator networks before testing the EVC, testing an infrastructure service provided by a Partner before testing the application deployed on that infrastructure service.)

[D-LSO-RA-10]:LSO **SHOULD** orchestrate the performing of Service Component level testing at the Resource Management level with systems capable of conducting and reporting on Service Component tests.

[R-LSO-RA-24]: LSO **SHALL** facilitate and coordinate end-to-end service tests, and issue testing requests, via APIs, to systems capable of conducting and reporting on Service Component tests.

[D-LSO-RA-11]:LSO **SHOULD** orchestrate Customer acceptance testing.

8.4 Service Problem Management

Service Problem Management capabilities for LSO support alarm surveillance, including the detection of errors and faults. LSO may receive trouble-related information about the Service, either end-to-end or per Service Component. This information is organized to facilitate the evaluation of the overall performance and status associated with the Customer's Services. Customers may be provided with trouble-related information by LSO so that they may track the service impact and status of trouble resolution. Reports related with the Services may be provided to the Customer, including: correlated alarms, performance events, trouble reports, the potential root cause, and the identified impact on the Services. The Customer may also control the filtering of reports and notifications and may provide reports of problems and related information back to the Service Provider in order to aid resolution. Service Problem Management capabilities in LSO also allow the Customer to provide feedback on the proposed resolution. The Customer may also request that service-related tests be performed by the Service Provider on their behalf.

Requirements:

[R-LSO-RA-25]: LSO **SHALL** support alarm surveillance: detection of errors and faults and correlation to services.

[R-LSO-RA-26]: LSO **SHALL** orchestrate service level fault verification, isolation, and testing.

[R-LSO-RA-27]: LSO **SHALL** evaluate and present the service impact of specific failure conditions (e.g., specifying which services are negatively impacted by a specific fault on a network resource)

[R-LSO-RA-28]: LSO **SHALL** report correlated alarms, performance degradations, trouble reports, etc. to the Customer, including the potential root cause and identified impact on services.

[R-LSO-RA-29]: LSO **SHALL** control filtering of problem related notifications.

[R-LSO-RA-30]: LSO **SHALL** provide problem related information allowing the status of problem resolution to be tracked.

[R-LSO-RA-31]: LSO **SHALL** orchestrate Service fault recovery.

8.5 Service Quality Management

Service Quality Management capabilities in LSO include the collection of service performance information (e.g., delay, loss, etc.) in support of key quality indicators across all Connectivity and Cloud Operators who participate in delivering the Service. This also includes gathering of feedback from the Customer, including Customer-provided performance measurements. Service quality is analyzed by comparing the service performance metrics with the service quality objectives described in the SLS. The results of the service quality analysis are provided to the Customer as well as information about known events that may impact the overall service quality (e.g., maintenance events, congestion, relevant known problems, demand peaks, etc.). LSO Service Quality Management capabilities also include capacity analysis in support of traffic engineering, traffic management, and service quality improvement. Holistic and responsive traffic engineering capabilities manage aggregate traffic flows though the network based on measured and predicted demands in order to effectively meet the demand while maintaining service quality objectives.

Requirements:

[R-LSO-RA-32]: LSO **SHALL** collect service performance related information across involved domains.

[D-LSO-RA-12]:LSO **SHOULD** gather Customer perceived quality feedback.

[R-LSO-RA-33]: LSO **SHALL** analyze service quality by comparing the service performance metrics with the service quality objectives described in the SLS.

[R-LSO-RA-34]: LSO **SHALL** allow the definition of thresholds on service performance metrics based on service quality objectives.

[R-LSO-RA-35]: LSO **SHALL** provide performance information relevant to the Service.

[R-LSO-RA-36]: LSO **SHALL** provide the results of the service quality analysis to the Customer, including information about known events that may impact the overall service quality (e.g., maintenance events, congestion, relevant known troubles, demand peaks, etc.).

[R-LSO-RA-37]: LSO **SHALL** perform traffic and capacity analysis in support of traffic engineering.

[R-LSO-RA-38]: LSO **SHALL** perform service quality improvement to address detected degradations.

[R-LSO-RA-39]: LSO **SHALL** coordinate the management of aggregate traffic flows though the network based on capacity analysis and projected demands.

[R-LSO-RA-40]: LSO **SHALL** allow the definition of end-to-end SLA enforcement / assurance / resolution policies associated with the Product Offering.

8.6 Billing and Usage Measurements

Billing and Usage Measurements capabilities in LSO enable operators to gather and provide usage measurements, traffic measurements, and service-related usage events (e.g., changes in service bandwidth, etc.) describing the usage of Service Components and associated resources. LSO billing and usage measurement capabilities are responsible for the collection and correlation of such information relative to specific Services. Exception reports may be generated to describe where Service Components and resources have been used beyond the usage commitments as described in the SLS.

Requirements:

[R-LSO-RA-41]: LSO **SHALL** support the reporting of the usage of service capabilities and associated resources.

[R-LSO-RA-42]: LSO **SHALL** assemble Service Component usage data to compose an end-to-end view of service usage.

[R-LSO-RA-43]: LSO **SHALL** capture control-based service events (change in bandwidth, change in VNF type, etc.).

[D-LSO-RA-13]:LSO **SHOULD** generate exception reports to describe where service resources have been used beyond the commitments as described in the SLS.

[D-LSO-RA-14]:LSO **SHOULD** include billing management capabilities as described in MEF 50.1[21].

8.7 Security Management

Security Management in LSO provides for the protection of management and control mechanisms, controlled access to the network and applications, and controlled access to service-related traffic that flows across the network and applications. Such security management capabilities support the authentication of users and applications and provide access control to the variety of capabilities on the APIs supporting management and control based on the roles assigned to each authorized user. The security management capabilities of LSO include encryption and key management to ensure that only authenticated users are allowed to successfully access the management and control entities and functions. LSO security thwarts network attacks by taking responsive steps, such as applying filtering controls on specified traffic flows, when a specific threat is identified. A LSO specific threat model may be developed as an additional LSO related document.

Requirements:

In order to ensure the integrity and security of the management and control mechanisms supported within LSO:

[R-LSO-RA-44]: LSO **SHALL** provide authentication for all management interactions across LSO Interface Reference Points.

[D-LSO-RA-15]:LSO **SHOULD** provide role-based access control for users.

[R-LSO-RA-45]: LSO **SHALL** support encryption across cross-administrative domain interfaces (e.g., Service Provider to Partner interfaces, and Service Provider to Customer interfaces) and the associated key management capabilities.

[R-LSO-RA-46]: LSO **SHALL** orchestrate the management of rule-based traffic filtering controls for Services.

[R-LSO-RA-47]: LSO **SHALL** maintain information related to trust relationships with the domains and entities with which the components in LSO interact.

8.8 Analytics

Analytics capabilities in LSO support the fusion and analysis of information among management and control functionality across management domains in order to assemble a relevant and complete operational picture of the end-to-end Services, Service Components, and the supporting network and application infrastructure – both physical and virtual. Analytics ensures that information is visible, accessible, and understandable when needed and where needed to accelerate decision-making. For example, LSO analytics may utilize service fulfillment, control, and usage information to predict and trend service growth for the Connectivity and Cloud Operators.

Requirements:

[R-LSO-RA-48]: LSO **SHALL** support the fusion and analysis of information among management and control functionality across management domains.

[R-LSO-RA-49]: In support of analytics, LSO **SHALL** assemble a relevant and complete operational picture of the Services, Service Components, and the associated supporting network and application infrastructure, both physical and virtual LSO **SHALL** ensure that information is visible and accessible when needed and where needed to accelerate decision-making.

[R-LSO-RA-50]: LSO **SHALL** support prediction and trending of service growth and resource demand as compared to available resource capacity.

8.9 Policy-based Management

The behavior of LSO may be prescribed by the set of rules under which the LSO orchestration, management and control logic must operate. Service policies may be encoded in such rules in order to describe and design the dynamic behavior of Services. Coordinated Service relies on the orchestration of distributed capabilities across potentially multiple Connectivity and Cloud Operators to enable end-to-end management. LSO policy-based management capabilities provide rules-based coordination and automation of management processes across administrative domains supporting effective configuration, assurance, and control of services and their supporting resources.

In LSO, service design policies may enable the design and creation of end-to-end Services, and are aimed at being automated to adhere to the MEF 3.0 paradigm as described in Section 6.1. Furthermore, service objectives may be implemented as sets of policies with event-triggered conditions and associated actions, as well as intent-based policies. Such policies would adjust the behavior of services and service resources – including bandwidth, traffic priority, and traffic admission controls – allowing Services to adapt rapidly to dynamic conditions in order to satisfy critical, ever-changing needs and priorities.

Requirements:

[D-LSO-RA-16]:LSO **SHOULD** provide rules-based coordination and automation of management processes across administrative domains supporting effective configuration, assurance and control of services and their supporting Service Components.

[D-LSO-RA-17]:LSO **SHOULD** support service-related policies that encode rules that describe the design and dynamic behavior of the services.

[D-LSO-RA-18]:LSO **SHOULD** support service objective based policies that implement sets of rules with event triggered conditions, and associated actions.

[D-LSO-RA-19]:LSO **SHOULD** adjust the behavior of services and service resources, including bandwidth, traffic priority, and traffic admission controls through policies, allowing Services to adapt rapidly to dynamic conditions.

[D-LSO-RA-20]: Within LSO, user / party and service policies **SHOULD** be used to control and bound the objects, parameters, value ranges and states that are allowed to be created, modified, or deleted.

8.10 Customer Management

There are many types of interactions between Customers and Service Providers that are relevant to LSO. For example, a Service Provider may interact with potential Customers to determine serviceability of a Product Offering, helping to ensure that the underlying infrastructure is both capable and available to support the desired Product Offering or Service for the Customer.

Requirements:

The following requirements support the Customer interactions with LSO:

[R-LSO-RA-51]:LSO **SHALL** provide capabilities for the Customer to browse the product catalog for Product Offerings.

[R-LSO-RA-52]:LSO **SHALL** provide capabilities for the Customer to create, place and track orders.

[R-LSO-RA-53]:LSO **SHALL** provide capabilities for the Customer to request modification of their Service, including rules guiding the dynamic service characteristics.

[R-LSO-RA-54]:LSO **SHALL** provide capabilities for the Customer to provide Customer acceptance feedback and view Customer acceptance testing information.

[R-LSO-RA-55]:LSO **SHALL** provide capabilities for the Customer to view service performance and fault information.

[R-LSO-RA-56]:LSO **SHALL** provide capabilities for the Customer to place and track trouble reports.

[R-LSO-RA-57]:LSO **SHALL** provide capabilities for the Customer to view usage and billing information.

8.11 Partner Management

In support of LSO for the end-to-end Service, the Service Provider will interact with Partners. For example, a Partner may interact with the Service Provider to help the Service Provider to determine Service feasibility.

Requirements:

The following requirements support the Partner interactions with LSO:

[R-LSO-RA-58]:LSO **SHALL** provide capabilities for the Partner to provide product catalog information for Product Offerings.

[R-LSO-RA-59]:LSO **SHALL** provide capabilities for the Service Provider to develop, place and track orders with the Partner.

[R-LSO-RA-60]:LSO **SHALL** provide capabilities for the Service Provider to modify their Service, including rules guiding the dynamic service characteristics with the Partner.

[R-LSO-RA-61]:LSO **SHALL** provide capabilities for the Service Provider to request test initiation and view test result information from the partner.

[R-LSO-RA-62]:LSO **SHALL** provide capabilities for the Partner to provide service performance and fault information.

[R-LSO-RA-63]:LSO **SHALL** provide capabilities for the Partner to receive trouble reports and provide trouble status updates.

[R-LSO-RA-64]:LSO **SHALL** provide capabilities for the Partner to provide usage and billing information

[R-LSO-RA-65]:LSO **SHALL** provide capabilities for the Service Provider and the Partner to track Service Level Objectives (SLOs) for the interactions between them such as the number of dynamic service attribute modification requests by the Service Provider that are delivered by the Partner.

8.12 License Management

In PNF (Physical Network Function) based networks, Service Providers purchase PNFs as boxed solutions. They establish a long-term commitment with suppliers/vendors on support of their products. New licenses are usually ordered every three to five years. Software is highly coupled with hardware.

In networks of PNFs and VNFs, VNFs are decoupled from hardware and not confined to a box as described in TMF-IG1143 [36] and ETSI GR NFV-EVE 010 [5]. Furthermore, VNFs can be purchased on-demand and scaled anytime. As a result, VNFs are highly likely to have frequent changes in their licenses. Subscription based and pay-as-you-use approach are common. This section addresses VNF license management and operations only.

Requirements:
In order to ensure the license management within LSO:

[R-LSO-RA-66]:LSO **SHALL** provide License Policy Management to manage license agreements and modifications for licenses associated with services provided by a Partner.

[R-LSO-RA-67]:LSO **SHALL** provide usage monitoring to ensure compliance with the license agreement.

Note that the license agreement may include license access and modification rights; license usage within service provider organizations, suppliers, customers, partners; and life cycle guidelines. This may require monitoring such as daily number of users or devices, time remaining that the subscription expires. The SP may have a license inventory that has a list of all VNF licenses in a repository with information related to license entitlement identifying which licenses the organization has the legal right to use and assign to users, purchase costs, proof of purchase, maintenance contracts, and service fees.

The SP may set license reminders for subscription renewals, maintenance times, and approvals.

