The Consultative Committee for Space Data Systems

Report Concerning Space Data System Standards

CROSS SUPPORT CONCEPT— PART 1: SPACE LINK EXTENSION SERVICES

Informational Report

CCSDS 910.3-G-3

Green Book
March 2006

AUTHORITY

Issue:	Informational Report, Issue 3
Date:	March 2006
Location:	Washington, DC, USA

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and reflects the consensus of technical panel experts from CCSDS Member Agencies. The procedure for review and authorization of CCSDS Reports is detailed in the *Procedures Manual for the Consultative Committee for Space Data Systems*.

This document is published and maintained by:

CCSDS Secretariat
Office of Space Communication (Code M-3)
National Aeronautics and Space Administration
Washington, DC 20546, USA

FOREWORD

This document is a technical Report to assist readers in understanding the Space Link Extension (SLE) service documentation. It has been prepared by the Consultative Committee for Space Data Systems (CCSDS). The Cross Support concept described herein is the baseline concept for ground data communication within missions that are cross-supported between Agencies of the CCSDS.

This Report describes a common framework and provides a common basis for understanding the SLE services. It is intended to assist implementing organizations within each Agency to proceed coherently with the development of compatible derived Standards for the ground systems that are within their cognizance.

Through the process of normal evolution, it is expected that expansion, deletion or modification to this document may occur. This Report is therefore subject to CCSDS document management and change control procedures that are defined in reference [1]. Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

Questions relating to the contents or status of this document should be addressed to the CCSDS Secretariat at the address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- Agenzia Spaziale Italiana (ASI)/Italy.
- British National Space Centre (BNSC)/United Kingdom.
- Canadian Space Agency (CSA)/Canada.
- Centre National d’Etudes Spatiales (CNES)/France.
- Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Federal Space Agency (Roskosmos)/Russian Federation.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Belgian Federal Science Policy Office (BFSPO)/Belgium.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- Centro Tecnico Aeroespacial (CTA)/Brazil.
- Chinese Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Danish Space Research Institute (DSRI)/Denmark.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Korea Aerospace Research Institute (KARI)/Korea.
- MIKOMTEK: CSIR (CSIR)/Republic of South Africa.
- Ministry of Communications (MOC)/Israel.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic & Atmospheric Administration (NOAA)/USA.
- National Space Program Office (NSPO)/Taipei.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

Document	Title	Date	Status
CCSDS 910.3-G-1	Report Concerning Space Data System Standards: Cross Support Concept—Part 1: Space Link Extension Services	May 1995	Original issue, superseded
CCSDS 910.3-G-2	Report Concerning Space Data System Standards: Cross Support Concept—Part 1: Space Link Extension Services	April 2002	First revision, superseded
CCSDS 910.3-G-3	Cross Support Concept—Part 1: Space Link Extension Services, Informational Report, Issue 3	March 2006	Current issue

CONTENTS

CONTENTS (continued)

CONTENTS (continued)

1 INTRODUCTION

1.1 PURPOSE

Cross Support occurs when one organization provides part of its Space Data System resources to service the space data transfer requirements of another organization. This Report explains how CCSDS Space Link Extension (SLE) services are designed to facilitate Cross Support and to maximize the performance and cost benefits to be gained by using standard protocols.

This Report presents the concept of Cross Support using SLE services; describing the background to the services and presenting the functional components of the ground element of a Space Data System that provide the services. It describes the SLE Data Transfer and SLE Service Management interfaces where Cross Support may occur and provides example scenarios to show how the concept may be applied in practice.

1.2 SCOPE

This Report summarizes the technical content of the SLE Recommendations including the *Cross Support Reference Model* (reference [8]), the *SLE Service Management Service Specification* (reference [9]), and the *SLE Transfer Service Specifications* (references [12] through [14]).

In presenting the SLE Cross Support concept, the following assumptions are made:

a) the context is that of a single space mission;

b) within this space mission a single spacecraft is considered;

c) this spacecraft's telemetry and command systems are compliant with CCSDS Telemetry (TM) Recommendations (references [2], [3], and [7]) and either the CCSDS Telecommand (TC) Recommendations (references [5] through [7]) or the Advanced Orbiting Systems (AOS) Recommendation (reference [4]);

d) all end-users (i.e., sources or sinks of space data on the ground) are affiliated to a single mission manager.

1.3 APPLICABILITY

The term 'Cross Support services' encompasses all data handling services that one organization can provide to support another organization's spacecraft operations. On the ground, Cross Support services are of three kinds:

– SLE services, extending CCSDS Space Link services as defined in CCSDS Space Link Extension Recommendations (references [8] through [14]).

- Ground Communications services, providing ground communications support, e.g., to relay operational data.
- Ground Domain services, covering all services that handle data related to spacecraft operations, but that do not use the Space Link data structures defined in CCSDS Space Link Recommendations. Examples of ground domain services are spacecraft tracking, exchanging spacecraft databases, and mission planning.

The Cross Support documentation is divided into three reports:

- Part 1 of the Cross Support Concept (this document) addresses SLE services. It is applicable to any organization that wishes to extend Space Link services across the ground using SLE services.
- Part 2 of the Cross Support Concept (to be developed in the future) will establish the Cross Support concept for the use of communication services underlying the ground element.
- Part 3 of the Cross Support Concept (to be developed in the future) will address Ground Domain services.

At the time of writing, there are no plans to produce Part 2. However, Part 3 is under consideration.

1.4 RATIONALE

The primary goal of CCSDS is to increase the level of interoperability between space organizations. CCSDS promotes interoperability by developing data handling techniques for spacecraft operations, space research, and space science applications.

There are substantial costs associated with implementing and operating Space Data Systems. A significant portion of these costs is associated with frequent design and development of customized systems. As the demands placed upon Space Data Systems increase, system cost-reduction measures assume greater importance. The development of standard methods of performing routine tasks will reduce the costs.

CCSDS performs end-to-end systems analyses and adopts or develops standard approaches for solving routine problems relating to the development of Space Data Systems. After these approaches are officially accepted by the CCSDS member agencies, they are issued as CCSDS Recommendations. The Recommendations are subsequently adopted by the International Organization for Standardization (ISO) for use by any space organization.

As space organizations adopt and implement CCSDS Recommendations / ISO Standards, they develop compatible cross support capabilities. It is then easier and more cost-effective for space organizations to:

- share the resources established to acquire data from remote sensors;

– transport this information to designated destinations;

– process their data for various purposes;

– store the data for future reuse.

SLE services further the goal of interoperability by establishing a standard for services to be used in the area where most Cross Support activity occurs—between the tracking stations or ground data handling systems of various organizations and the mission-specific components of a mission ground system. The SLE services are applicable to routine, contingency and emergency operations.

1.5 DOCUMENT STRUCTURE

This Report is organized as follows:

a) Section 1 outlines the purpose, scope, applicability, and rationale of this Report, lists the definitions and references used throughout the Report, and presents the Cross Support documentation structure.

b) Section 2 provides a high level overview of the SLE services and an explanation of why they are needed and what benefits they offer.

c) Section 3 identifies the environment in which Cross Support occurs and the concept of the SLE services as extending the availability of space data in distance, information, and time.

d) Section 4 presents the Data Transfer aspects of the SLE services, relating them to the Space Link services and introducing the concept of SLE Complexes.

e) Section 5 presents the SLE Service Management concept and the time span of SLE Agreements and Service Packages.

f) Annex A defines the acronyms used in the text of this Report.

g) Annex B contains a list of terminology to be used in describing Cross Support.

h) Annex C identifies and provides examples of Cross Support scenarios.

i) Annex D provides SLE Service Management scenarios.

j) Annex E provides security scenarios.

1.6 DEFINITIONS

The definitions provided below are those needed for understanding this Report as a whole:

– **Return Data**. For the purposes of this Report, return data are all data that are sent from the space element to the ground element (e.g., telemetry).

– **Forward Data**. For the purposes of this Report, forward data is all data that are sent from the ground element to the space element (e.g., telecommand).

Many other terms are defined throughout this Report. In addition, several terms are used that have been defined in the *Cross Support Reference Model* (reference [8]).

1.7 REFERENCES

The following documents are referenced in the text of this Report. At the time of publication, the editions indicated were valid. All documents are subject to revision, and users of this Report are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently valid CCSDS Recommendations.

[1] *Procedures Manual for the Consultative Committee for Space Data Systems*. CCSDS A00.0-Y-9. Yellow Book. Issue 9. Washington, D.C.: CCSDS, November 2003.

[2] *TM Space Data Link Protocol*. Recommendation for Space Data System Standards, CCSDS 132.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[3] *TM Synchronization and Channel Coding*. Recommendation for Space Data System Standards, CCSDS 131.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[4] *AOS Space Data Link Protocol*. Recommendation for Space Data System Standards, CCSDS 732.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[5] *TC Synchronization and Channel Coding*. Recommendation for Space Data System Standards, CCSDS 231.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[6] *TC Space Data Link Protocol*. Recommendation for Space Data System Standards, CCSDS 232.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[7] *Space Packet Protocol*. Recommendation for Space Data System Standards, CCSDS 133.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[8] *Cross Support Reference Model — Part 1: Space Link Extension Services*. Recommendation for Space Data Systems Standards, CCSDS 910.4-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, September 2005.

[9] *Space Link Extension—Service Management—Service Specification*. Draft Recommendation for Space Data System Standards, CCSDS 910.11-R-1. Red Book. Issue 1. Washington, D.C.: CCSDS, March 2006.

[10] *Space Link Extension—Return Operational Control Fields Service Specification*. Recommendation for Space Data System Standards, CCSDS 911.5-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, December 2004.

[11] *Space Link Extension—Return Channel Frames Service Specification*. Recommendation for Space Data System Standards, CCSDS 911.2-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, December 2004.

[12] *Space Link Extension—Return All Frames Service Specification*. Recommendation for Space Data System Standards, CCSDS 911.1-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, December 2004.

[13] *Space Link Extension—Forward CLTU Service Specification*. Recommendation for Space Data System Standards, CCSDS 912.1-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, December 2004.

[14] *Space Link Extension—Forward Space Packet Service Specification*. Recommendation for Space Data System Standards, CCSDS 912.3-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, December 2004.

[15] *Space Link Extension—Internet Protocol for Transfer Services*. Draft Recommendation for Space Data System Standards, CCSDS 913.1-R-1. Red Book. Issue 1. Washington, D.C.: CCSDS, October 2005.

[16] *Information Technology—Abstract Syntax Notation One (ASN.1): Specification of Basic Notation*. International Standard, ISO/IEC 8824-1:2002. 3rd ed. Geneva: ISO, 2002.

[17] *Information Processing Systems—Open Systems Interconnection—Basic Reference Model—Part 4: Management Framework*. International Standard, ISO/IEC 7498-4:1989. Geneva: ISO, 1989.

[18] *Information Technology—Open Systems Interconnection—The Directory: Public-Key and Attribute Certificate Frameworks*. International Standard, ISO/IEC 9594-8:2001. 4th ed. Geneva: ISO, 2001.

[19] *Orbit Data Messages*. Recommendation for Space Data System Standards, CCSDS 502.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2004.

The relationship of this document to the other SLE documents is shown in figure 1-1. The suite of Cross Support documents includes:

– *Space Link Extension — Cross Support Reference Model*: a Recommendation that introduces the *Cross Support Reference Model* (reference [8]);

– *Space Link Extension — Service Management Specifications*: a suite of Recommendations that define the management interface required for the provision of one or more SLE services (reference [9]);

– *SLE Transfer Service Specifications*: Recommendations that define the data transfer services (references [10] through [14]).

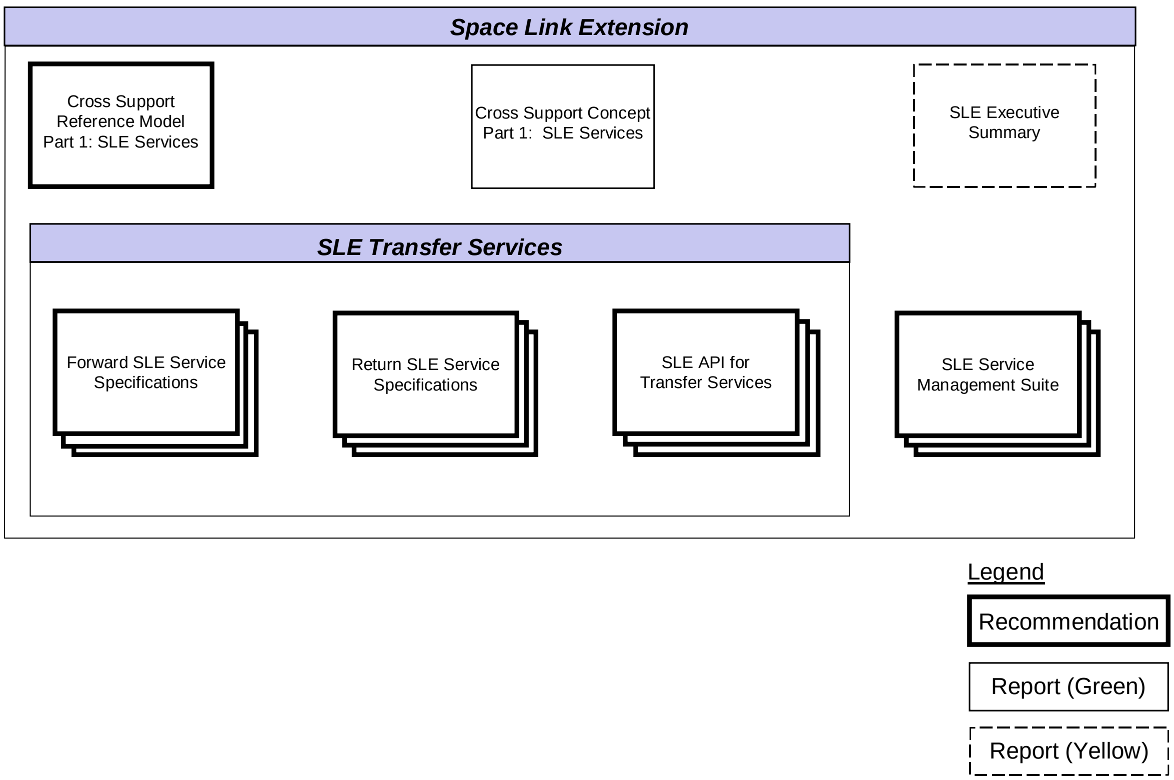


Figure 1-1: SLE Services Documentation Set

2 BACKGROUND TO SLE SERVICES

2.1 WHAT ARE SLE SERVICES?

The SLE Services extend the return Telemetry (TM), forward Telecommand (TC), and forward Advanced Orbiting Systems (AOS) services defined by CCSDS that are used by many spacecraft operators on the space link between ground stations and spacecraft (see figure 2-1).

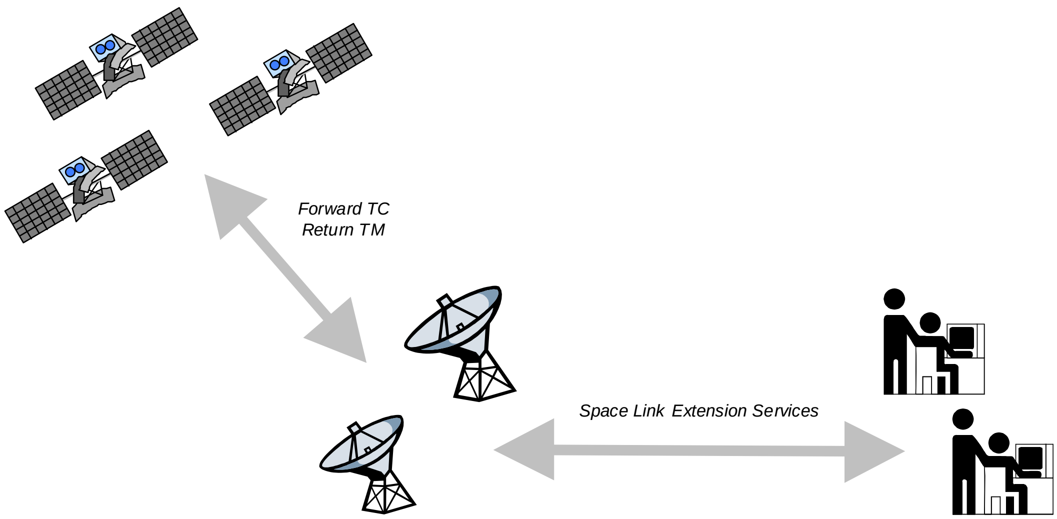


Figure 2-1: Ground Stations Provide SLE Services to Users

The SLE Services include two major elements:

– data transfer services that move space link data units between ground stations, control centers, and end-user facilities;

– management services that control the scheduling and provisioning of the data transfer services.

The SLE services operate in two phases:

– the definition phase, when most of the management activities take place;

– the utilization phase, when the data transfer takes place; this can be either in real-time or off-line with respect to the contact time with the spacecraft.

The information carried by the SLE services can be anything from spacecraft commands in the forward direction to science data in the return direction. In addition, the service will convey information such as TM data reception times and ground station configuration information.

2.2 WHY DO WE NEED SLE SERVICES?

The need for SLE Services arises from the desire of spacecraft operations organizations to standardize the interfaces for the transport and management of space data on the ground so that the technical, management and operational costs of providing Cross Support between the organizations can be greatly reduced.

The SLE Services that have been defined by CCSDS at the time of writing cover on-line ‘conventional’ TM and TC services. These are the services that are used by the majority of missions.

Other SLE services may be defined in the future, as the need for them arises.

2.3 WHAT ARE THE BENEFITS OF SLE SERVICES?

SLE Services enable the ground segment assets of space agencies, ground station operators, and space data users to interoperate without the need for ad hoc and complicated gateways specifically designed for each new mission.

By standardizing on the SLE Services, different organizations will be able to link discrete elements of their ground segments to suit a given mission’s needs without having to re-create the interfaces for each new mission.

Since the SLE protocols run over existing communications infrastructure, they help integrate Space Data Systems into the global communications network.

The advantages of SLE services are that:

– space organizations will be able to provide Cross Support to one another more efficiently;

– ground station owners will be able to provide standard services to operators of spacecraft that use CCSDS Space Link protocols;

– users of spacecraft data will be able to command their payloads and access their data through a familiar interface, using widely available underlying telecommunications technology such as the internet or ISDN lines;

– the standardization of ground station, control center and end user interfaces will permit re-use of systems for successive missions and eliminate the costs and risks associated with mission-specific implementations;

– a global market for standard Telemetry, Tracking and Command (TT&C) Commercial Off The Shelf (COTS) products will be stimulated, reducing the cost of these systems;

– SLE services are scalable so that only the actual services required by a service user or a service provider need to be implemented.

3 CROSS SUPPORT ENVIRONMENT

3.1 CONTEXT

A space mission is an undertaking to explore fields of interest by using one or more spacecraft. Each space mission involves significant functionality, both in space and on the ground. This functionality may be implemented by a single space organization, or it may be divided among multiple space organizations. Cross Support is when multiple space organizations provide space mission functionality.

The data services provided between the spacecraft and the ground are services built upon the CCSDS Space Link Recommendations (references [2] through [7]) for conventional telecommand systems, conventional telemetry systems, and advanced orbiting systems.

The CCSDS Space Link Recommendations define services that transfer data across the Space Link. However, in order to use the Space Link services to support mission requirements, Space Data Systems generally require additional features on the ground to extend the Space Link services beyond the ground termination of the Space Link. SLE services provide these additional features by extending the Space Link from systems on board the spacecraft, attached to onboard local area networks, to systems on the ground, attached to terrestrial wide area networks. This extension is illustrated in figure 3-1. SLE services also provide the ability to hold data at one or more intermediate points.