[R-LSO-RA-68]: LSO **SHALL** support VNF charging models such as time-based, fixed-pricing, and usage-based model including charging for automated scaling of VNFs, and VNF redundancy, etc.

Note that VNF redundancy is needed to minimize service downtime. It could be in the form of 1:1, 1+1 or 1:n.

[R-LSO-RA-69]: LSO **SHALL** support on demand activation/deactivation of VNF licenses.

Note that the VNF license on-demand activation and deactivation are usually encountered during VNF, VM, and Container failures and failure recoveries.

9 LSO Reference Architecture

The LSO Reference Architecture characterizes the management and control domains and functional management entities that enable cooperative LSO capabilities. The architecture also identifies the Management Interface Reference Points, the logical points of interaction between specific functional management entities. These Management Interface Reference Points are further defined by Interface Profiles and implemented by APIs. The High-Level LSO Reference Architecture is shown in Figure 2. Note that this is a functional architecture that does not describe how the functional management entities are implemented (e.g., single vs. multiple instances), but rather identifies functional management entities that provide logical functionality as well as the points of interaction among them.

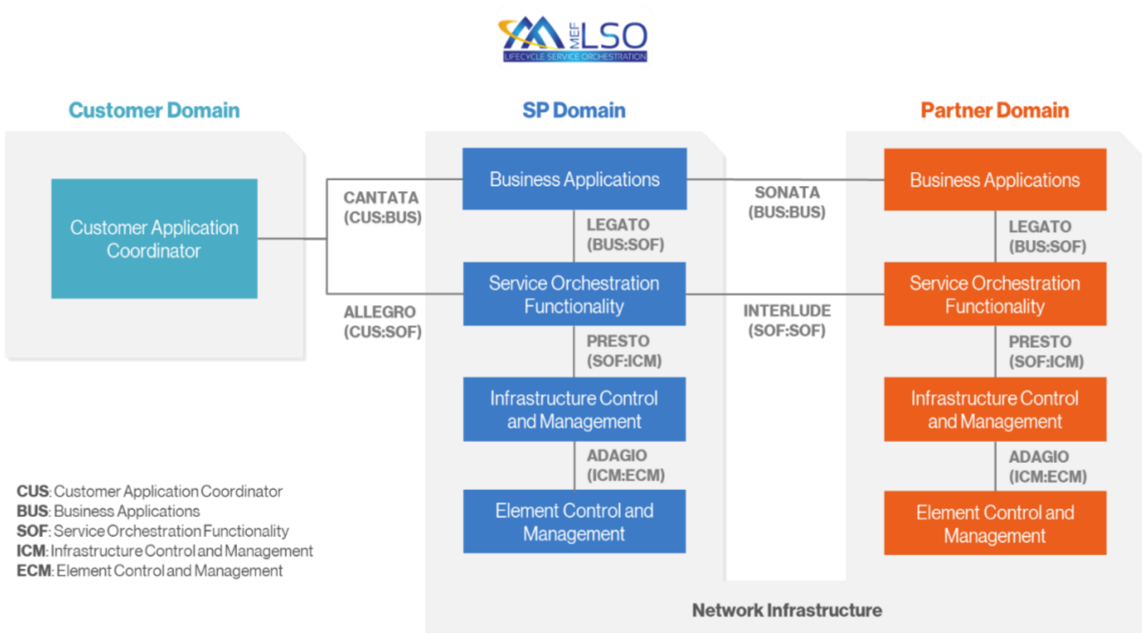


Figure 2 – LSO Reference Architecture

9.1 Definition of LSO Functional Management Entities

This section defines each of the LSO functional management entities within the LSO ecosystem that are involved in providing the cooperative LSO capabilities. The definition for each functional management entity describes its logical scope of functionality. The abbreviation that is used within the LSO Reference Architecture for each functional management entity is also provided.

- Business Applications (BUS): The Service Provider functionality supporting Business Management Layer functionality (e.g., product catalog, ordering, billing, relationship management, license management, etc.).
- Service Orchestration Functionality (SOF): The set of service management layer functionality supporting an agile framework to streamline and automate the service lifecycle in a sustainable fashion for coordinated management supporting design, fulfillment, control, testing, problem management, quality management, usage

measurements, security management, analytics, runtime implementations of license policies, and policy-based management capabilities providing coordinated end-to-end management and control of Services.

- Infrastructure Control and Management (ICM): The set of functionality providing domain specific network, application and topology view resource management capabilities including configuration, control and supervision of the infrastructure. ICM is responsible for providing coordinated management across the resources within a specific management and control domain. For example, a system supporting ICM capabilities provides connection management and/or application management across a specific domain. Such capabilities may be provided within systems such as subnetwork managers, SDN controllers, VNFM, NFVO, etc. Section 9.1.1 provides some ICM implementation examples.
- Element Control and Management (ECM): The set of functionality supporting element management layer capabilities for individual infrastructure elements including virtual infrastructure. While a system supporting ECM capabilities provides for the abstraction of individual infrastructure elements, it may reflect the element view for multiple elements, but not provide coordinated management across the set of elements.
- Customer Application Coordinator (CUS): A functional management entity in the Customer domain that is responsible for coordinating the management of the various Service needs (e.g., compute, storage, network, etc.) of specific applications. The CUS may be responsible for the harmonization of Services on behalf of multiple applications. The CUS supports Customer interactions with the Service Provider to request, order, modify, manage, control, and terminate Products or Services.

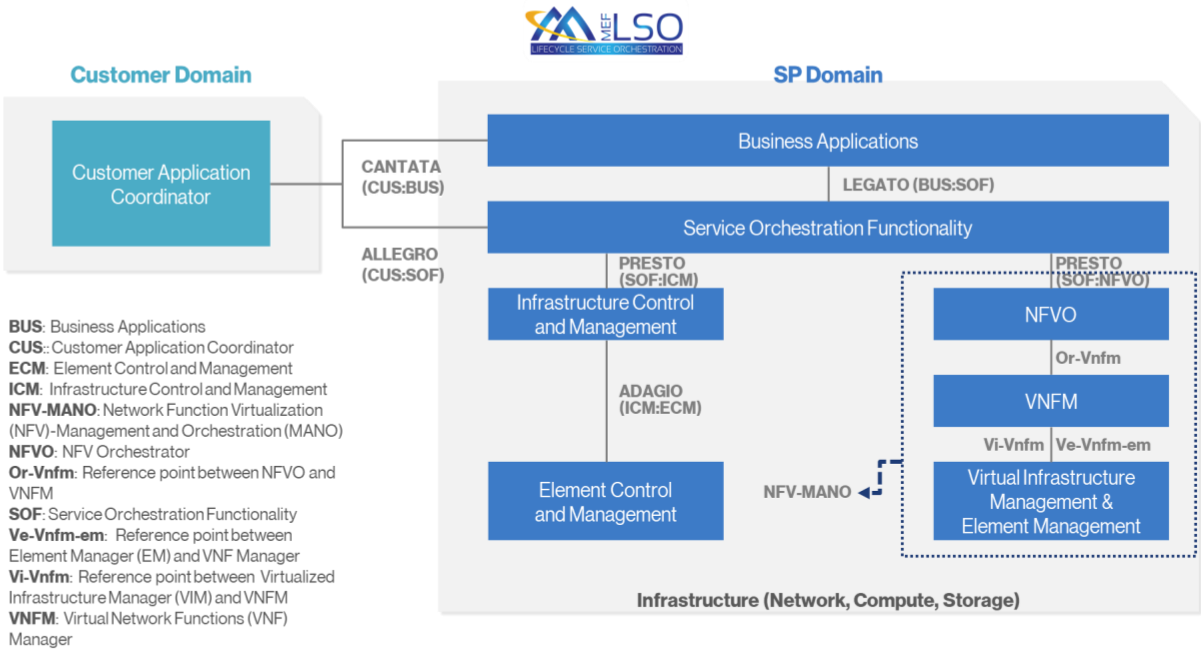
9.1.1 Examples of SDO Architectural Elements within Infrastructure Control and Management

This section gives some examples of SDO defined architectural elements that provide functionality within the scope of the LSO ICM functional management entity, namely the ONF SDN Controller, the ETSI NFV Management and Orchestration Network Functions Virtualization Orchestrator (NFVO), VNFM, and MEF EMS (or Subnetwork Manager).

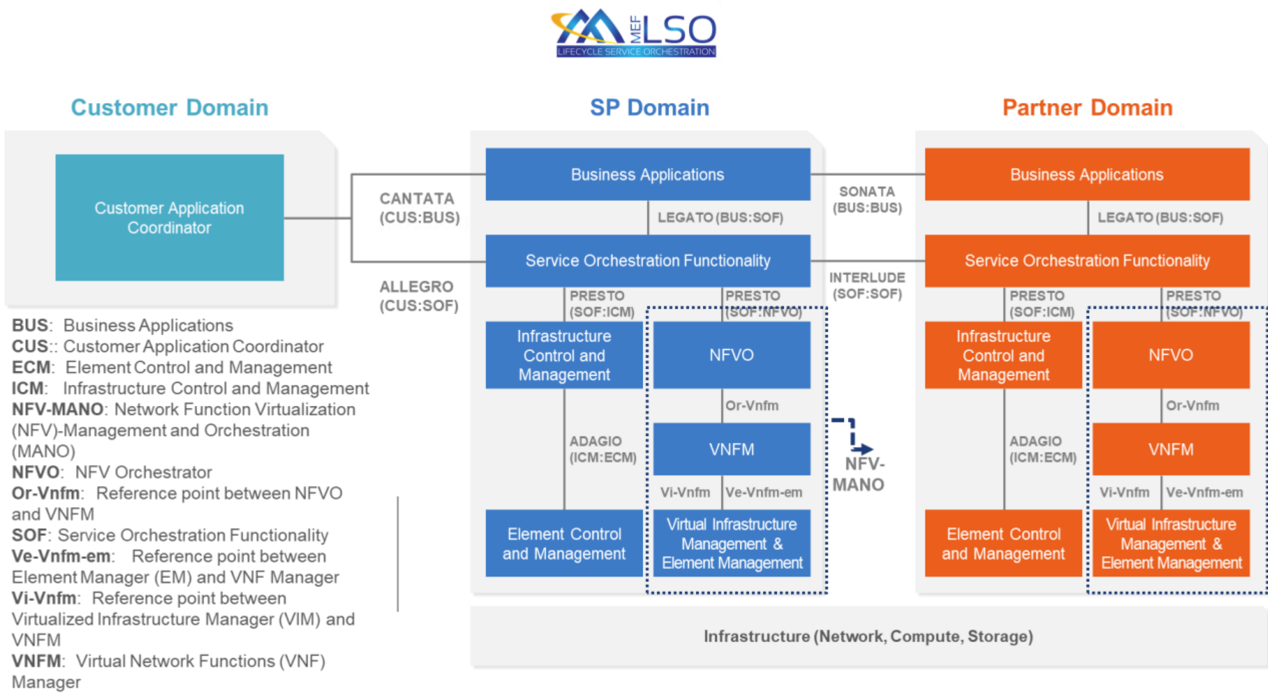
- ONF SDN Controller [ONF TR-521.1 [31]]: The functionality in charge of translating the network requirements from the SDN Application layer down to the SDN Datapath and providing the SDN Applications with an abstract view of the network including statistics and events.
- ETSI NFV Management and Orchestration (MANO) - NFV Orchestrator (NFVO) [ETSI GS NFV-MAN 001[2]]: The functionality that manages the ETSI Network Service (NS) lifecycle defined in ETSI GS NFV 003 [4] and coordinates the management of the NS lifecycle. NF lifecycle supported by the VNF Manager (VNFM) and Network Functions Virtualization Infrastructure (NFVI) resources supported by the Virtualized Infrastructure Manager (VIM), and optimized allocation of the necessary resources and connectivity between two NFVI-POPs supported by Wide Area Network Infrastructure Manager (WIM).

  Figure 3 depicts the integration of the ETSI NFV Management and Orchestration (MANO) architecture (ETSI GS NFV-MAN 001 [2]) with the MEF LSO architecture to support the management and orchestration of Cloud Services. Both Connectivity and Cloud Operators can play a role of a SP or a Partner.

- EMS or Subnetwork Manager: The ICM may also be implemented by traditional subnetwork managers (aka WAN Managers.) and EMSs that manage the connectivity and application across specific network domains or subnetworks in MEF 15 [15].



(a) ETSI NFV-MANO Integrated LSO Architecture of a SP



(b) ETSI NFV-MANO Integrated LSO Architecture of a SP and a Partner

Figure 3 – LSO Reference Architecture Integrated with MANO

9.2 Definition of Management Interface Reference Points

Definitions for each Management Interface Reference Point within the LSO Reference Architecture are provided in this section. Each Management Interface Reference Point is identified with a name (e.g., Cantata), as well as a context identifying the interacting LSO functional management entities (e.g., CUS:BUS).

- Cantata (CUS:BUS): The Management Interface Reference Point that provides a Customer Application Coordinator (including enterprise Customers) with capabilities to support the operations interactions (e.g., ordering, billing, trouble management, etc.) with the Service Provider's Business Applications for a portion of the Service Provider service capabilities related to the Customer's Products and Services (e.g., Customer Service Management interface). Since cross domain interactions are supported, additional security considerations need to be addressed on this Management Interface Reference Point.
- Allegro (CUS:SOF): The Management Interface Reference Point that allows Customer Application Coordinator supervision and control of dynamic service behavior (see Section 8.2.3) of the LSO service capabilities under its purview through interactions with the Service Orchestration Functionality. When a Customer exercises dynamic service behavior via Allegro, the Service Orchestration Functionality must validate each request using the Service specific policies that govern such dynamic behavior. Such dynamic behavior and associated constraints are defined based on the Product Specification implemented by the Service. For example, a Service specific dynamic service policy may

describe the range of bandwidth in which the Customer is permitted to throttle. Allegro may also be used to share service level fault information with the Customer, and/or request testing. Since cross domain interactions are supported, additional security considerations need to be addressed on this Management Interface Reference Point.