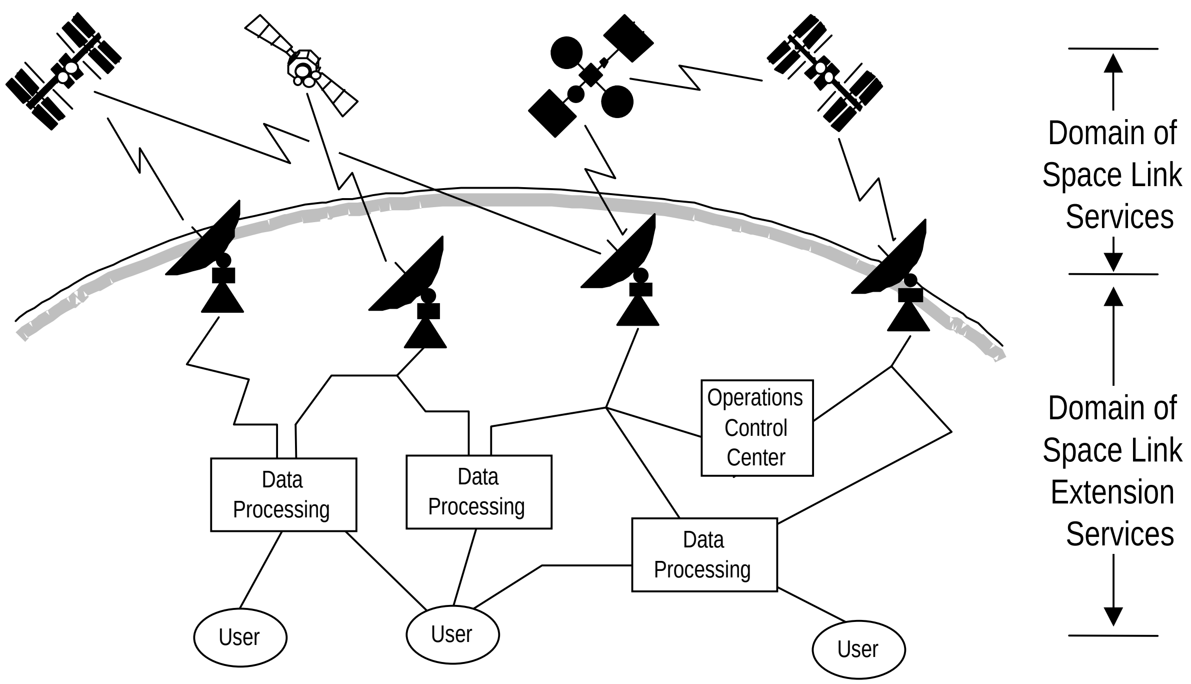


Figure 3-1: Domains of Space Link and Space Link Extension Services

3.2 SPACE DATA SYSTEMS

A Space Data System has two primary elements: a space element and a ground element. The space element and ground element are linked by Space Link communications services. The Space Link services are extended on the ground using the Space Link Extension services. The SLE services extend the Space Link services as follows:

– over distance;

– in time; and/or

– by adding information (i.e., annotation).

The Cross Support environment includes some services normally provided in ground stations, operation control centers, and data processing facilities, as illustrated in figure 3-1.

Figure 3-2 illustrates the two main components of a Space Data System: the space element and the ground element. The space element consists of the payloads, the astronauts and cosmonauts (including electronic systems that they use), and the spacecraft operations subsystems that control and monitor the spacecraft. The ground element consists of the ground data systems. The space element and ground element exchange return and forward data using a Space Link.

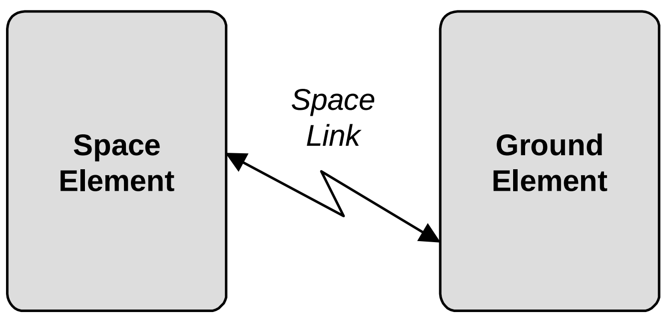


Figure 3-2: Space Data System

The space element acts as a source of return data and as a sink for forward data. In this Report, the space element represents a single spacecraft and comprises the platform and all payloads or instruments of this spacecraft. The space vehicles in a multi-spacecraft mission are treated as individual space elements. Platform and payloads or instruments are considered here only for their capability of generating or receiving data. Any processing that they perform on these data is outside the scope of the SLE services.

3.3 SPACE LINK EXTENSION SYSTEM ENVIRONMENT

The ground element of a Space Data System includes an SLE System and a Mission Data Operation System (MDOS). It may also contain other components, but these are outside the scope of this Report. Figure 3-3 illustrates these components.

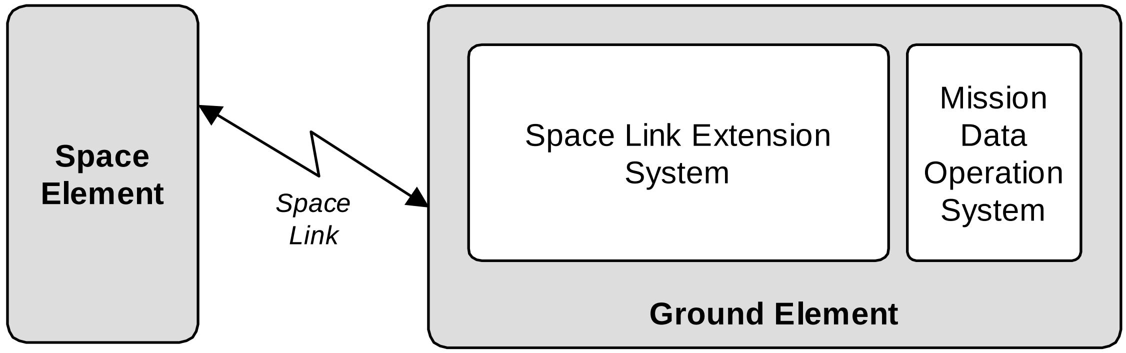


Figure 3-3: Ground Element of a Space Data System

The SLE System extends the transfer and delivery of forward and return data between a Space Link ground termination point and the MDOS. In the context of a mission, the SLE System is managed by the MDOS.

The complete set of SLE services is identified later in this Report. A particular mission may use all or a subset of the SLE services. In providing SLE services, the SLE System performs

- RF modulation/demodulation at the ground termination of the Space Link;
- ground termination of the Space Link protocols (that is, the protocols for the CCSDS Space Link services) used by the mission;
- adding of value-added information (called annotation) to Space Link service data;
- terrestrial networking among the ground elements that host the ground applications;
- interface to ground-side Space Link protocol processing and ground-side RF modulation/demodulation functions.

The MDOS is dedicated to a particular space mission. It acts as a source of forward data and as a sink of return data. In practice, the MDOS may not be the ultimate source of forward data or sink of return data on the ground element, but from the perspective of this Report it acts as if it were these sources and sinks.

3.4 SPACE LINK EXTENSION SERVICE CONCEPT

3.4.1 OVERVIEW

The Space Link protocols are unique to the space mission environment and provide the customized services and data communications protocols necessary to perform space/ground and space/space communications.

A key feature of the Space Link protocols is the capability to divide each Space Link physical channel into several separate logical data channels, known as Space Data Channels. Each Space Data Channel can carry data units, such as frames or packets. Different combinations of data units can be nested in the Space Data Channels but not all combinations are valid - the Space Link Recommendations define which combinations can be used. Different missions may select different combinations.

The SLE services provide the capability to transfer the data units between the Space Data Channels and the users on the ground, by providing users with access to the ground termination of the Space Link. This 'extension' of Space Link services has three aspects:

– **Extension over Distance**. Space Link protocol processing may be performed in multiple locations that are geographically separated from the place where the RF signal is received.

– **Extension in Information**. Information, indicating the conditions at the time of receipt, is added to the Space Link data. This is referred to as 'annotation'.

– **Extension in Time**. The SLE System also extends Space Link services in time, by allowing Space Link data to be transferred through the ground element and/or the Space Link at different times.

The following subsections elaborate these themes.

3.4.2 EXTENSION OVER DISTANCE

The Cross Support environment allows SLE services to be implemented by multiple service provider facilities, each performing a portion of the necessary processing. Each SLE service provider facility interfaces either to the MDOS, the Space Link or to another service provider facility. In the latter case, the facilities may be separated geographically.

In order to reduce processing and communications overhead, Space Link protocols frequently are not layered as rigorously as ground protocols. For example, a single data field may be reused for several purposes, or the value of a data field located in one layer may be needed for processing at higher layers. Such information might be lost if processing is performed at different locations. For example, the CCSDS Application Identifier (APID), located in the CCSDS Packet, is unique within a spacecraft but is not necessarily unique across missions. The Spacecraft Identification (SCID) field, located in the frame or Virtual Channel Data Unit (VCDU), serves to qualify the APID. This qualifying information would be lost if VCDUs were processed at a different location from the location of packet processing, unless provisions were made to send the SCID information. Therefore, each facility must send sufficient information to the next facility to allow the processing to be completed.

In addition to information associated with Space Link data, information is also needed that is not associated with an individual data unit. An example is 'loss of frame synchronization'. Such information about each SLE service must be defined as part of its service specification.

Figure 3-4 shows a situation that might arise when all SLE processing is not done at the ground station where the Space Link is terminated. In this case, the ground station provides the stream of frames extracted from the Space Link signal (shown as 'all frames' in the figure), along with associated annotation, to a geographically separate data processing facility that produces Space Packets and sends them directly to the user facility.

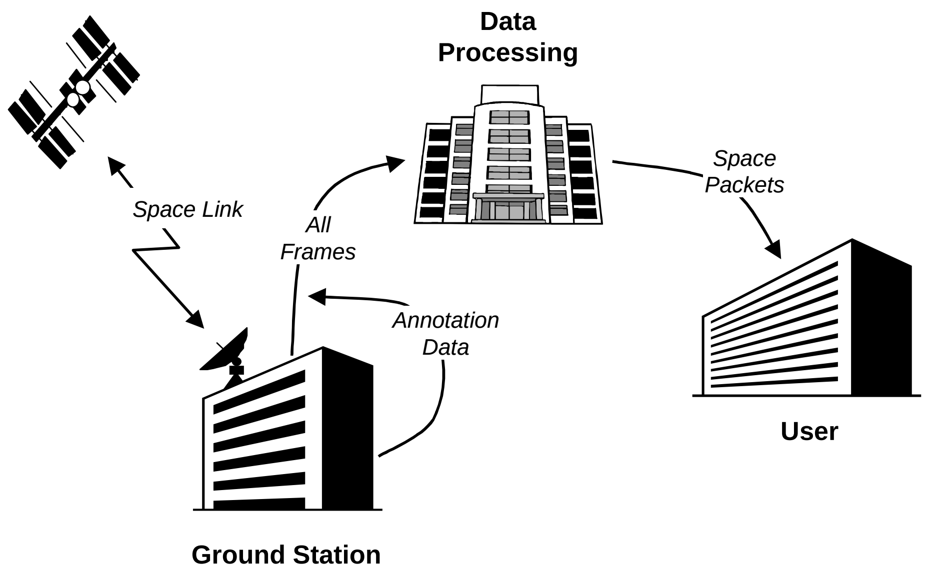


Figure 3-4: Example of Space Link Extension (Distance)

3.4.3 EXTENSION IN INFORMATION

Some information is not known prior to receipt of the Space Link data and must be added by the service facility receiving the Space Link data unit. For example, information that might be added to return link data by an SLE service may include:

– **time** when the Space Link data was received at the RF service facility;

– **identification** of the ground location where the data was received;

– **quality** of the data unit, indicating whether the data unit is correct and complete according to the criteria available;

– **sequence quality**, indicating whether the data unit is a direct successor to the previous instance of the data unit (i.e., an indication of missing data).

CCSDS Space Link protocols use out-of-band signaling for delivery of management information in order to save communications bandwidth and to minimize processing. Examples of additional information that may be added by the SLE service to supplement Space Link data are:

– **data type**, indicating whether the data were stored temporarily on board the spacecraft;

– **operations mode**, indicating the source of the data (e.g., spacecraft, simulator, test, etc.) and delivery option (online, offline/rate buffered, etc.);

– **signaling of data format parameters**, such as frame size or presence or absence of an insert.

Extension in information is illustrated in figure 3-5. This figure shows a simple case in which the Space Link is terminated at a ground station and annotation data are attached to the data stream. In the example, all SLE processing has been done at the ground station and fully processed space packets and annotation data are delivered directly to the user.

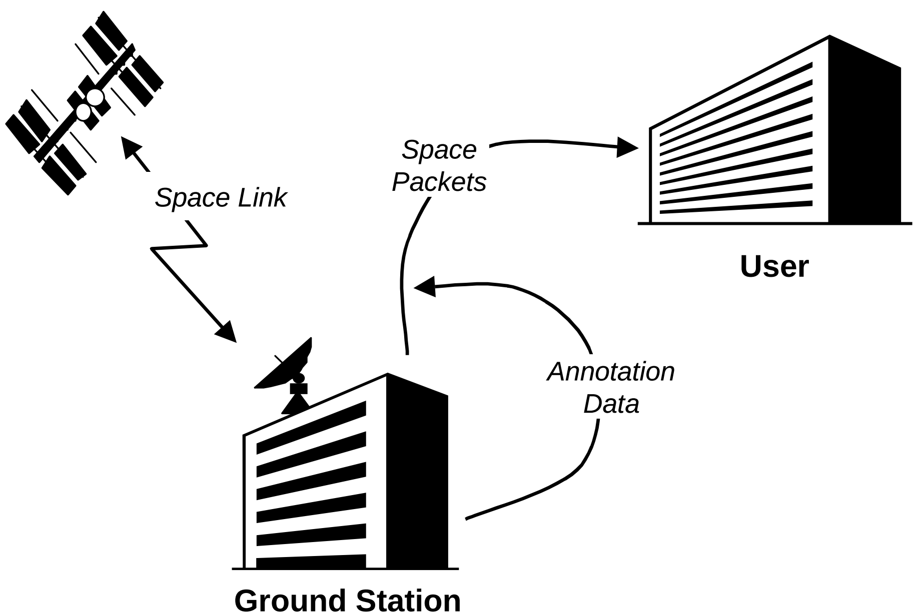


Figure 3-5: Example of Space Link Extension (Information)

3.4.4 EXTENSION IN TIME

The SLE System also extends Space Link services in time; making Space Link data available at a time after the data units are first received on the ground by adding information to ensure that the data are useful at this later time and by temporarily storing the data until they can be transferred to the user. The difference in data delivery time can vary widely. Delivery may be delayed very briefly, such as by a reduced communication rate on the ground, or it may be delayed greatly, such as by temporary storage and later forwarding.

3.4.5 EXTENSION IN DISTANCE, INFORMATION AND TIME

Figure 3-6 shows all three aspects of Space Link service extension. The signal is terminated at the ground station, where annotation data are attached to the data stream. This ground station provides a stream of frames extracted from the Space Link signal (shown as 'all frames' in the figure) to a remote data processing facility. This facility performs the next layer of SLE processing and provides a stream of Virtual Channel (VC) Frames to yet another data processing facility. The path is shown as a broken line to indicate great geographic distance, perhaps to a different continent. At the second facility the data might be stored temporarily. At a later date, the SLE processing is completed and space packets are produced and sent directly to the user facility.

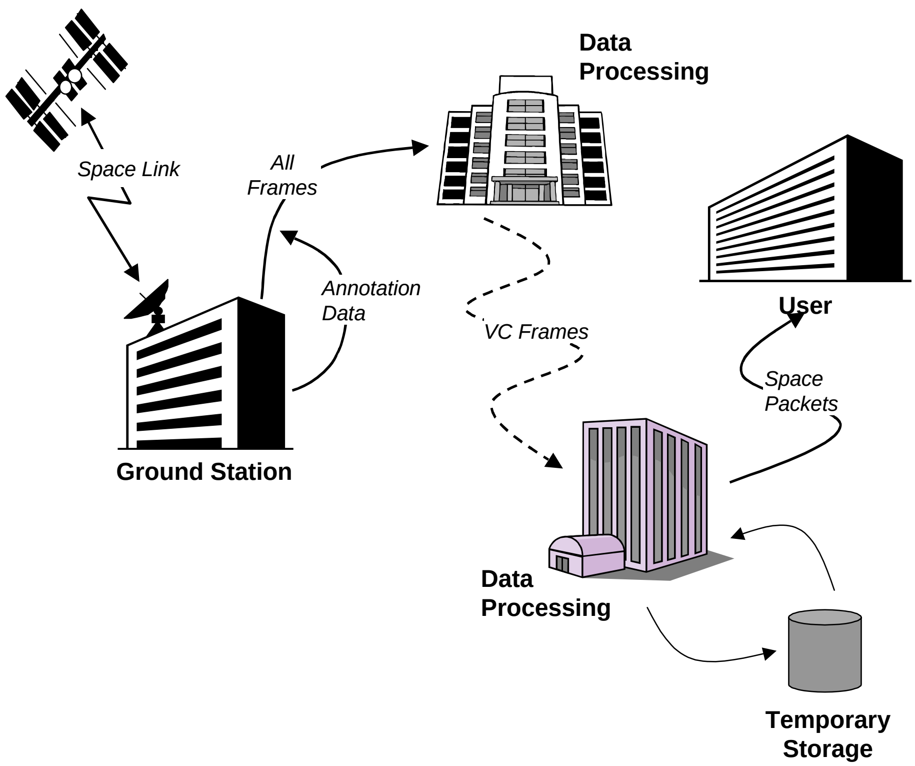


Figure 3-6: Example of Space Link Extension (Distance, Information and Time)

4 SLE DATA TRANSFER

4.1 OVERVIEW OF SLE DATA TRANSFER SERVICES

SLE data transfer services can be considered from both a functional and a physical viewpoint.

From the functional point of view, the processing required to provide SLE services is accomplished by a set of fundamental building blocks called Functional Groups. Functional Groups are derived from the layered functionality of channels, frames and packets specified in the Space Link protocols (references [2] through [7]). As it is a fundamental building block, a Functional Group cannot be decomposed further and a single Functional Group represents the set of SLE services that can be implemented in a real system at that layer. The *Cross Support Reference Model* (reference [8]) provides more detail concerning the functionality of Functional Groups.

From the physical point of view, these Functional Groups can be implemented in various facilities, such as those described in 3.4. The physical facilities that implement a set of Functional Groups are called SLE Complexes.

SLE data transfer services are separated into three categories:

– return SLE services;

– forward telecommand SLE services;

– forward AOS SLE services.

The following subsections describe each of these sets of SLE services and the SLE Functional Groups that are needed to provide the services.

The final subsections describe the SLE Complexes that implement the services, SLE data transfer service operations, and the SLE Application Program Interface.

4.2 RETURN SLE SERVICES

4.2.1 LIST OF RETURN SLE SERVICES AND THEIR INTERFACES

The return SLE services associated with TM include:

– **Return All Frames (RAF).** Provides the TM frames from a single space link symbol stream to spacecraft operators and other users who might need all the frames.

– **Return Channel Frames (RCF).** Provides Master Channel (MC) or specific VCs, as specified by each RCF service user.

– **Return Frame Secondary Header (RFSH).** Provides MC or VC Frame Secondary Headers (FSHs), as specified by each RFSH service user.

– **Return Operational Control Field (ROCF).** Provides MC or VC Operational Control Fields (OCFs), as specified by each ROCF service user.

– **Return Space Packet (RSP).** Enables single users to receive packets with selected Application Process Identifiers (APIDs) from one spacecraft VC.

– **Return Insert (R-Insert) and Return Bitstream (R-Bitstream).** Are AOS services (not yet defined by CCSDS).

Figure 4-1 shows the data transfer interfaces for these services. (Depending on the services required, implementations will expose only a subset of these interfaces as SLE services.)