- Legato (BUS:SOF): The Management Interface Reference Point between the Business Applications and the Service Orchestration Functionality needed to allow management and operations interactions supporting Services. For example, the Business Applications may, based on a Customer order, use Legato to request the instantiation of a Service. Legato may also allow the SOF to describe Services and capabilities it is able to instantiate.
- Sonata (BUS:BUS): The Management Interface Reference Point supporting the management and operations interactions (e.g., ordering, billing, trouble management, any other interactions with potential commercial implications) between two Operators (e.g., Service Provider Domain and Partner Domain). For example, the Service Provider Business Applications may use Sonata to place an order to a Partner provider for an access service that is needed as a part of an end-to-end Connectivity Service. Similarly, the Service Provider Business Application may use Sonata to place an order to a Partner for Application (s) that is needed for a Cloud Service. Since cross domain interactions are supported, additional security considerations need to be addressed on this Management Interface Reference Point.
- Interlude (SOF:SOF): The Management Interface Reference Point that provides for the coordination of a portion of LSO services within the partner domain that are managed by a Service Provider's Service Orchestration Functionality within the bounds and policies defined for the service. Through Interlude, the Service Orchestration Functionality may request initiation of technical operations or dynamic control behavior associated with a Service with a Partner domain (see Section 8.2.3). Such requests must be within the constraints set forth in the policies associated with established Services and performed without impacting business applications. For example, to satisfy a Customer request, the Service Orchestration Functionality may request changes to a CE-VLAN ID mapping at a UNI that resides in a partner domain. Interlude may also be used to share service level fault information with the partner domain and/or request testing. Since cross domain interactions are supported, additional security considerations need to be addressed on this Management Interface Reference Point.
- Presto (SOF:ICM): The resource Management Interface Reference Point needed to manage the infrastructure, including network, applications, and topology view related management functions. For example, the Service Orchestration Function will use Presto to request ICM to create connectivity or functionality associated with specific Service Components of an end-to-end Connectivity Service within the domain managed by each ICM. Similarly, SOF can use Presto to request ICM to configure VNFs or Network Services (NSs)-as defined in ETSI GS NFV 003 [4]- of a Cloud Service. Presto may also allow the ICM to describe Resources and capabilities it is able to instantiate.
- Adagio (ICM:ECM): The element Management Interface Reference Point needed to manage resources, including element view related management functions. For example, ICM can use Adagio to implement cross-connections or network functions on specific elements via the ECM functionality responsible for managing the element. Similarly, ICM can use Adagio to configure vCPU and memory for a given VNF.

Note that more details about the types of interactions envisioned for each Management Interface Reference Point are given in Table 4 of Appendix A.

10 Operational Threads for LSO

This section is focused on the Operational Threads for the LSO Reference Architecture. Operational Threads describe the high-level Use Cases of LSO behavior as well as the series of interactions among LSO management entities, helping to express the vision of the LSO capabilities. The interactions described within each Operational Thread address the detailed involvement of the Interface Reference Points in the LSO Reference Architecture. Each subsection identifies and outlines some of the operational threads that are being developed as part of the LSO Reference Architecture. Each Operational Thread describes the orchestration within the LSO Reference Architecture highlighting the coordination within a Service Provider domain and also addressing the interactions with both the Customer domain and Partner domain. In addition, Operational Threads are mapped to the requirements they support in the LSO Reference Architecture and Framework. The detailed Operational Threads defined in this section describe the interactions relative to each Interface Reference Point. These interaction details will serve as a foundation for future work on the functional requirements for each Interface Reference Point. Such functional requirements will be used as the basis for Interface Profile definitions.

Operational Threads identified for LSO include:

- Partners on-boarding
- Product Ordering and Service Activation Orchestration
- Controlling a Service
- Customer Viewing Service Performance and Fault Reports and Metrics
- Placing and Tracking Trouble Reports
- Assessing Service Quality
- Collection and Reporting of Billing and Usage
- Managing Licenses

10.1 Partners On-boarding

The Service Provider begins a business relationship with Partners. The Product Offering capabilities of each Partner are shared with the Service Provider, along with any associated billing information and quality objectives. Rules guiding the business arrangement with the Partner may be codified within Policies. The Service Provider may use the details of the Partner's Product Offerings to identify the potential capabilities of Service Components that could be implemented using the Partner's products.

Note that the commercial agreements are established prior to the Partner on-boarding. During the on-boarding process, the Partner shares its product information with the Service Provider. The SP may choose to update its product catalog with the Partner’s product information.

10.2 Product Ordering and Service Activation Orchestration

10.2.1 Purpose:

Describe an operational thread for ordering of a product through activation orchestration within the LSO ecosystem for a service both within the provider domain and the partner domain. The intent of this operational thread is to include the various components within the ordering lifecycle including, but not limited to, the use of a product catalog, quoting, and serviceability. These capabilities may exist independently outside the order submission process or may be inclusive within the defined order submission workflow.

The operational thread for product ordering and service activation is depicted in Figure 4.

10.2.2 Steps:

1. Customer retrieves Product catalog and existing Product assets (e.g., existing service locations, existing UNIs, existing Product Instances, etc.): Customer-> Cantata -> Business Applications
2. Customer selects, specifies parameters and gets serviceability and a quote for the Product: Customer -> Cantata -> Business Applications
3. Business Applications decompose the product into its services and SOF decomposes the services into its service components:
   a) Business Applications begin determination of the Product serviceability (e.g., interacts with Billing, selection of Partner products, etc.)
   b) Business Applications request that SOF determines components of the service within the SP footprint and within the Partner footprint. BUS-sp -> (Legato) -> SOF

      - An alternative is for the Business Applications to lookup Partner service options using a Product Catalog instead of topology information.

   c) Business Applications inquire the SP footprint aspects of serviceability BUS-sp -> Legato -> SOF
   d) Business Applications inquire the Partner footprint aspects of the service and interrogate the Partner for Serviceability and quotes BUS-sp -> Sonata -> BUS-partner
   e) Business Applications generate the quote for the Customer: Business Applications -> Cantata -> Customer
4. Customer orders connectivity Product: Customer -> Cantata -> Business Applications
5. Business Applications perform Product to Service mapping
6. Business Applications analyze Partner footprint aspect of the ordered Product and places the appropriate Product Orders with Partners (and receives Partner commitments): BUS-sp -> Sonata -> BUS-partner
7. Business Application requests fulfillment of the Service(s) within the SP footprint: Business Applications -> Legato -> SOF
8. SOF designs the Service Components within the SP footprint (some may exist, some may need to be designed and created) including forwarding constructs across forwarding domains and associated interfaces as well as network functions to support the Service, including identification of the Partners(e.g., access providers) for any additional

forwarding constructs and network functions within the Partner footprint.

Note that determination of Service Components within the Partner footprint may be determined by the Business Applications before the service request is placed or via Partner domain discovery at Service level.

Note that SOF might need to initiate the installation request for hardware (e.g., CPE) and be aware of scheduling and lifecycle of all service components).

9. SOF requests configuration and activation of interfaces, forwarding constructs and network functions:
    a) SOF requests configuration and activation of network functions and forwarding constructs across each internal forwarding domain: SOF -> Presto -> ICM
    b) SOF requests fulfillment of Product Orders or Service Requests to Partner for services including components such as network functions, interfaces, and forwarding constructs across each Partner forwarding domain. There are two options for such interactions between the Service Provider and the Partner:
        a. SOF-sp -> Interlude -> SOF-partner (Guided by policy rules with the service definition)
        b. SOF-sp -> Legato -> BUS-sp -> Sonata -> BUS-partner
10. Each ICM determines the elements involved and controls the activation of the network functions and forwarding construct across each element: ICM -> Adagio- > ECM
11. Once the Service Components supporting the Service are successfully configured and activated, SOF orchestrates Service Activation Testing (Note: can be staggered when more than 2 sites): SOF -> Presto -> ICM (also SOF-sp -> Interlude -> SOF-partner for partner components)
12. When the end-to-end testing is successful:
    a) SOF synchronizes and activates proactive performance monitoring for the service and components (can be staggered when more than 2 sites)
       [Note: It is possible to address testing failures with policy driven closed loop control]
13. When all testing is completed (can be staggered when more than 2 sites), the SOF performs the state change for the Service (per order component) and informs the Business Applications that the service is now active. (Note: state changes will be tracked and made available to the customer throughout ordering and activation): SOF -> Legato -> Business Applications
14. The customer is notified that the Product Instance is ready to use: Business Applications -> Cantata -> Customer
15. Customer performs testing and accepts the Product Instance: Customer -> Cantata -> Business Applications
    a) E.g., Billing capability for the product assets (can be staggered); Billing commences

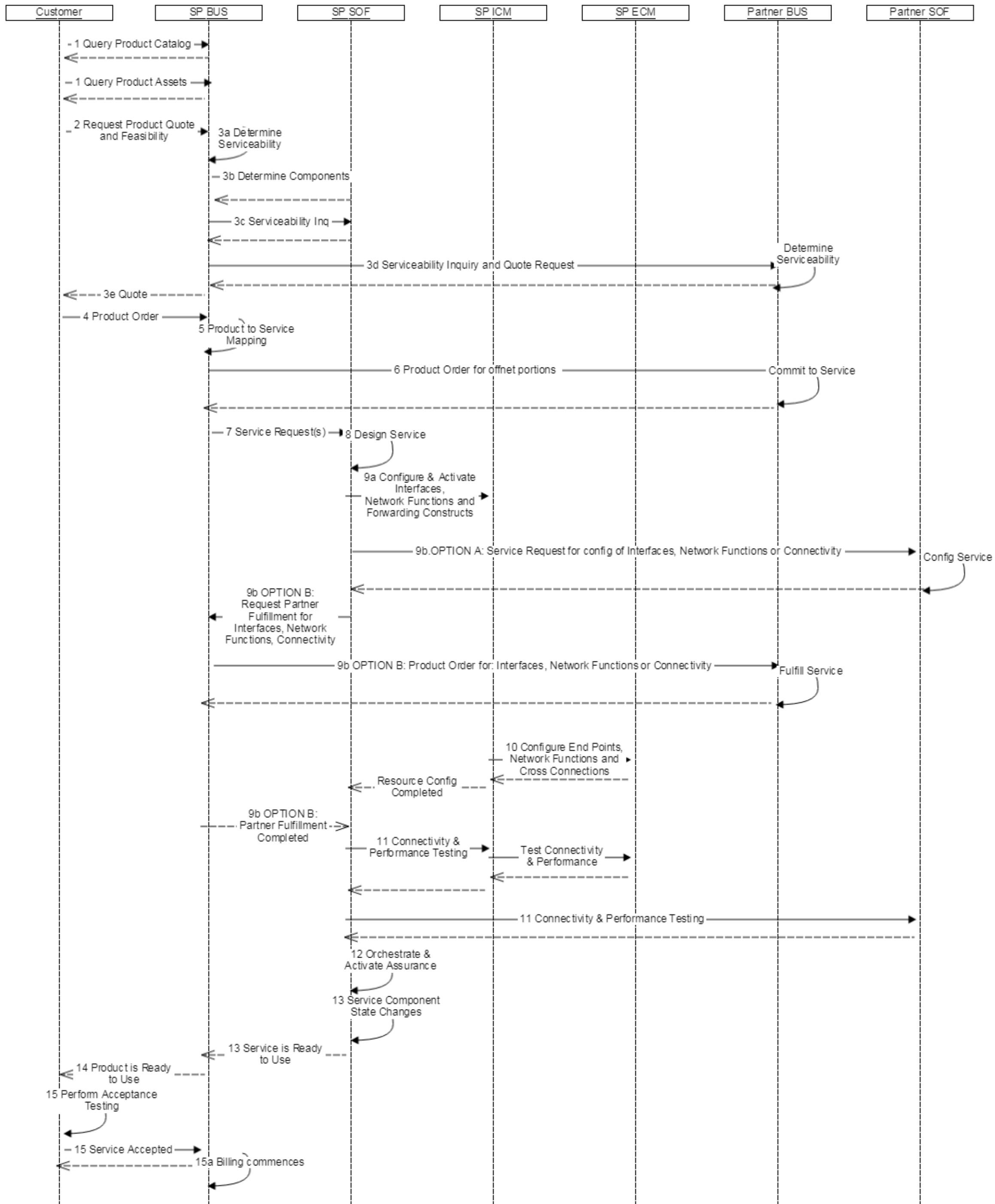


Figure 4 – Product Ordering and Service Activation Orchestration Sequence Diagram

10.3 Controlling a Service

10.3.1 Purpose:

The Customer initiates a request to dynamically control a permitted aspect of its Service (e.g., bandwidth change or implementing traffic filtering controls, etc.).

In the Service Provider domain, LSO uses the defined service constraints and policies to determine if the dynamic control request is permitted and parameterized within the permissible bounds. If the dynamic control request needs to be supported by Service Components within a Partner domain, LSO coordinates the changes needed to support the request with the Partner. In addition, LSO effects the necessary changes within its own domain to service the request. The updated Service Components are tested. The Customer is also informed about the status of the request.