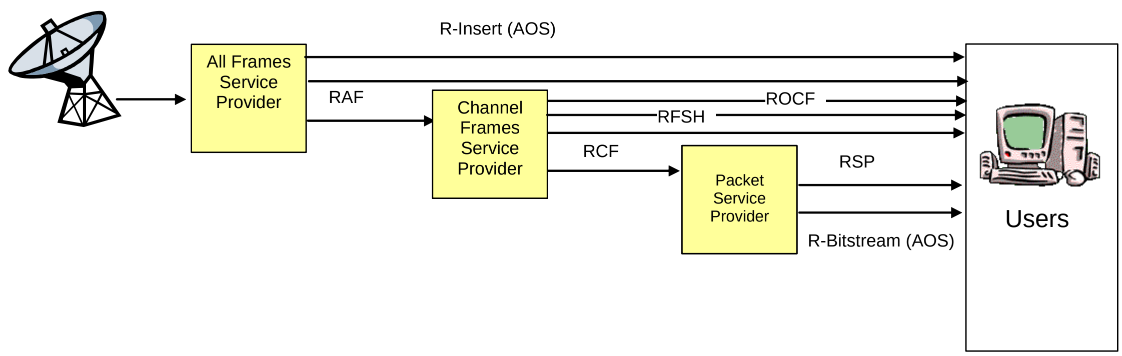


Figure 4-1: Return SLE Services Are Provided in Three Stages

Figures 4-2 and 4-3 show the Space Link data structures of the return frame and the return packet, which are transported by the return SLE services. The RAF service transfers all the frames shown in figure 4-2, whilst the RCF service transfers a selection of the frames, chosen on the basis of the Master Channel or Virtual Channel ID contained in the Transfer Frame Primary Header. The RFSH and ROCF services provide the FSH and OCF extracted from the frame. The RSP service transfers the packets shown in figure 4-3, selected on the basis of the Application Process ID.

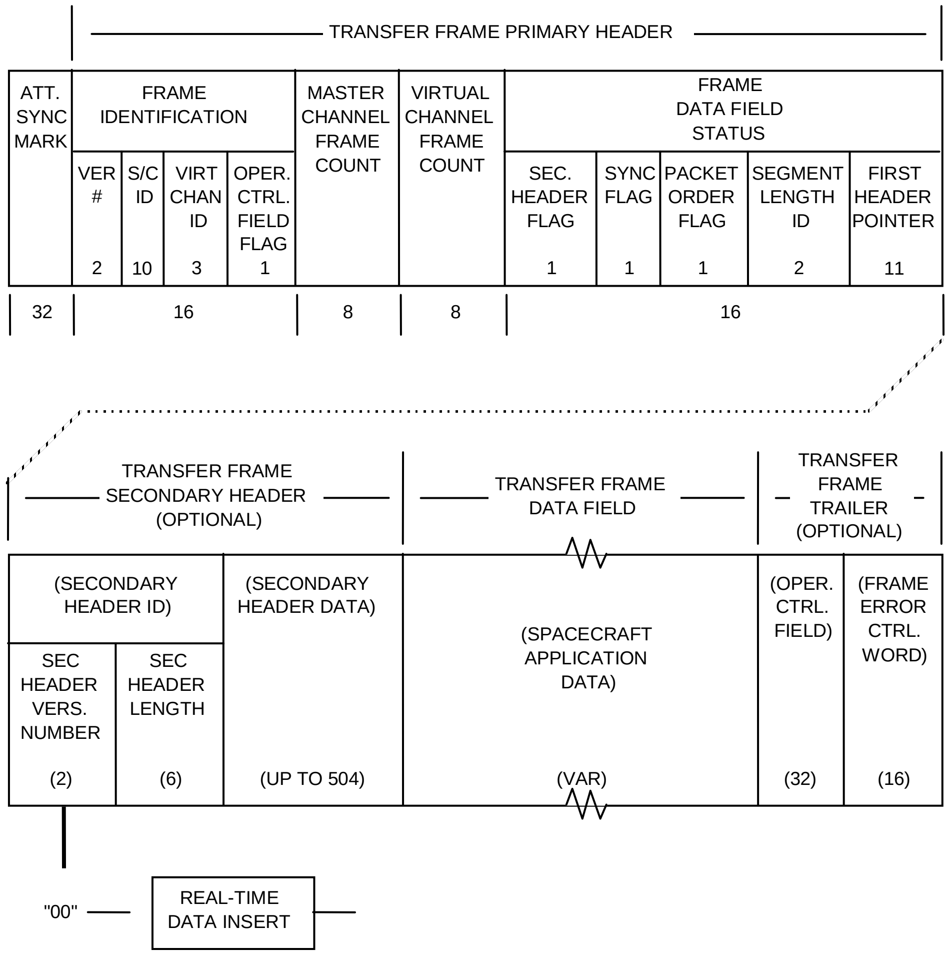


Figure 4-2: Return Frame Data Structure

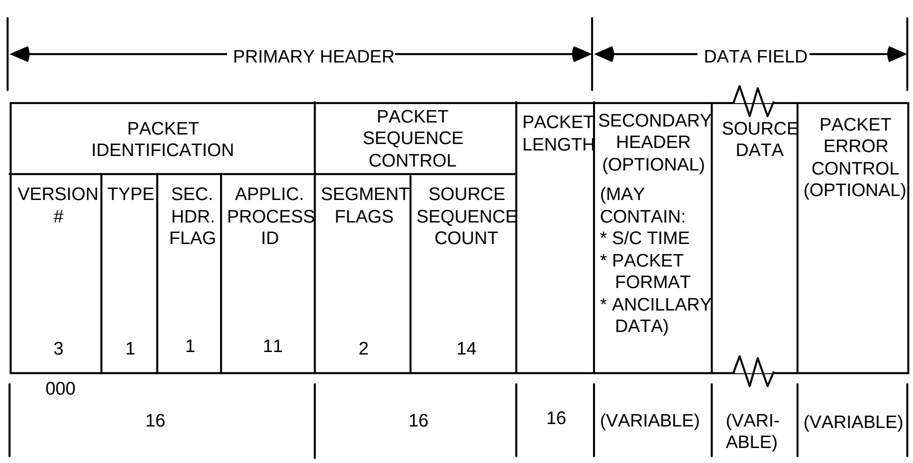


Figure 4-3: Return Packet Data Structure

4.2.2 MAPPING OF RETURN SLE SERVICES TO FUNCTIONAL GROUPS

A Return SLE service instance is produced by a succession of up to three ‘SLE Functional Groups’ (SLE-FGs) each corresponding to a ‘Service Provider’ box as shown in figure 4-1. For each of the service instances produced by a particular SLE-FG, the SLE-FG exposes an interface to be accessed by a registered user (or by the subsequent SLE-FG). For example, organizations wishing to act only as an ‘All Frames Service Provider’ can do so by implementing just a ‘Return Space Link Processing’ SLE-FG. However, to provide a return packet service requires all three SLE-FGs, which may all be implemented by one organization or may be distributed between different organizations.

To be a …	… the provider must have implemented …
All Frames Service Provider	Return Space Link Processing SLE-FG
Channel Frames Service Provider	Return Frame Processing SLE-FG Return Space Link Processing SLE-FG
Packet Service Provider	Return Frame Data Extraction SLE-FG Return Frame Processing SLE-FG Return Space Link Processing SLE-FG

Figure 4-4 shows the Functional Groups involved in performing return SLE services and identifies the services that are provided by each. All of the services shown are available for Cross Support, including those between the Functional Groups.

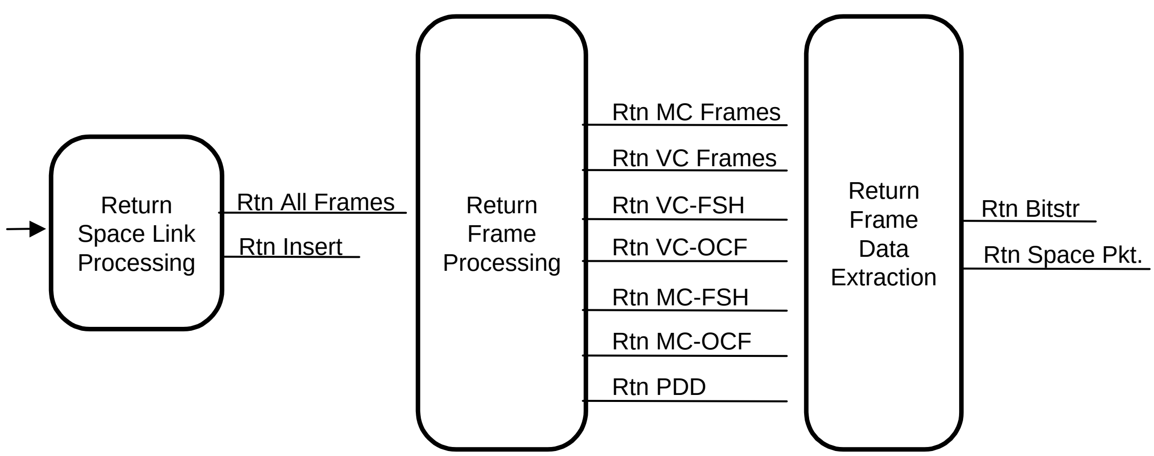


Figure 4-4: Return SLE Functional Groups and Services

Some SLE services are related in ways that make it impossible to distribute them across different Functional Groups. For example, processing for the Return Insert service must be merged in the same Functional Group with processing for the Return All Frames service since the insert data are included in each and every frame.

4.3 FORWARD TELECOMMAND SLE SERVICES

4.3.1 LIST OF FORWARD TELECOMMAND SLE SERVICES AND THEIR INTERFACES

The forward SLE services associated with conventional TC include:

– **Forward Space Packet (FSP).** Enables single users to provide packets for uplink to a spacecraft without needing to co-ordinate with other users of the spacecraft.

– **Forward Telecommand Virtual Channel Access (FTCVCA)**. Enables users to provide complete VCs for uplink.

– **Forward Telecommand Frame (FTCF).** Enables users to provide TC frames to be transformed to Communication Link Transmission Units (CLTUs) ready for uplink.

– **Forward Communications Link Transmission Unit (CLTU).** Enables users to provide CLTUs for uplink to spacecraft.

Figure 4-5 shows the data transfer interfaces for these services. (Depending on the services required, implementations will expose only a subset of these interfaces as SLE services.)