The operational thread for controlling a service is shown in Figure 5.

10.3.2 Steps:

1. Customer queries the defined constraints and policies that describe the permitted dynamic behavior of the Service (e.g., bandwidth change or implementing traffic filtering controls, bounds on parameters, etc.): Customer -> Allegro > SOF-sp
   [Note that dynamic behavior at the service level is derived from the customer product contract]
2. Customer requests changes in Service-related parameters as permitted by the defined constraints and policies: Customer -> Allegro > SOF-sp
3. SOF-sp verifies that the requested changes fall within the permitted constraints, bounds, and policies.
4. SOF-sp identifies Service Components including forwarding constructs across forwarding domains and associated interfaces as well as network functions that need to be reconfigured in support of the request. This includes identification of the Service Components supported by Partners (e.g., access providers) for update to any necessary forwarding constructs and network functions within the Partner footprint.
5. SOF-sp requests reconfiguration of identified interfaces, forwarding constructs and network functions:
   1. SOF-sp requests reconfiguration of identified Service Components within the internal forwarding domains of the Service Provider: SOF-sp -> Presto -> ICM
   2. SOF-sp requests reconfiguration to Partners for identified Service Components (Partner Services) across each external forwarding domain: SOF-sp -> Interlude -> SOF-partner
6. Each ICM determines the elements involved and controls the reconfiguration of the network functions and forwarding construct across each element: ICM -> Adagio- > ECM
7. Each ICM reports back the results of the reconfiguration request: ICM -> Presto -> SOF-sp
8. Each External Provider reports back the results of the reconfiguration request: SOF-partner -> Interlude-> SOF-sp
9. SOF-sp requests testing (i.e., in-service or out-of-service testing) of identified Service Components (e.g., interfaces, forwarding constructs and network functions):

    1. SOF-sp requests testing of identified Service Components within the internal forwarding domains of the Service Provider: SOF-sp -> Presto -> ICM
    2. SOF-sp requests testing to External Providers for identified Service Components (Partner Services) across each external forwarding domain: SOF-sp -> Interlude -> SOF-partner

10. Each ICM determines the elements involved and controls the testing of the network functions and forwarding construct across each element: ICM -> Adagio- > ECM
11. Each ICM reports back the results of the testing request: ICM -> Presto -> SOF-sp
12. Each External Provider reports back the results of the testing request: SOF-partner -> Interlude-> SOF-sp
13. Once the Service Components supporting the request are successfully reconfigured and tested, SOF-sp synchronizes the Inventory and Assurance capabilities for the Service and Service Components.
14. SOF-sp generates a Usage Event to the Business Applications for Product Instance: SOF-sp -> Legato -> Business Applications

    Note that he Business Applications determine billing impact of the Usage Events due to service control changes.
15. The customer is notified that the Service is updated, tested, and is ready with requested changes to use: SOF-sp -> Allegro-> Customer

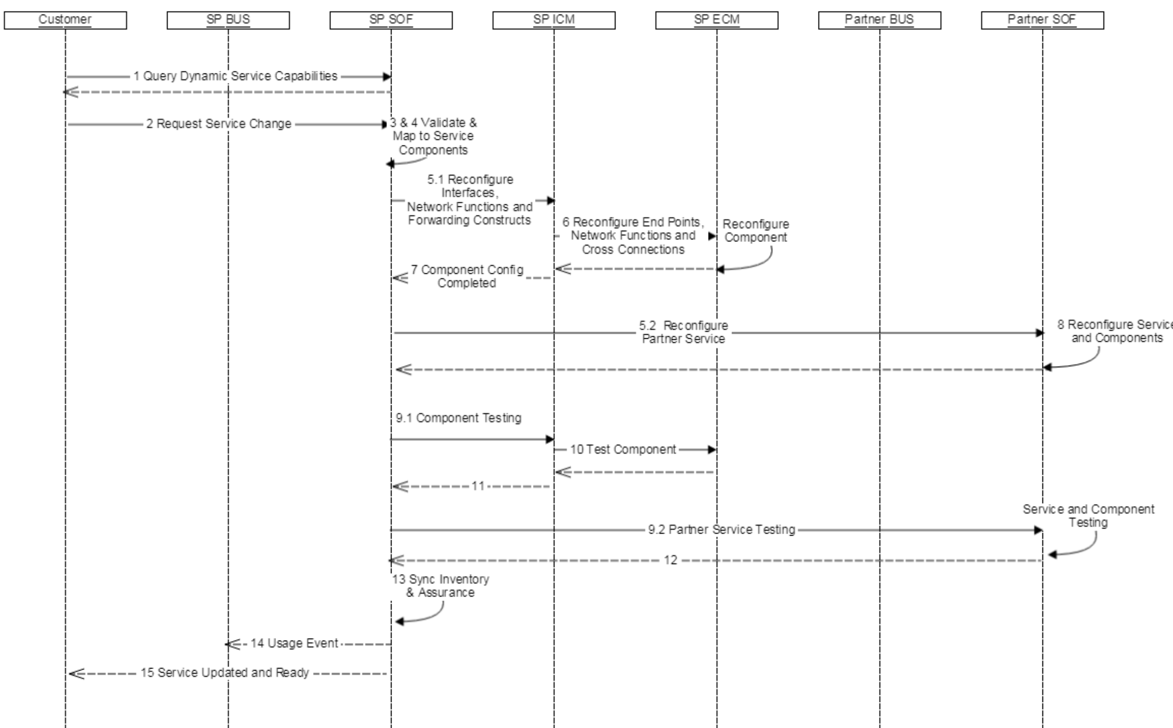


Figure 5 – Controlling a Service Sequence Diagram

After a dynamic control request from a Customer is implemented, dynamic modifications may be tested by the SP and the Customer.

10.4 Customer Viewing of Performance and Fault Reports and Metrics

10.4.1 Purpose:

The Customer wishes to view performance and fault information related to its Product Instances and associated Services. In the Service Provider domain, LSO may receive fault and performance related information about the Service, either end-to-end or per each Service Component. This information is organized to facilitate the evaluation of the overall performance and status associated with the Customer's Services and Product Instances. LSO gathers the information requested by the Customer and assembles it into a report. The Customer may also request that reports be generated on a scheduled or exception basis.

The customer viewing of service performance, fault reports, and metrics is depicted in Figure 6.

10.4.2 Steps:

1. The Customer retrieves information about the types of Performance and Fault Reports that may be requested for a specific Service: Customer -> Allegro -> SOF-sp
2. The Customer requests a specific Performance or Fault Report related to existing Services and / or a visible Service Components (e.g., If the SP permits the Customer to view specific connectivity flows or network functions etc.): Customer -> Allegro -> SOF-sp
3. SOF-sp determines the Information (e.g., Performance or Fault Metrics) that are needed in order to assemble the Performance or Fault Report requested by the Customer.
4. If the needed Information are not cached and current:
    1. SOF-sp requests the Information from the ICM domains that are responsible for generating the needed pieces of information: SOF-sp -> Presto -> ICM; ICM -> Adagio -> ECM
    2. SOF-sp requests the Information from the Partner domains that are responsible for generating the needed pieces of information: SOF-sp -> Interlude -> SOF-partner
5. SOF-sp assembles the Performance or Fault Report containing the Information requested by the Customer, and alerts the Customer of the Performance or Fault Report availability: SOF-sp -> Allegro -> Customer
6. The Customer retrieves the Performance or Fault Report from the Service Provider: SOF-sp -> Allegro -> CUSTOMER

Variations:

- Scheduled Performance Reports
- Triggered Performance Reports (e.g., SLS threshold exceeded, policy based, etc.)

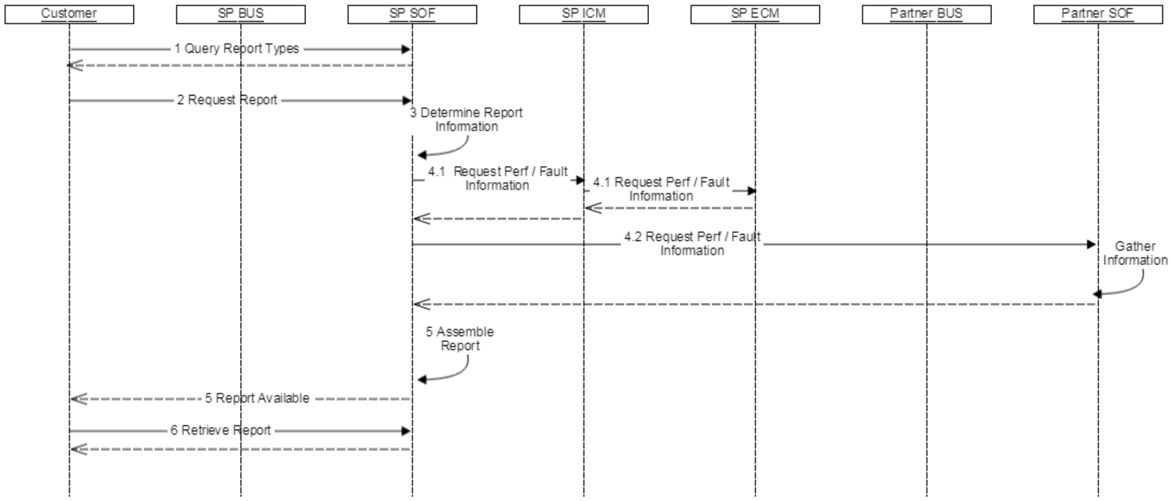


Figure 6 – Customer Viewing Service Performance and Fault Reports and Metrics Sequence Diagram

10.5 Placing and Tracking Trouble Reports

10.5.1 Purpose:

Trouble Reports related with the Customer's Product Instances and Services may be placed by the Customer. In the Service Provider, LSO gathers and fuses trouble and fault information related to the Customer's Product Instances and Services and associates it to the Trouble Report. LSO would also attempt to remedy the reported trouble by reconfiguring, reassigning, and / or rerouting aspects of the Service. LSO also indicates if manual intervention is required to resolve the trouble, and tracks the status of any associated repair activities to help determine trouble resolution status. The status of trouble resolution is reported back to the Customer.

The operational thread for placing and tracking trouble reports is depicted in Figure 7.

10.5.2 Steps:

1. Customer provides a Trouble Report related to a connectivity Product: Customer -> Cantata -> Business Applications
2. Business Applications perform Product to Service mapping
3. Business Applications inform SOF of the customer reported problem with the connectivity Service(s): Business Applications -> Legato -> SOF-sp
4. SOF analyzes the reported problem on the Service and identifies related Service Components including forwarding constructs across forwarding domains and associated interfaces as well as network functions supporting the Service, including identification of the Service Components provided by Partners
5. SOF identifies any previously detected errors and faults that are correlated to the Service or Service Components.
6. SOF requests current configuration and fault information related to the identified interfaces, forwarding constructs and network functions:

    1. SOF requests configuration and fault information of network functions and forwarding constructs across each internal forwarding domain: SOF-sp -> Presto -> ICM
    2. SOF requests configuration and fault information of Service Components in each external forwarding domain: SOF-sp -> Interlude-> SOF-partner
7. If not already cached, each ICM determines the elements involved and requests fault and configuration for the network functions and forwarding construct across each element: ICM -> Adagio- > ECM
8. Once the configuration and fault information for the Service Components supporting the Service are successfully gathered, SOF analyzes the information to diagnose and identify the Trouble, if necessary SOF orchestrates additional end-to-end and per Service Component testing: SOF-sp -> Presto -> ICM (also SOF-sp -> Interlude -> SOF-partner for partner components)
9. SOF coordinates and tracks the resolution of the Trouble, including reconfiguring, reassigning, and / or rerouting aspects of the Service.
10. SOF also indicates if manual intervention is required to resolve the trouble, and tracks the status of any associated repair activities to help determine trouble resolution status.
11. The status of trouble resolution is reported to the Business Applications: SOF-sp -> Legato -> Business Applications
12. The status of trouble resolution is reported to the Customer: Business Applications -> Cantata -> Customer (Alternative, SOF could provide updates via Allegro)

Service Provider may detect a problem and issue a trouble ticket as well.