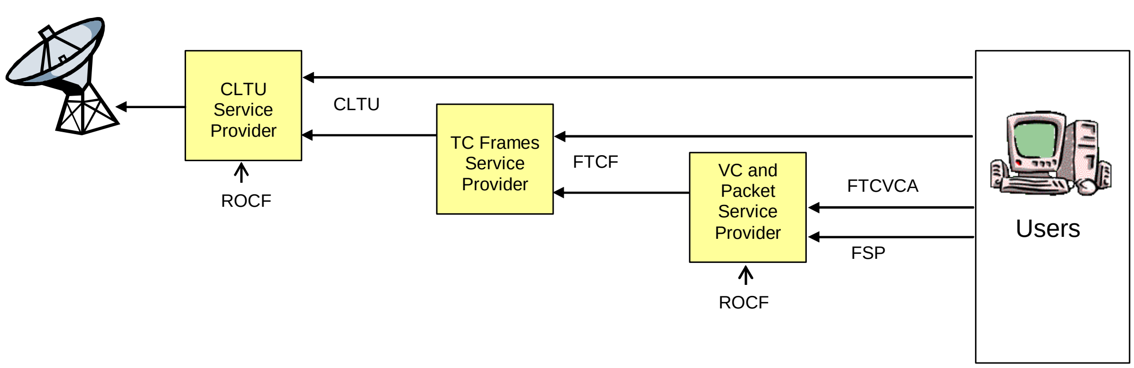


Figure 4-5: Three-Stage Approach to Conventional Forward SLE Services

Figure 4-5 shows the ROCF service as an input to the CLTU Service Provider and the VC and Packet Service Provider. The ROCF service provides the Communications Link Control Word (CLCW), which is required by the CLTU Service Provider to determine the availability of the physical space link channel and by the VC and Packet Service Provider to determine if TC frames need to be retransmitted when the Communications Operation Procedure 1 (COP-1) is in effect.

Figures 4-6 through 4-8 show the Space Link data structures of the telecommand packet, telecommand frame, and CLTU. These are the data structures carried by the FSP, FTCF and CLTU services, respectively.

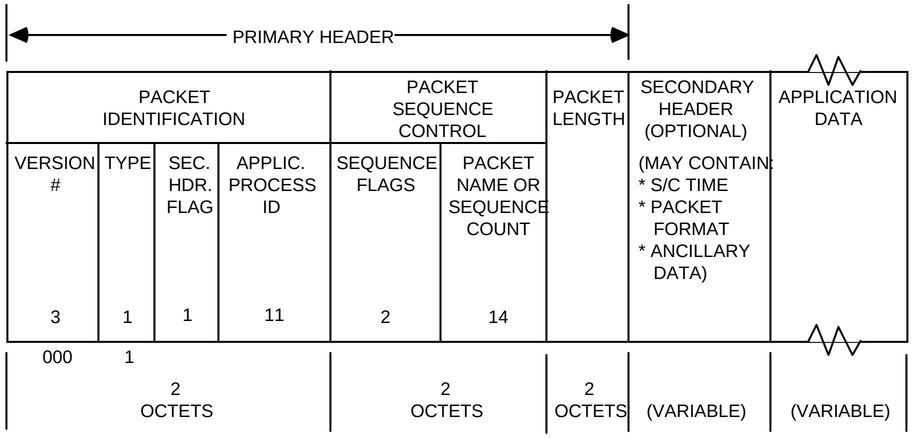


Figure 4-6: Telecommand Packet Format

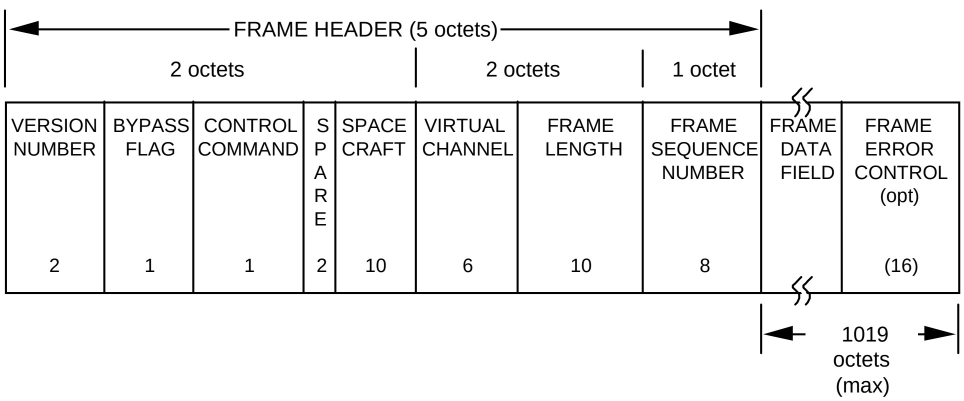


Figure 4-7: Forward Telecommand Frame Format

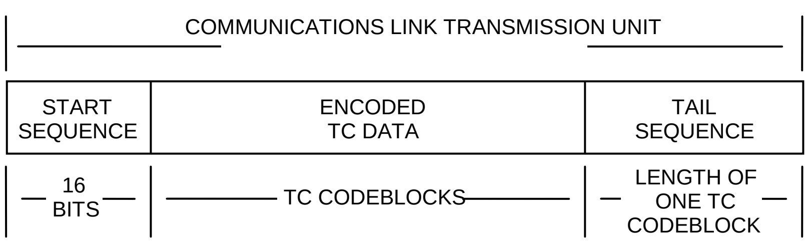


Figure 4-8: Components of the CLTU

4.3.2 MAPPING OF FORWARD TELECOMMAND SLE SERVICES TO FUNCTIONAL GROUPS

A Forward SLE service instance is produced by a succession of up to three SLE-FGs each corresponding to a 'Service Provider' box shown in figure 4-5. For each of the service instances produced by a particular SLE-FG, the SLE-FG exposes an interface to be accessed by a registered user (or by the subsequent SLE-FG). For example, organizations wishing to act only as a 'CLTU Service Provider' can do so by implementing just a 'Forward Telecommand Space Link Processing' SLE-FG. However, to provide a forward packet service requires all three SLE-FGs, which may all be implemented by one organization or may be distributed between different organizations.

To be a …	… the provider must have implemented …
CLTU Service Provider	Forward Telecommand Space Link Processing SLE-FG
TC Frames Service Provider	Forward CLTU Generation SLE-FG Forward Telecommand Space Link Processing SLE-FG
VC and Packet Service Provider	Forward TC VC Data Insertion SLE-FG Forward CLTU Generation SLE-FG Forward Telecommand Space Link Processing SLE-FG

Figure 4-9 shows the Functional Groups involved in Forward Telecommand SLE services and identifies the services that are provided by each. All of the services shown are available for Cross Support, including those between the Functional Groups.

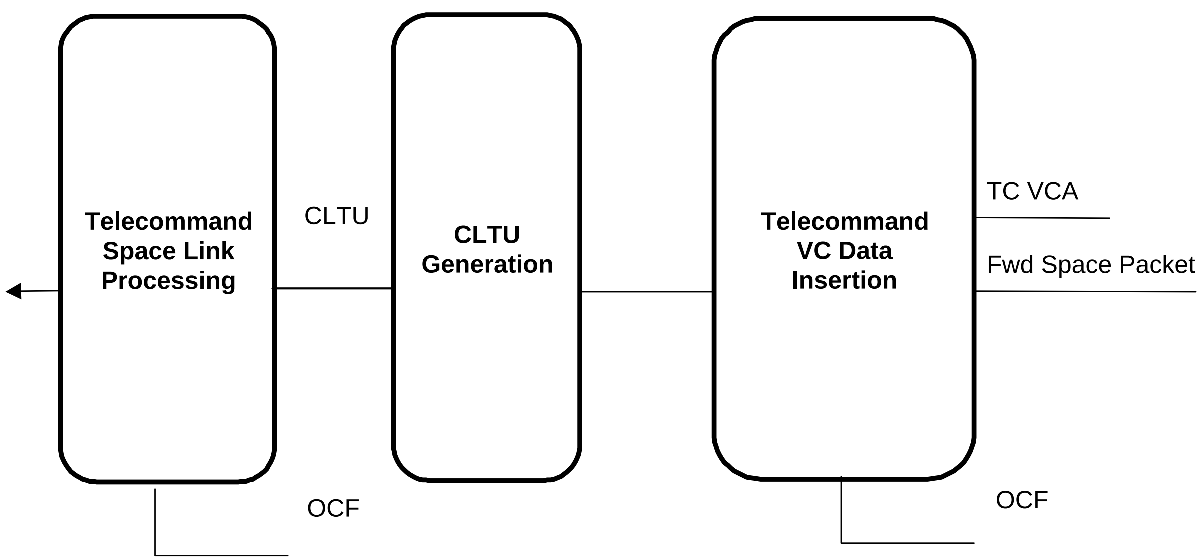


Figure 4-9: Forward Telecommand SLE Functional Groups and Services

4.4 FORWARD AOS SLE SERVICES

4.4.1 LIST OF FORWARD AOS SLE SERVICES AND THEIR INTERFACES

The forward SLE services associated with the Advanced Orbiting Systems (AOS) protocols intended for use in applications such as space station operations are not yet defined by CCSDS but will include:

– Forward AOS Space Packet (F-AOSSP);

– Forward AOS Virtual Channel Access (F-AOSVCA);

– Forward Bitstream (F-Bitstream);

– Forward Proto Virtual Channel Data Unit (F-pVCDU);

– Forward Coded Virtual Channel Data Unit (F-cVCDU);

– Forward Virtual Channel Data Unit (F-VCDU);

– Forward Insert (F-Insert).

The services will be implemented in two SLE-FGs as shown in figure 4-10.