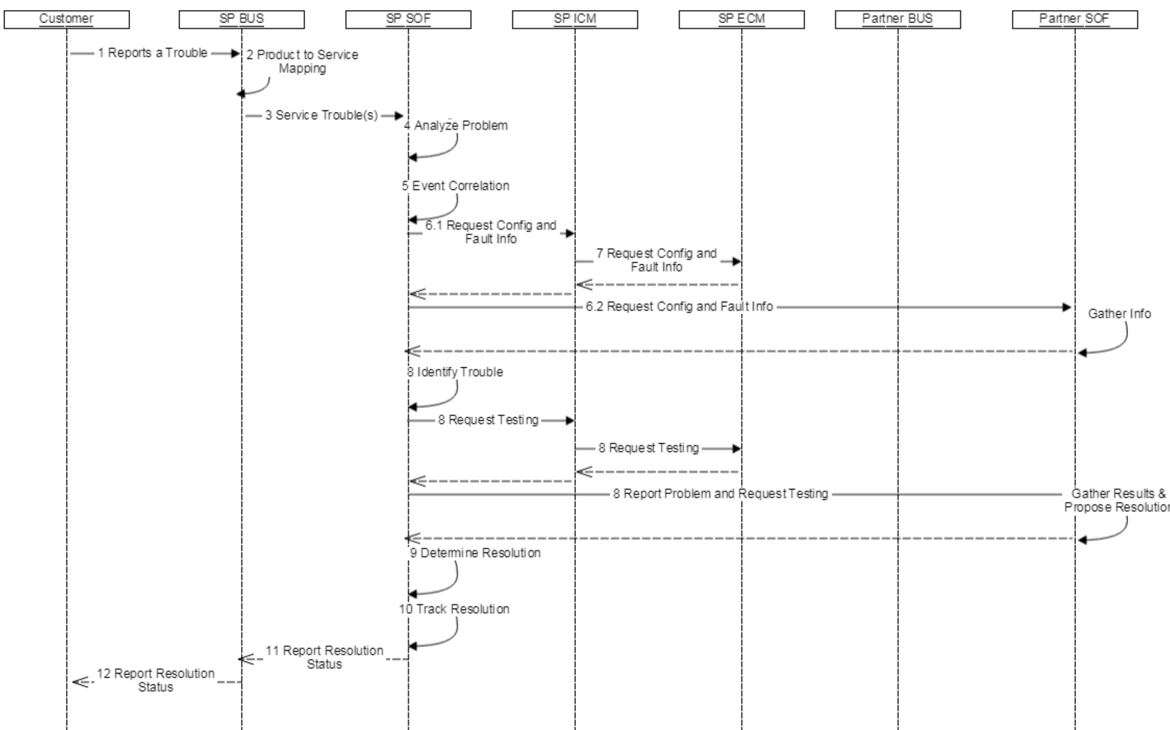


Figure 7 – Placing and Tracking Trouble Reports Sequence Diagram

10.6 Assessing Service Quality

10.6.1 Purpose:

The Service Provider needs to determine if the SLS for a Service and for dynamic service attribute modifications are being met. As depicted in Figure 8, service quality is analyzed by gathering the necessary service performance related measurement and comparing these service performance metrics with the service performance objectives and dynamic service attribute modification objectives described in the SLS.

10.6.2 Steps:

1. Periodically, SOF requests current performance information related to the identified interfaces, forwarding constructs and network functions. (Note: Instead of, or in addition to, periodic polling, the ICM (via Presto) or the partner SOF (via Interlude) might also send Threshold Crossing Alerts when particular performance thresholds are crossed):
    1. SOF requests performance information of network functions and forwarding constructs across each internal forwarding domain: SOF-sp -> Presto -> ICM
    2. SOF alerts Partner and requests performance information of Service Components in each external forwarding domain: SOF-sp -> Interlude -> SOF-partner
2. If not already cached, each ICM determines the elements involved and requests performance information for the network functions and forwarding construct across each element: ICM -> Adagio- > ECM
3. Once the performance information for the Service Components supporting the Service are successfully gathered, SOF analyzes the information based on the SLS to identify the performance degradation, if necessary SOF orchestrates additional end-to-end and per Service Component testing: SOF-sp -> Presto -> ICM (also SOF-sp -> Interlude -> SOF-partner for partner components)
4. SOF coordinates and tracks the resolution of SLS related degradations, including reconfiguring, reassigning, and / or rerouting aspects of the Service.
5. SOF also indicates if manual intervention is required to resolve the degradation, and tracks the status of any associated activities to help determine resolution status.
6. The status of SLS degradation resolution is reported to the Business Applications: SOF-sp -> Legato -> Business Applications
7. The status of SLS degradation resolution is reported to the Customer:
Business Applications -> Cantata -> Customer (Alternative, SOF could provide updates via Allegro)

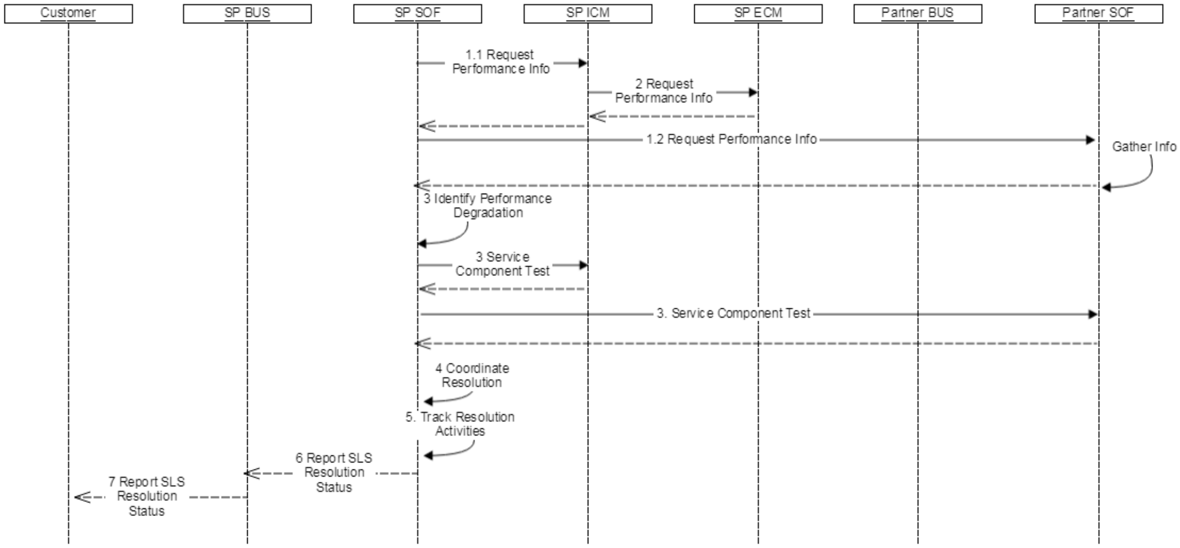


Figure 8 – Assessing Service Quality Based on SLS Sequence Diagram

10.7 Collection and Reporting of Billing and Usage

10.7.1 Purpose:

The Service Provider gathers relevant usage measurements and usage events in order to generate and provide a bill to the Customer as shown in Figure 9. LSO collects usage measurements, traffic measurements, and service-related usage events (e.g., Customer initiated changes in service bandwidth, etc.) describing the usage of Service Components and associated resources. This information is correlated to specific Services and Product Instances. The appropriate business applications perform rating and billing based on the usage information and business rules. Where Service Components have been used beyond their SLS commitments (e.g., counting yellow traffic that provides an opportunity to upsell the customer), exception reports may be generated. Note that Partner domains may also be involved in reporting usage and generation of billing information.

10.7.2 Asynchronous Event-Driven Steps:

i. SOF reports service usage events to business applications: SOF-sp -> Legato -> BUS
ii. SOF reports SLS violations and beyond SLS exceptions to business applications: SOF-sp -> Legato -> BUS

10.7.3 Steps:

1. SOF requests current traffic and usage information related to the interfaces, forwarding constructs and network functions related to the service instance. Note: Usage can be delivered as scheduled reports:
    1. SOF requests traffic and usage information of network functions and forwarding constructs across each internal forwarding domain: SOF-sp -> Presto -> ICM
    2. SOF requests traffic and usage information of Service Components in each external forwarding domain: SOF-sp -> Interlude -> SOF-partner

2. If not already cached, each ICM determines the elements involved and requests traffic and usage information for the network functions and forwarding construct across each element: ICM -> Adagio- > ECM
3. Once the traffic and usage information for the Service Components supporting the Service are successfully gathered, SOF analyzes the information for specific Service instances.
4. SOF reports traffic and usage summary to business applications: SOF-sp -> Legato -> BUS
5. Business applications perform rating and billing based on the usage information and business rules
6. Bill is provided to Customer: BUS-sp -> Cantata -> Customer

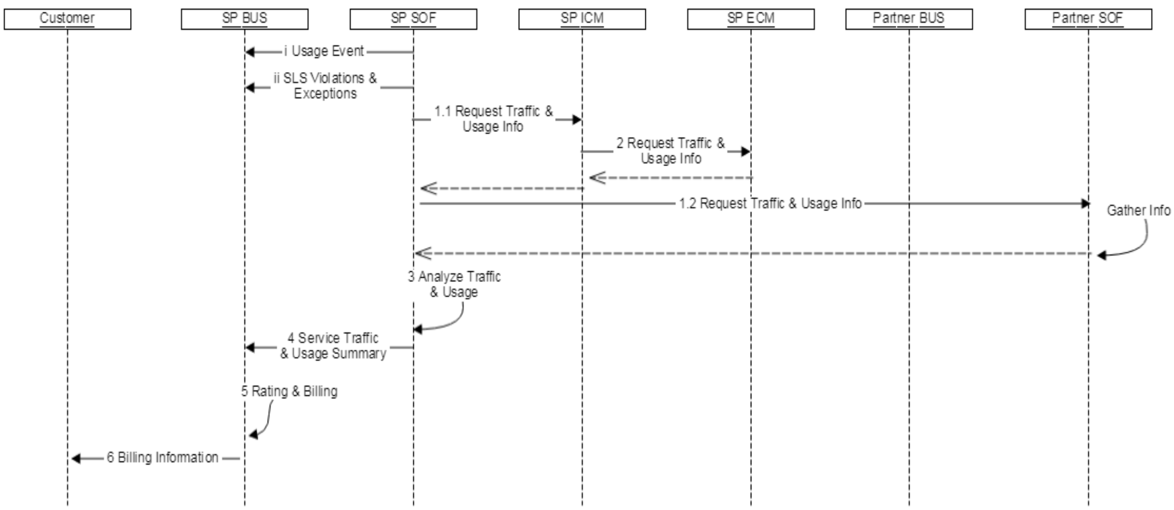


Figure 9 – Collection and Reporting of Billing and Usage Sequence Diagram

For dynamic service attribute modifications, the Service Provider collects historical information for SLS parameters and provide a bill to the Customer.

10.8 Managing Licenses

The Service Provider needs to manage licenses associated with resources supporting its services. License management functional blocks that the SP needs to support are:

- End-to-end License Coordination with Vendors and Partners: End-to-end license coordination is expected to be performed by OSS/BSS through its Sonata and Legato interfaces.
- License Manager: License manager ensures that all license categories related to a resource such as a VNF including Vendor specific, Open Source, co-developed, home-made, etc., are maintained within License Management. License Manager APIs hide the vendor specific APIs by providing a standardized interface which is vendor agnostic.
- License Policy Enforcement: Runtime implementation of License policies is handled by SOF. SOF interfaces License Manager for the Policy rules.
- License Usage Charging and Billing: Usage events are collected by SOF and passed them to OSS/BSS for charging and billing.

Most of the licenses don’t need to be managed during the service, therefore, their management is not an LSO function. On the other hand, some of the licenses such as those for VNFs need to be managed during the service. Their management is an LSO function.

License management process for VNFs will follow the processes described in Sections 10.2, 10.3 and 10.7.

11 LSO Management Abstractions and Constructs

In this section, LSO Management Abstractions and constructs are described in terms of information categories and high-level information classes including sample properties (e.g., attributes and associations), while the detailed logical information model will be documented by MEF. These abstractions and constructs define a common technology independent representation of connectivity, applications, topology and infrastructure, while providing the means to extend the model with technology specific details in a semantically rich fashion (including MEF specific service attributes). This will help ensure that the LSO functionality and information is developed in a logically consistent fashion, allowing Service Providers to readily integrate such capabilities into their management environments.

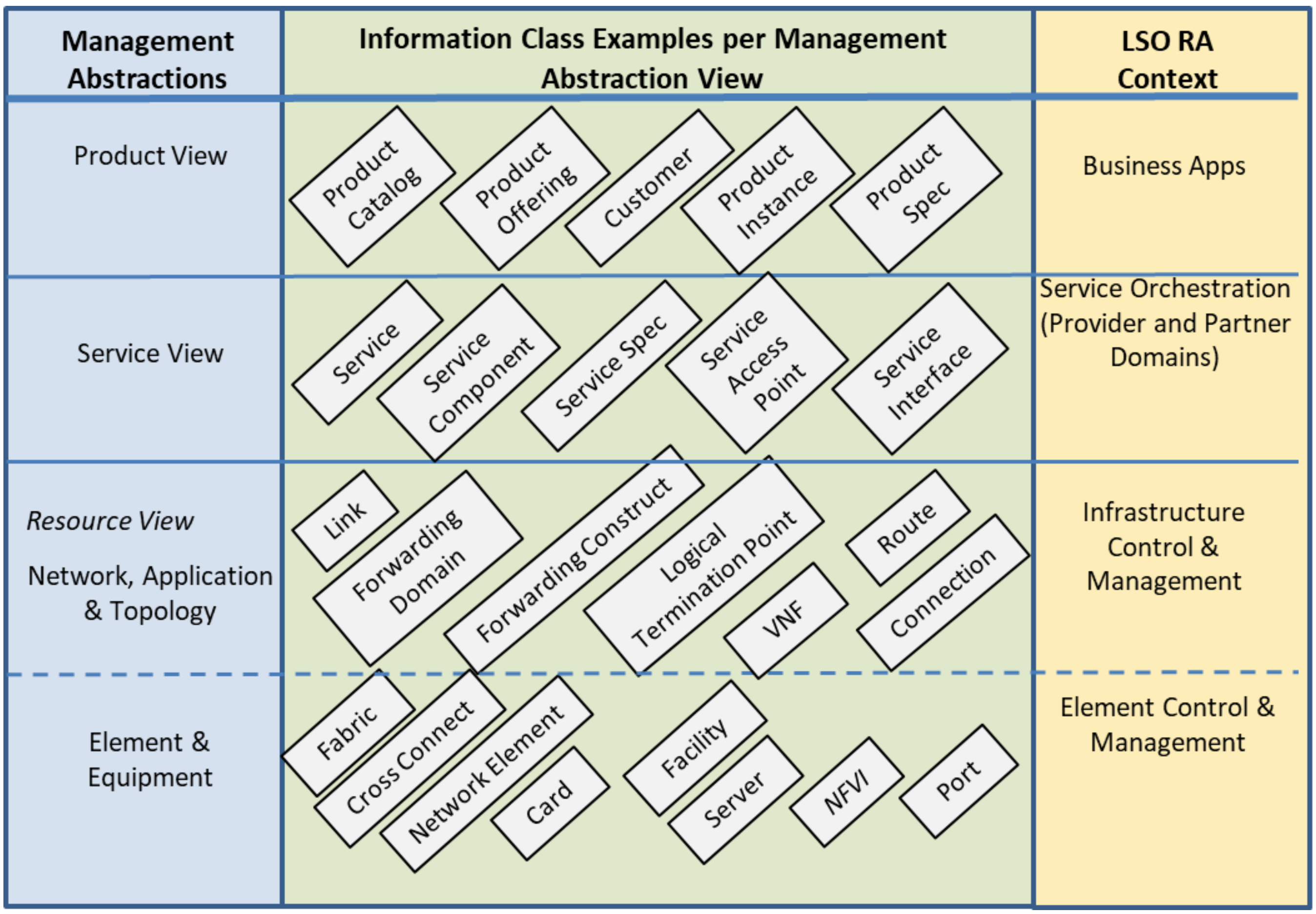


Figure 10 – A View of Management Abstractions

Figure 10 shows the different Management Abstractions in the context of LSO, along with some example information classes.

There are three main abstracted management views in the LSO environment:

- **Product View:** The product domain is specific to the interaction between the Customer and the Product Offerings of a Service Provider. The Product Instance involves the purchasing, or procurement, of specific Product Offerings from a product catalog by a Customer, and all other commercial aspects related to the Customer's Product Instance, such as billing. Product Specifications define the individual product characteristics that are used to create differentiated Product Offerings. Software systems implementing Product Offering and Product Instance related functionality have traditionally been business support systems in the business management domain.

- **Service View**: A Product Instance is realized as one or more Services and associated resources; thus Services are tightly bound to Product Instances and may be viewed to represent the Customer experience of the Product Instance that has been realized within the Service Provider's infrastructure. A Service is visible and directly usable by the Customer, but may be divided within the Service Provider's infrastructure into one or more Service Components, for instance corresponding to forwarding domains at the resource layer or to underlying access services that the Service Provider has purchased from a Partner domain. Service Components are not visible to the Customer. Software systems implementing service-related functionality have traditionally been operational support systems in the service management domain or service management systems.

  Note that in the TM Forum SID (TMF GB922 [34]), a Service is referred to as a Customer Facing Service (CFS) and a Service Component is referred to as a Resource Facing Service (RFS).
- **Resource View***:* Services are delivered via resources in the network, whether physical or logical. Physical resources are actual hardware, and logical resources can be viewed as functionality provided by specific pieces of hardware. The resource view can be further sub-divided into the Network, Application and Topology View; and the Element and Equipment View. The Network, Application and Topology View encompasses all the functions across network and application elements, on the basis of administrative network domains. The Element and Equipment View pertains to the management of a specific set of devices. Software systems implementing Network, Application and Topology View functionality have traditionally been operational support systems in the network management domain or network management systems. The Element and Equipment View focuses on the physical and logical resources within a single network element, or group of similar network elements. Software systems implementing Element and Equipment View functionality have traditionally been operational support systems in the element management domain or element management systems.

Each of these management views is further described in the following subsections.

11.1 Product View Abstractions

Customers need to be able to express their needs in order to determine which Product Offerings can support their requirements and Service Providers need to be able to match these requirements to technical specifications to realize the Product Offering. A Product Offering represents what is externally presented to the market for the market's use. It can be assembled from a reusable Product Specification which describes characteristics of the Product Offering that are made externally available, both tangible and intangible objects. A product catalog contains a list of Product Offerings for sale, with prices and illustrations, for example in book form or on the web. A Product Instance represents the subscription of a Product Offering by a Customer, who normally is the purchaser of the Product Instance. Thus, the Product Instance is the instantiation of a Product Offering for a given Customer.

The Product Specifications can be used by Service Providers to create differentiated Product Offerings. For example, for Carrier Ethernet these specifications may define traditional E-Line, E-LAN, and E-Tree product characteristics for EVC based services, as well as specialty E-Access and E-Transit characteristics for OVC based services. These Product Specifications will define the characteristics of UNI / ENNI service interfaces, the EVC / OVC as Connectivity Services, and the associated service access points, or endpoints of the connection.

For the most part, these product characteristics will map 1-to-1 to the service characteristics found in a Service Specification in the Service View, and, in the case of Ethernet Services, reflect the service attributes found in MEF 6.3 [11] , MEF 10.4 [13], MEF 51.1[23], and MEF 26.2 [17] technical specifications. The linkage from the Product View and the Service View is precisely through the Product Specification to the Service Specification, and from the Product Instance to the Service.

Table 2 and Table 3 below show an example of part of a Product Offering definition, e.g., "Super Metro Ethernet Line" being offered by Service Provider "World Telco". In this case, the Product Offering corresponds to an EPL service.

Note that the definition of the Product Offering is applicable to ALL Product Instances that are created by the Service Provider.

UNI Product Characteristics	Product Characteristic Value
Product Offering Name	"Super Metro Ethernet Line"
Physical Layer	10M Full Duplex, 100M Full Duplex, 10/100M Auto-Negotiation, 1 G Full Duplex, or 10G Full Duplex
Service Multiplexing	“None”
Bundling	“None”
Max Frame Size	“1522”
All to One Bundling	“Enabled”
Max number of EVCs	“1”
Etc….	

Table 2 – Example Definition of UNI Product Characteristics in a Product Offering

EVC Product Characteristics	Product Characteristic Value
Product Offering Name	"Super Metro Ethernet Line"
EVC Type	"P2P"
MaxNumUNIs	"2"
CE Vlan Id Preservation	"True"
CE Vlan Cos Preservation	"True"
Unicast Service Frame Delivery	"Unconditional"
Broadcast Service Frame Delivery	"Unconditional"
Etc….	

Table 3 – Example Definition of EVC Product Characteristics in a Product Offering

11.2 Service View Abstractions

The service represents the intent of the Service Provider to deliver the features as specified in the Customer's Product Instance. For example, in the case of Carrier Ethernet, the Service may be a UNI-to-UNI EVC based service offered by a Service Provider, or a UNI-to-ENNI, ENNI-to-ENNI OVC based service offered by an operator. A Service may be divided into one or more Service Components, each representing a portion of the end-to-end connectivity that traverses a single administrative domain. If, for example, a Service Provider buys an OVC from an Operator in order to provide an end-to-end EPL Service to the Customer, the Service Provider and the Operator may have different perspectives on the OVC. Within the Service Provider's management system, the OVC is viewed as a Service Component of the end-to-end EPL Service, whereas within the Operator's management system, the OVC is viewed as the Service. These different perspectives are illustrated in Figure 11. In Figure 11, interfaces at the boundaries between different parts of a Service Provider's internal network are labeled 'INNI'.

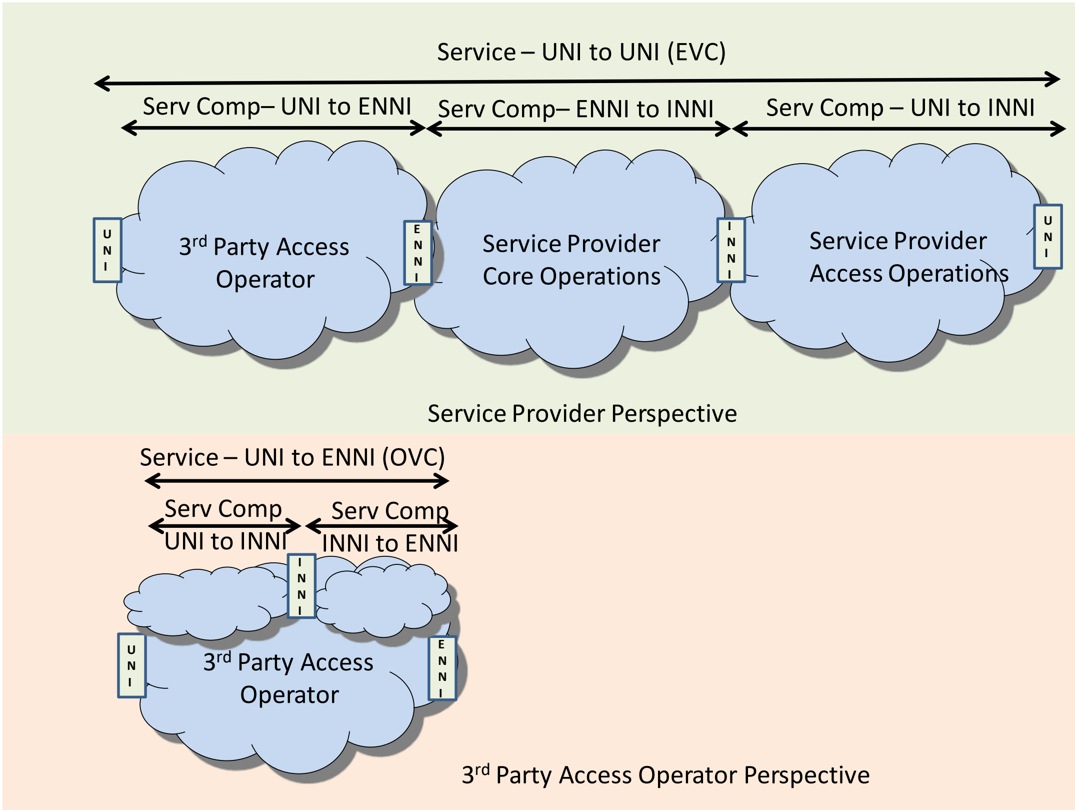


Figure 11 – Example of Service View Abstraction

11.3 Resource View Abstractions

The Resource View is comprised of the Network, Application, and Topology View; and Element and Equipment View abstractions. The next two subsections use the ONF Core Model (ONF TR-512.1 [31]) to describe an example of Network and Topology View and Element and Equipment View abstractions.

11.3.1 Network, Application and Topology View Abstractions

The Network Control Domain represents the scope of control that a particular Network Domain Controller or WAN controller has with respect to a particular network, i.e., encompassing a designated set of interconnected (virtual) network elements. The topology of the network may be defined based on Forwarding Domains (FDs) and Links, which represent adjacency between FDs. The FD is the topological component which represents the opportunity to enable forwarding between points represented by Logical Termination Points (LTPs). The LTP encapsulates the termination, adaptation and OAM functions of one or more transport layers.

The FD contains instances of Forwarding Constructs (FCs) of one or more layer networks (e.g., OCh, ODU, ETH, and MPLS), thus defining the transport for any given Service. The FD provides the context for instructing the formation, adjustment and removal of FCs. The FD supports recursive aggregation such that the internal construction of an FD can be exposed as multiple lower level FDs and associated Links (partitioning).

The FC effects forwarding of transport characteristic (layer protocol) information between two or more LTPs. The association of the FC to LTPs is made via Endpoints (essentially the ports of the FC).

An FC supports recursive aggregation such that the internal construction of an FC can be exposed as multiple lower level FC objects (partitioning). An FC can have zero or more Routes, each of which is defined as a list of lower level FCs.

The FC can represent many different structures including point-to-point (P2P), point-to-multipoint (P2MP), rooted-multipoint (RMP) and multipoint-to-multipoint (MP2MP) bridge and selector structure for linear, ring or mesh protection schemes.

11.3.2 Element and Equipment View Abstractions

The Network Element represents a network device in the data plane or the virtual infrastructure for the network element. In the direct interface from an SDN controller to a network device in the data plane or from a Virtual Infrastructure Manager (VIM) to NFVI, the Network Element defines the scope of control for the resources within the network element, e.g., internal transfer of user information between the external terminations such as ports/UNIs or Connection End Points, encapsulation, multiplexing /demultiplexing, and OAM functions, etc. The Network Element provides the scope of the naming space for identifying objects representing the resources within the Network Element.

12 References

[1] ECMA, “The JSON Data Interchange Format, ECMA-404”, October 2013.

[2] ETSI, “Network Functions Virtualisation (NFV); Management and Orchestration, ETSI GS NFV-MAN 001, V1.1.1”, December 2014.

[3] ETSI, “Network Function Virtualisation: Architectural Framework”, ETSI GS NFV 002 v1.2.1 (2014-12)”.

[4] ETSI, “Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV, ETSI GS NFV 003, V1.2.1, December 2014.

[5] ETSI, “Network Functions Virtualisation (NFV) Release 3; Licensing Management; Report on License Management for NFV, ETSI GR NFV-EVE 010 V3.1.1”, December 2017.

[6] IETF, “On the Difference between Information Models and Data Models”, RFC 3444, January 2003.

[7] IETF, “The JavaScript Object Notation (JSON) Data Interchange Format”, RFC 7159, March 2014.

[8] IETF, “Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing”, RFC 7230, June 2014.

[9] MEF 3.0 Overview- https://www.mef.net/mef30/overview

[10] MEF Forum, Metro Ethernet Network Architecture Framework - Part 1: Generic Framework, MEF 4, April 2004.