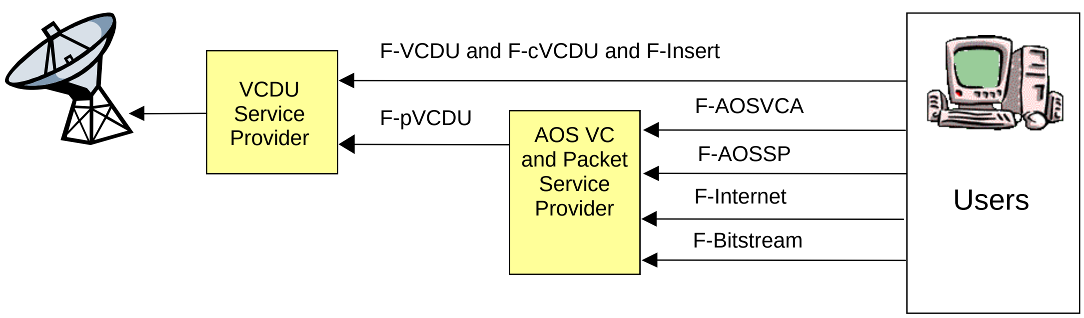


Figure 4-10: AOS Forward SLE Services Are Provided in Two Stages

4.4.2 MAPPING OF FORWARD AOS SLE SERVICES TO FUNCTIONAL GROUPS

The Forward AOS SLE services are implemented as a modular architecture consisting of two SLE-FGs corresponding to the 'Service Provider' boxes shown in figure 4-5.

To be a …	… the provider must have implemented …
VCDU Service Provider	Forward AOS Space Link Processing SLE-FG
AOS VC and Packet Service Provider	Forward AOS VC Data Insertion SLE-FG Forward AOS Space Link Processing SLE-FG

Figure 4-11 shows the Functional Groups involved in performing Forward Link AOS SLE services and identifies the services that are provided by each. All of the services shown are available for Cross Support, including those between the Functional Groups.

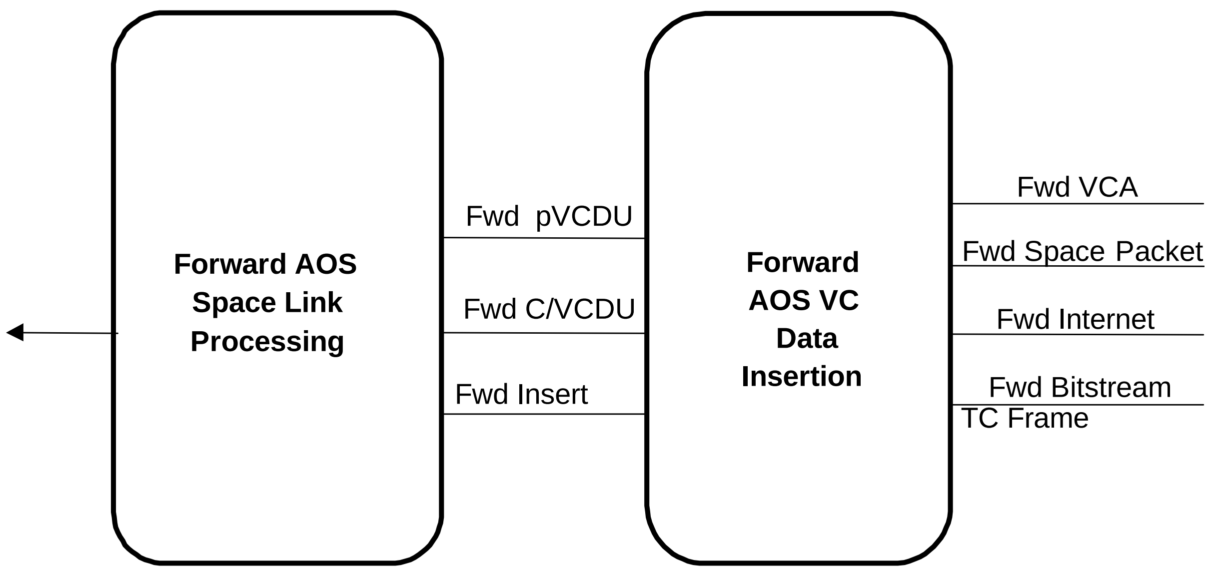


Figure 4-11: Forward AOS SLE Functional Groups and Services

4.5 SLE COMPLEXES

The SLE Complex is a logical grouping of Functional Groups that collectively perform a subset of the entire SLE service functionality. One or more Functional Groups can be implemented in one or more SLE Complexes by different organizations. Individual functions of a Functional Group cannot be distributed across multiple SLE Complexes - all functions of a Functional Group must be implemented within a single SLE Complex.

SLE Systems may be made up of several SLE Complexes that interoperate to provide a space mission with SLE services. These physical systems may be operated independently by different agencies. Each Complex must implement all the functions required to provide the particular SLE services that it offers. However, not all functions need necessarily to be exposed outside the Complex - only those that provide the SLE Service required by the user.

From a Cross Support view, the SLE System is composed of one or more SLE Complexes as illustrated in figure 4-12. The interfaces between SLE Complexes must conform to the interfaces defined for SLE Services in the SLE Service Management and SLE Transfer Service suite of Recommendations.

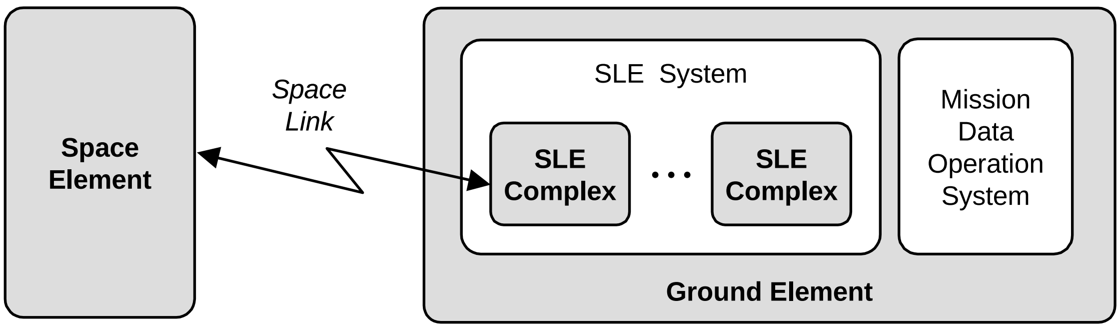


Figure 4-12: SLE Complexes

Figure 4-13 shows an example of how three SLE Complexes could be combined to provide the RSP and FSP services to users in an MDOS.

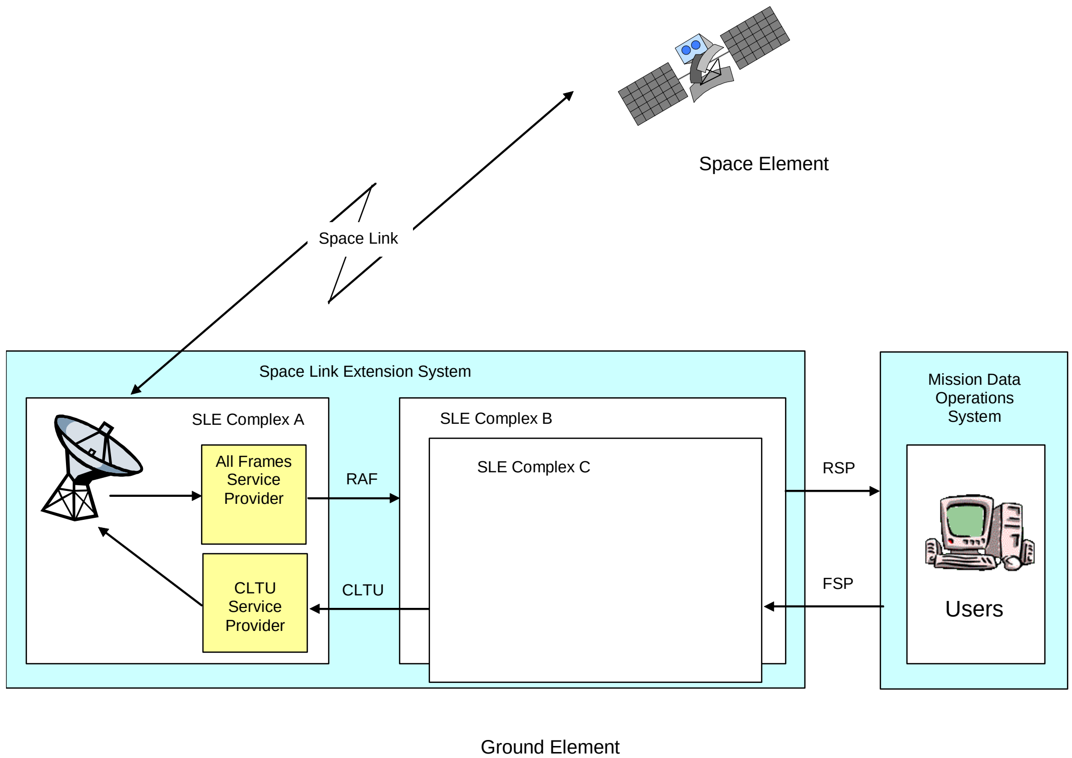


Figure 4-13: Example of Connected SLE Complexes

4.6 SLE DATA TRANSFER SERVICE OPERATIONS

4.6.1 DESCRIPTION OF OPERATIONS

The following operations are common to all the SLE Data Transfer services:

– **BIND** and **UNBIND.** These operations make and (cleanly) break the connection between user and provider.

– **PEER-ABORT.** This operation breaks the connection between user and provider in the event of an SLE protocol problem.

– **START** and **STOP.** These operations change the state of the user and provider to and from the active state in which SLE data units may be transferred from one to the other.

– **TRANSFER-DATA.** This operation transfers data units from the user to provider in the forward case, and from the provider to user in the return case.

– **SYNC-NOTIFY** and/or **ASYNC-NOTIFY.** These operations either synchronously or asynchronously to the TRANSFER-DATA operation, enable notifications about the state of the service to be sent to the user from the provider.

– **SCHEDULE-STATUS-REPORT** and **STATUS-REPORT**. These operations allow a user to schedule status reporting from the provider and allow a provider to provide the requested status reports.

– **GET-PARAMETER.** This operation allows a user to get the value of a parameter from a provider, which is relevant to the current service instance.

Some services include these operations too:

– **THROW-EVENT.** This operation allows a forward direction service user to send an event that requires management action.

– **INVOKE-DIRECTIVE.** This operation allows a ‘privileged’ FSP or TCVCA user to invoke TC directives related to, for example, the operation of a Communications Operation Procedure (COP).