[11] MEF Forum, Subscriber Ethernet Services Definitions, MEF 6.3, November 2019

[12] MEF Forum, Carrier Ethernet Service Information Model, MEF 7.3, February 2017.

[13] MEF Forum, Subscriber Ethernet Service Attributes, MEF 10.4, December 2018.

[14] MEF Forum, User Network Interface (UNI) Requirements and Framework, MEF 11, November 2004.

[15] MEF Forum, Requirements for Management of Metro Ethernet Phase 1 Network Elements, MEF 15, November 2005.

[16] MEF Forum, Service OAM Framework and Requirements – Phase 1, MEF 17, April 2007.

[17] MEF Forum, External Network Network Interfaces (ENNI) and Operator Service Attributes, MEF26.2, August 2016.

[18] MEF Forum, Service OAM Fault Management Implementation Agreement Phase 2, MEF 30.1, April 2013.

[19] MEF Forum, Service OAM Performance Monitoring Implementation Agreement, MEF 35.1, May 2015.

[20] MEF Forum, Carrier Ethernet Service Lifecycle Process Model, MEF 50, December 2014.

[21] MEF Forum, MEF Services Lifecycle Process Flows, MEF 50.1, August 2017

[22] MEF Forum, Operator Ethernet Service Definitions, MEF 51.1, December 2018

[23] MEF Forum, IP Service Attributes, MEF 61.1, May 2019.

[24] MEF Forum, Subscriber Layer 1 Service Attributes, MEF 63, August 2018.

[25] MEF Forum, SD-WAN Service Attributes and Services, MEF 70, July 2019.

[26] OASIS BPEL, Web Services Business Process Execution Language Version 2.0. OASIS Standard. 11 April 2007. http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html.

[27] OASIS TOSCA Simple Profile for Network Functions Virtualization (NFV) Version 1.0, 17 March 2016. http://docs.oasis-open.org/tosca/tosca-nfv/v1.0/tosca-nfv-v1.0.pdf.

[28] OASIS TOSCA 1.0, http://docs.oasis-open.org/tosca/TOSCA/v1.0/os/TOSCA-v1.0-os.pdf , November 2013.

[29] Object Management Group, “OMG Unified Modeling Language TM (OMG UML)””, Version 2.5, March 2015.

[30] Open Networking Forum, SDN Architecture Overview, TR-504, Version 1.1, November 2014.

[31] Open Networking Forum, “Core Information Model (CoreModel)”, TR-512.1, Version 1.3.1, January 2018.

[32] TM Forum, “Business Process Framework (eTOM), Process Decompositions and Descriptions, GB921 Addendum D, Release 15.0.0”, October 2015.

[33] TM Forum, Business Process Framework (eTOM), Business Process Framework Primer, GB921 Addendum P, Release 14.5.0, November 2014.

[34] TM Forum, Information Framework (SID), GB922, Release 15.0.0, September 2015.

[35] TM Forum, Frameworx Standard, Application Framework, The Digital Services Systems Landscape, Addendum D, Release 14.5.1, March 2015.

[36] TM Forum, License Management, IG1143 Release 16.5.1, March 2017.

[37] W3C, Extensible Markup Language (XML) 1.0 (Fifth Edition), November 2008.

Appendix A Management Interface Reference Point Examples (Informative)

The LSO Management Interface Reference Points portray points of interaction between LSO functional management entities in the LSO reference architecture. To help characterize the behavior of each LSO Management Interface Reference Point, this appendix provides informative examples of high-level interactions.

LSO Management Interface Reference Point	**High Level Interaction Examples (non-exhaustive)**
Cantata **(CUS:BUS)**	*Supports Product related management interactions between the Service Provider's Business Applications and the Customer Application Coordinator.* Customer Application Coordinator browses the product catalog for Product Offerings that are available for the Customer to select. Based on Product Offerings, Customer Application Coordinator develops, places, tracks, and changes Product Orders. Customer Application Coordinator requests modification of Product Instances. Customer Application Coordinator receives information about the scheduled maintenance that may impact their Product Instances. Customer Application Coordinator places and tracks trouble reports. Customer Application Coordinator queries and views usage and billing information.

LSO Management Interface Reference Point	High Level Interaction Examples (non-exhaustive)
Allegro (CUS:SOF)	*Supports Service-related management interactions between the Customer Application Coordinator and the Service Provider's Service Orchestration Functionality.* Customer Application Coordinator controls Service by requesting changes to dynamic parameters as permitted by service policies. Customer Application Coordinator queries operational state of the Service. Customer Application Coordinator requests change to administrative state or permitted attributes of a Service. Customer Application Coordinator provides and views customer acceptance testing information. Customer Application Coordinator views Service performance and fault information. Customer Application Coordinator receives Service specific event notifications from the Service Provider. Customer Application Coordinator receives Service specific performance information from the Service Provider. Customer Application Coordinator request test initiation and receive test results from the Service Provider.
Sonata (BUS:BUS)	*Supports Product related cross domain interactions between the Service Provider's Business Applications and the Partner's Business Applications.* Service Provider browses the Partner's product catalog (e.g., wholesale catalog) for Product Offerings that are available for the Service Provider to select. This may include some geographical and Service information to support availability queries of a Product Offerings at some geographical area. Service Provider develops (based on Product Offerings), places, tracks, and changes Product Orders with the Partner Service Provider requests modification of Product Instances. Service Provider receives Product Instance performance and fault information provided by the Partner. Service Provider receives information from the Partner about the scheduled maintenance that may impact their Product Instances. Service Provider places and tracks trouble reports. Service Provider exchanges usage and billing information.

LSO Management Interface Reference Point	High Level Interaction Examples (non-exhaustive)
Interlude (SOF:SOF)	*Supports control related management interactions between the Service Provider and the Partner.* Service Provider controls aspects of the Service within the Partner domain (on behalf of the Customer) by requesting changes to dynamic parameters as permitted by Service policies. Service Provider queries operational state of the Service. Service Provider requests change to administrative state or permitted attributes of a Service. Service Provider request creation of connectivity between two Service Interfaces as permitted by established business arrangement. Service Provider queries the Partner for detailed information related to Services provided by the Partner to the Service Provider. Service Provider receives Service specific event notifications from the Partner. Service Provider receives Service specific performance information from the Partner. Service Provider request test initiation and receive test results from the Partner.
Legato (BUS:SOF)	*Supports interactions between the Business Applications and the Service Orchestration Functionality.* Business Applications request Service feasibility determination. Business Applications request reservation of resources related to a potential Service. Business Applications request activation of Service. Business Applications receive Service activation tracking status updates. Business Applications receive request to initiate Product Order with a Partner provider (for off net portions of the service). Business Applications receive usage events due to a Customer initiating dynamic activity on their Service (e.g., increase in bandwidth). Business Applications receive a summary of Service quality and usage information. Business Applications receive Service Activation Testing results. Business Applications receive capability information about the Service layer.

LSO Management Interface Reference Point	High Level Interaction Examples (non-exhaustive)
Presto (SOF:ICM)	*Supports the management of the network infrastructure, including network and topology view related management functions*. SOF requests ICM to create network connectivity or functionality associated with specific Service Components of an end-to-end Connectivity Service within the domain managed by each ICM SOF receives topology, connectivity and routing information from ICM SOF receives performance and fault information from ICM. SOF queries ICM for Resource Inventory (including capabilities) information.
Adagio (ICM:ECM)	*Support the management of discrete network resources, including element view related management functions.* ICM requests implementation of cross-connections or network functions on specific elements via the ECM functionality responsible for managing the element. ICM requests the change in administrative state of specific resources management by the ECM. ICM discovers element level configuration information from the ECM. ICM receives element level fault and performance information from ECM.

Table 4 – Examples of High-Level Interactions per LSO Management Interface Reference Point

Appendix B Relation of LSO Functional Areas to MEF 50 (Informative)

The LSO Reference Architecture and Framework segments the functional requirements into sections within the document based on the functional area covered by each set of requirements. This appendix provides a mapping of LSO reference architecture and framework requirements functional areas to MEF 50 [20] related process flows.

LSO Requirements Functional Area	Related MEF 50 Process Flows
Agile Product / Service Design	Product Design; Service and Resource Design;
Order Fulfillment Orchestration	Sales Proposal and Feasibility; Capture Customer Order
Service Control Orchestration	(no mapping)
Service Configuration and Activation Orchestration	Service Configuration and Activation
Service Delivery Orchestration	Service Configuration and Activation
Service Testing Orchestration	End-to-End Service Testing
Service Problem Management	Service Problem Management
Service Quality Management	Service Quality Management
Billing and Usage Measurements	Billing and Revenue Management
Security Management	(no mapping)
Analytics	Service Quality Management
Policy-based Management	(no mapping)
Customer / Partner Management	Establish Relationship between Service Provider and Access Provider; Terminate Customer Relationship
License Management	(no mapping)

Table 5 – Mapping of LSO Reference Architecture and Framework Functional Areas to MEF 50 Related Process

Appendix C TOSCA Service Templates (Informative)

This section focuses on TOSCA templates. The TOSCA specification OASIS BPEL2.0 [26] provides an approach for defining the relationships between Customers, Service Providers and developers of IT services, and can be applied to many Service types, including connectivity Services. A Service Provider can use a TOSCA Service Template to offer and orchestrate the deployment of Services for Customers. The use of a standardized interface with automation tools enables easier interaction between Customers and Service Providers/Partners.

A TOSCA Service Template is a combination of a TOSCA Topology Template and one or more TOSCA Orchestration Plans as shown in Figure 12 below. The topology includes the TOSCA Components (i.e., TOSCA Node Templates) and relationships between them as described in TOSCA Relationship Templates.

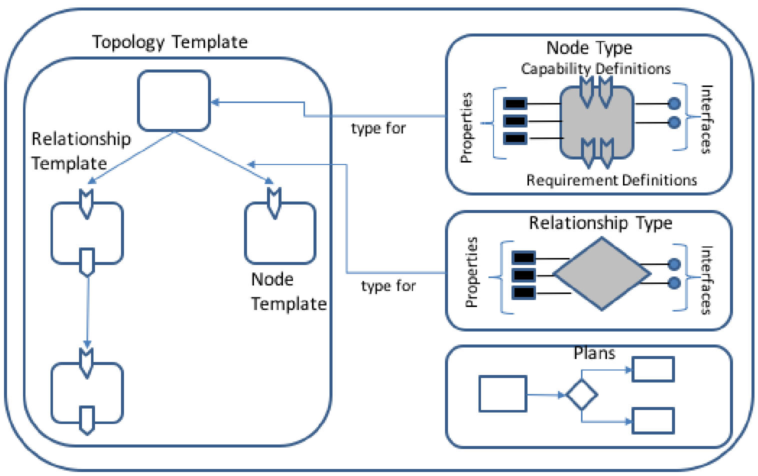


Figure 12 – TOSCA Service Template [OASIS BPEL 2.0 [26]]

The Relationship between TOSCA Nodes, as a graph, can describe the associations between a given Service instance and one or more Service Components, or, between a given Service Component and one or more Resources. For example, Service Components can be Classes of objects and when instantiated into Service instances, the TOSCA Topology Template can be used to show either a vertical relationship between Service and Resource layers or horizontal relationship within a given layer as shown in Figure 13.

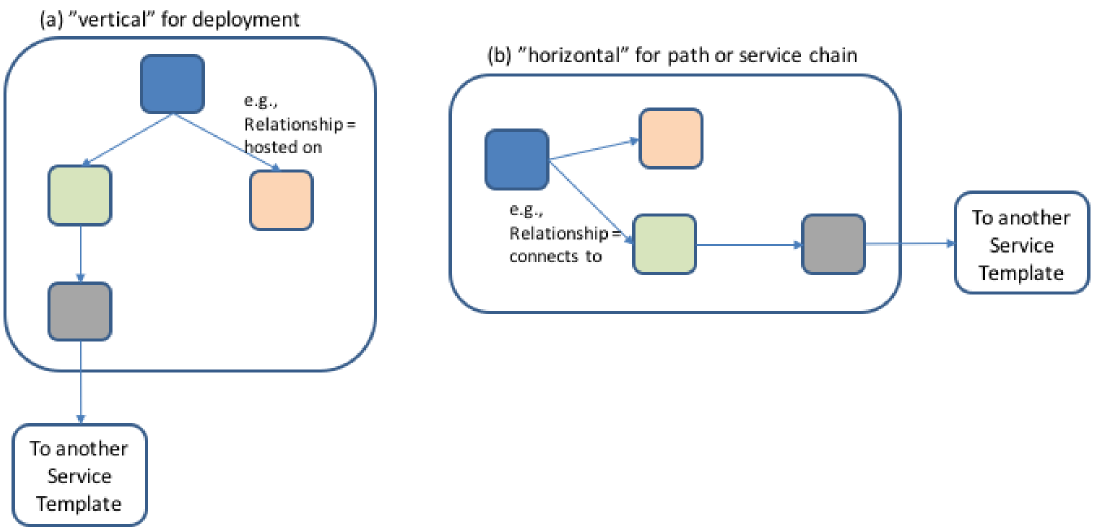


Figure 13 – Use of TOSCA Topology Template

One example of a TOSCA Topology is in OASIS TOSCA [27]. The TOSCA Topology graph can also be used in a Topology API Service to client systems for sharing details such as graph of Service Components only or include more details such as graph of Resources for a given Service instance. TOSCA Service Templates could be nested, i.e., a TOSCA Node within a TOSCA Service Template can be another TOSCA Service Template.

TOSCA Plans explicitly define which Nodes will be deployed and how they will be connected. TOSCA Plans are typically expressed as Business Process Execution Language (OASIS BPEL 2.0 [26], OASIS NFV [27]) or shell scripts.

TOSCA Service Templates in MEF LSO RA

One possible use of TOSCA Service Templates is in the Service Orchestration Function (SOF) entity of MEF LSO Reference Architecture (Figure 2) for the orchestration of Service Components across one or more domains. The templates can also be used in an Infrastructure Control and Management (ICM) entity for the orchestration of Resources within a domain, in support of one or more Service Components. Additionally, as discussed in Section 11.1, a Product Offering in a Product catalog would represent what is externally presented to the market with appropriate mapping to one or more Services. Thus, a Product can be represented as a TOSCA Node Type and/or TOSCA Service Template for marketable entities.

TOSCA Service Templates are defined more broadly than MEF Services and can be used to describe more than just MEF Service Components or Forwarding Domains. They include metadata to orchestrate the Service lifecycle via the TOSCA Plans, i.e., 'how' to deploy a Service, whereas, a MEF Service describes the behavior as seen by an external observer, i.e., the 'what'. In other words, a MEF Service can be shown as a TOSCA Node Type in a TOSCA Topology view and used, as example for a Virtual Link, in the TOSCA Simple Profile for NFV [OASIS TOSCA [27]]. Additionally, a MEF Service can be made up of a hierarchy of TOSCA Nodes and TOSCA Relationships as shown in Table 6. However, while TOSCA Plans provide for orchestration of the Service lifecycle, Operational Threads and Policy based Management are also required to address behaviors for the entire lifecycle of the Service.

One key value of using TOSCA Service Templates is interoperability with structure of the Service, i.e., Service composition/definition, and deployment lifecycle. TOSCA Service Templates could be part of the Service Catalog in a SOF or Resource Catalog in an ICM. Some TOSCA Service Templates in a Service Catalog can be exchanged with other Operators across Sonata (via Business Applications entity) or Interlude reference points to request a specific deployment configuration. Standardized TOSCA Service Templates as part of a Service Catalog, for example, can enable interoperable definitions as well as implementations. Standard TOSCA Service Templates can also allow for Service composition when components are deployed in different domains within a Service Provider or different Operators. Service composition can be dynamic, e.g., at the time of request for a given Service, with choice of suitable TOSCA Node Templates and TOSCA Relationship Templates. Both the structure of the Service as well as the TOSCA Plans (or Business Process) could help with interoperability when Services span multiple domains.

Section 11 and Section 11.2 describe the Service View as consisting of one or more Service Components within the Service Provider's infrastructure. Some of the Service Components in a Service Provider's management system, for example an OVC, maybe viewed as a Service by an Operator. Additionally, Section 8.2.2 has a number of requirements related to Service configuration and activation including topology, determining the necessary Service Components and configurations, etc.

In particular, Section 8.2.4 identified the need for coordinated execution of the Service delivery orchestration plan for Service Components implementation. To enable this coordinated execution, a TOSCA Service Template can be useful since it provides the required set of Service Components, Resources and the relationship as well as dependencies for a given Service instance.

Example – TOSCA Topology Model for MEF Services

An example TOSCA Topology Model is discussed for illustration purposes only. Formal definition of TOSCA Topology Template, TOSCA Node Template and TOSCA Relationship Template is outside the scope of this document.

A topology model for a MEF Service can be for the Service View as well as the Resource View and the dependencies between them. As example, the TOSCA Node Types for TOSCA Topology of MEF Services can be as shown in Table 6 and can be related directly to the classes or objects in MEF Information Model specifications.

<table>
<tr><th>TOSCA Topology</th><th>Example TOSCA Node Types</th><th>Reference</th></tr>
<tr><td>Service View</td><td>1. MEF CE Service
2. EVC
3. EVC End Points
4. UNIs
5. OVC
6. OVC EndPoints
7. ENNIs</td><td>MEF 7.3 [12]Classes in Figure 14</td></tr>
<tr><td>Resource View</td><td>1. FCs
2. LTPs
3. Ports
4. VNFs</td><td>ONF TR-512.1[31]
OASIS TOSCA [28]</td></tr>
<tr><td>Facilities View</td><td>Equipment, Media (e.g., Fiber)</td><td></td></tr>
</table>

Table 6 – Example TOSCA Node Types for MEF Services

Node Types can be based on individual Service attributes like EVC Type instead of the entire EVC class. Such granularity could help with dynamic Service composition by choosing suitable TOSCA Node Types and TOSCA Relationship Types. The Node Type definitions are left for further study.

For MEF Services, the TOSCA Topology view is a graph with Relationships between the TOSCA Node Types such as those shown in Table 6. The association between Service Components or the dependencies to Resources can be identified with the TOSCA Relationship Templates. As example, TOSCA Topology can be for a Product that might be offered as a bundle of multiple MEF Services (e.g., EVPL-1, EVPL-2 and EVPLAN-3) with TOSCA Relationships between TOSCA Node Types for Product and MEF Services. In this example, MEF Services might be for different use cases of a given Customer, e.g., EVPL-1 for connecting Head Office to a Backup Site, EVPL-2 for connecting Head Office to peering with Cloud Provider, and, EVPLAN-3 for intranet. Each TOSCA Node for MEF Service can then shown to be with TOSCA Relationship to TOSCA Node(s) for Service Components.

Specifically, in the context of MEF 55.1, the TOSCA Topology description for Product, e.g., HQ Hub with three MEF Services, could be used at the Business Applications Layer for use across Cantata or Sonata reference points. Likewise, the TOSCA Topology description for each MEF Service could be in Service Orchestration Function for use across Legato reference points. The TOSCA Topology for Product to Service is as shown in Figure 14 below:

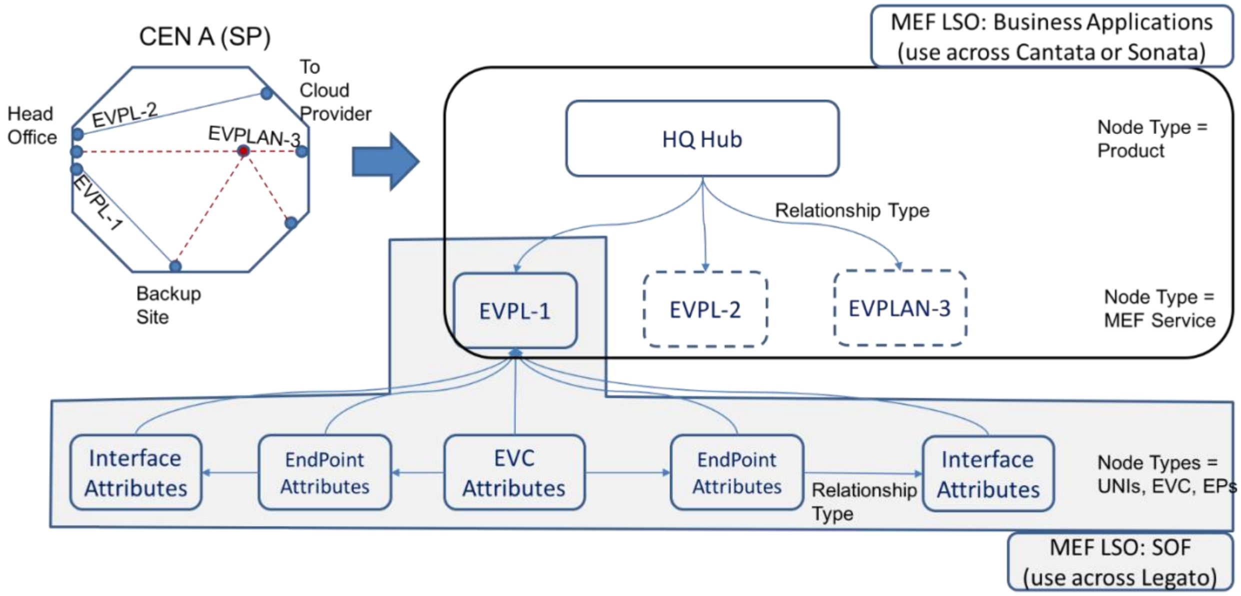


Figure 14 – Example TOSCA Topology for Product to Service View

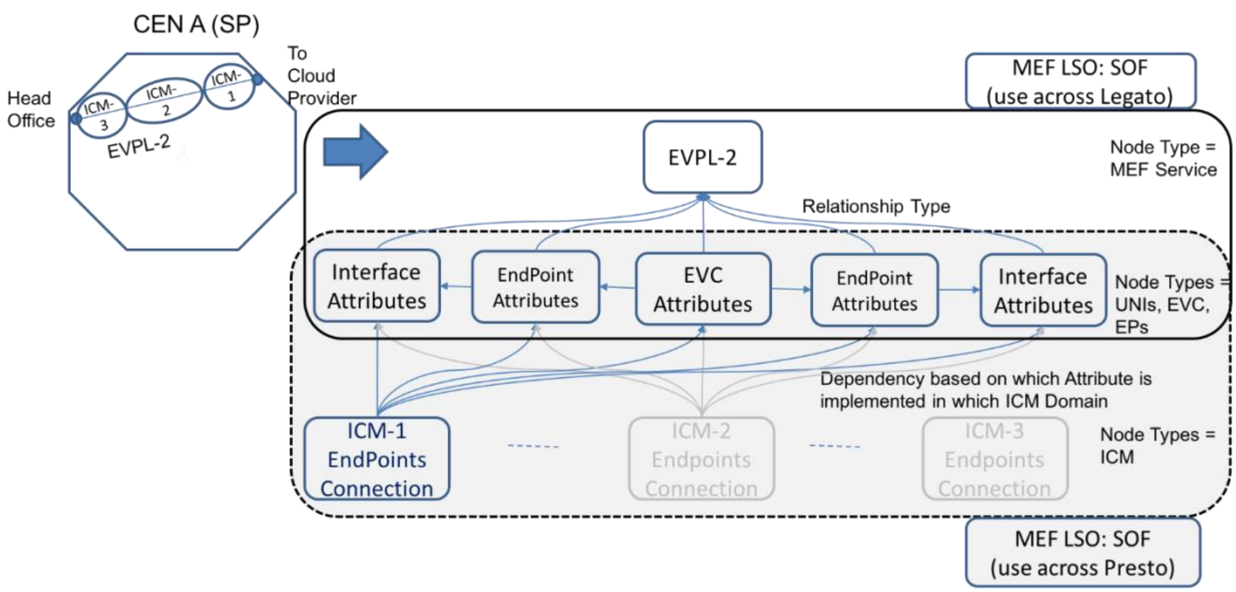


Figure 15 – Example TOSCA Topology for Service to Domain View

CEN A might have one or more ICM Domains. For example, EVPL-2 might traverse three ICM domains. In addition to use of TOSCA description across the Legato reference point, SOF can express relationship of Service Components to a given ICM Domain as shown in Figure 15.

When MEF Service is across more than one Operator domain, e.g., 2 CEN Operators, then, the TOSCA Topology view can be shown for each Service Component (OVC) in the EVC as in Figure 16 below. For example, EVPL-1 is shown to use OVC-A across CEN A and OVC-B across CEN B. With CEN A's SOF performing Service decomposition, TOSCA Topology descriptions can be used across Legato/Presto reference points within CEN A and across Sonata/Interlude reference points for requesting Service Components in CEN B. Figure 16 shows the SOF using Business

Applications for sending the initial request for OVC-B related attributes via Sonata while using Interlude for Service Control.

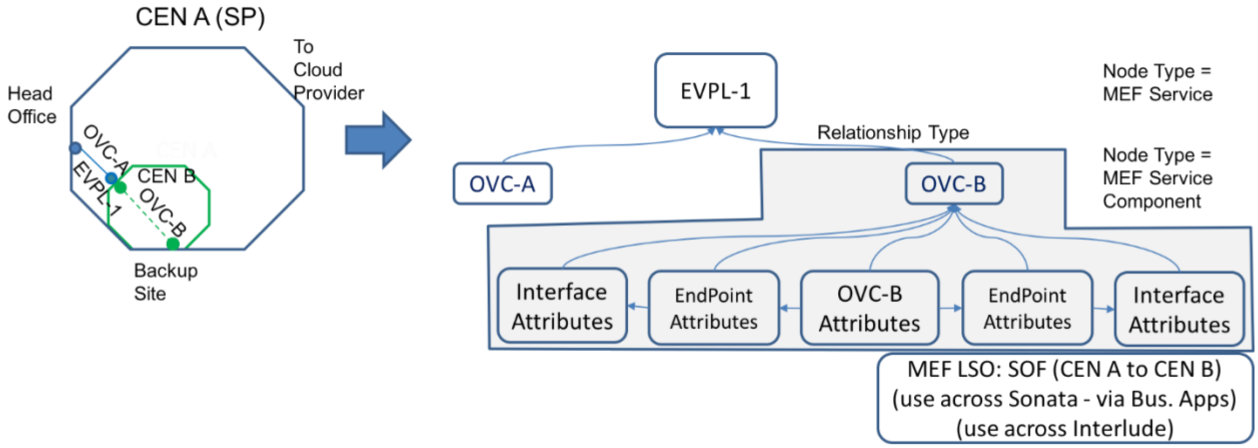


Figure 16 – Example TOSCA Topology for Service View across 2 Operators