4.6.2 STATE TRANSITIONS

Figure 4-14 shows an example of the state transitions of a service instance on the SLE service provider side, using the forward CLTU service as an example.

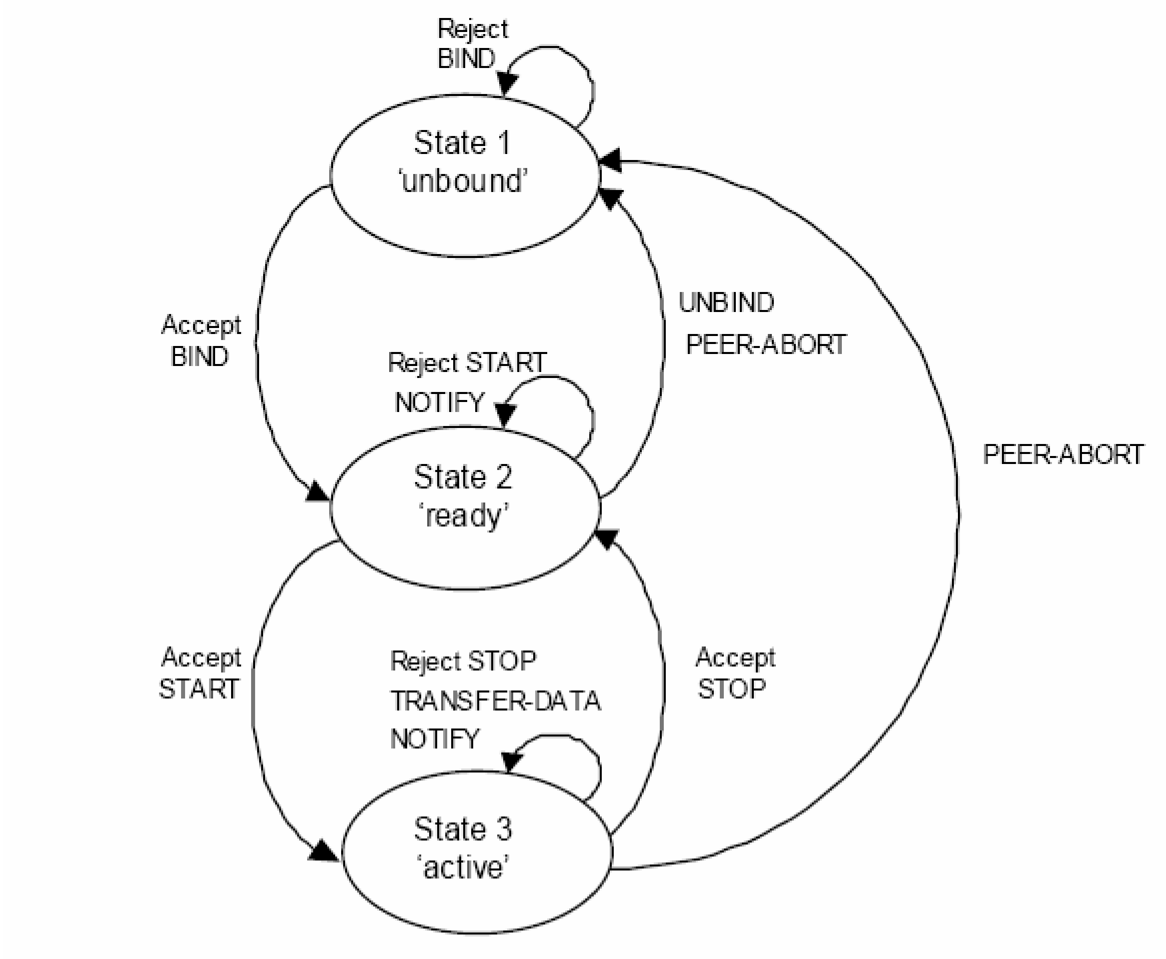


Figure 4-14: Simplified State Transition Diagram for CLTU Service

4.6.3 OPERATION EXAMPLE

Each operation in an SLE service specification has a table of parameters such as the one shown in figure 4-15. Each parameter in the table is described in the specification, which also lists the values that the parameter may have.

In the table, 'M' refers to a mandatory parameter and 'C' to a conditional parameter. All parameters are mandatory except the `diagnostic` parameter, which is only returned by the provider if the `result` parameter has a value of 'negative'.

Parameter	Invocation	Return
invoker-credentials	M	
performer-credentials		M
invoke-ID	M	M
start-time	M	
stop-time	M	
global-VCID	M	
result		M
diagnostic		C

Figure 4-15: Example of SLE Service Operation Table (RCF-START)

Each operation has a corresponding ASN.1 definition in an annex to the service specification. Figure 4-16 provides an example of ASN.1 notation for RCF-START.

```
RcfStartInvocation       ::=  SEQUENCE
{    invokerCredentials       Credentials
,    invokeId                 InvokeId
,    startTime                ConditionalTime
,    stopTime                 ConditionalTime
,    gvcId                    GvcId
}
```

Figure 4-16: Example of ASN.1 Notation for RCF-START

4.7 THE SLE APPLICATION PROGRAM INTERFACE (API)

The SLE API for transfer services provides an interface to SLE application programs for exchange of SLE operation invocations and returns between an SLE service user and an SLE service provider. The API consists of two distinct layers, the API Proxy and the API Service Element, as shown in figure 4-17. The SLE Application is not part of the API, but it must provide some interfaces for use by the API, which will be defined in an SLE API Specification (planned for future publication).

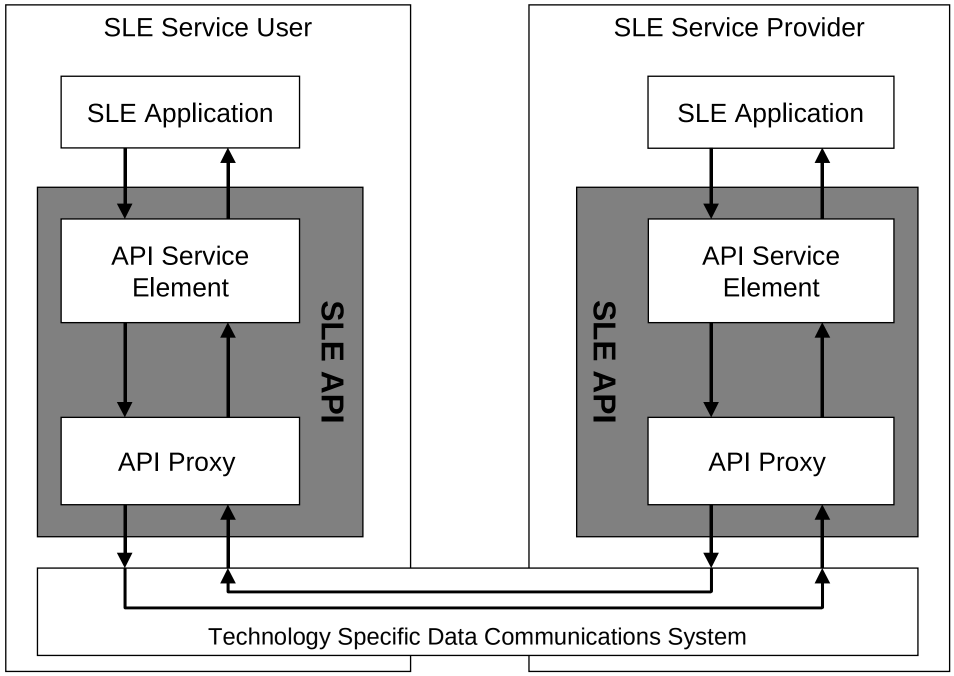


Figure 4-17: Layers of the SLE API

The purpose of the API Proxy is to provide a common, technology independent interface to the data communications system used for transmission of SLE protocol data units. It makes higher layers of the API and SLE applications independent of the specific communications technology and shields applications from technology changes.

The API Service Element implements those aspects of SLE transfer service provisioning that can be clearly separated from service production and service use, offloading applications from lower level aspects of the SLE transfer service protocol. The API Service Element enforces conformance to the state tables defined by the CCSDS Recommendations and checks parameters of SLE operations on completeness, consistency and range. These features support early detection of any potential problems in SLE applications and protect the application software from errors induced by a peer system.

4.8 FUTURE DATA TRANSPORT SERVICES

Currently the Data Transfer services are limited to the CCSDS SLE services described in the previous sections. However, CCSDS is considering the creation of Data Transfer services for non-CCSDS space link protocols that would utilize the same basic operations and the SLE Service Management services described in section 5.

Other Data Transport services to be defined within CCSDS include radio-metric, monitoring, and return unformatted data.

These services will be addressed in future Cross Support Services concept books.

5 SLE SERVICE MANAGEMENT

5.1 OVERVIEW OF SLE SERVICE MANAGEMENT CONCEPTS

SLE Service Management facilitates the collection and exchange of all information which is required to agree, schedule, prepare and access SLE services. To achieve this, SLE Service Management provides standardized information templates and interaction mechanisms.

Information templates capture all the parameters necessary for:

– detailing the service types and service instances required to support a mission phase;

– scheduling of the service instances;

– determining the SLE Complex resources to be allocated for SLE service production;

– creating and exposing interfaces for access to, and control of, service instances;

– configuring, monitoring and controlling by the provider of the SLE Complex resources during the execution of the service.

Interaction mechanisms ensure the safe and efficient exchange of the information items between ‘Utilization Management’ (UM) on the user side and ‘Complex Management’ (CM) on the provider side.

The management interactions between space mission users and service providers cover:

– negotiating a Service Agreement;

– defining Configuration Profiles for the service;

– generating Service Packages;

– and, where applicable, Trajectory Predictions used by the provider to acquire the spacecraft.

The standard operations and data structures associated with these processes are defined in the *SLE Service Management Service Specification* (reference [9]).

The entities involved in SLE Service Management are shown in figure 5-1 and are:

– Complex Management in one or more SLE Complexes;

– Utilization Management in a Mission Data Operations System.

Figure 5-1 shows how SLE Service Management fits in the context of the SLE System, the MDOS, and the SLE Data Transfer services that were described in section 4.

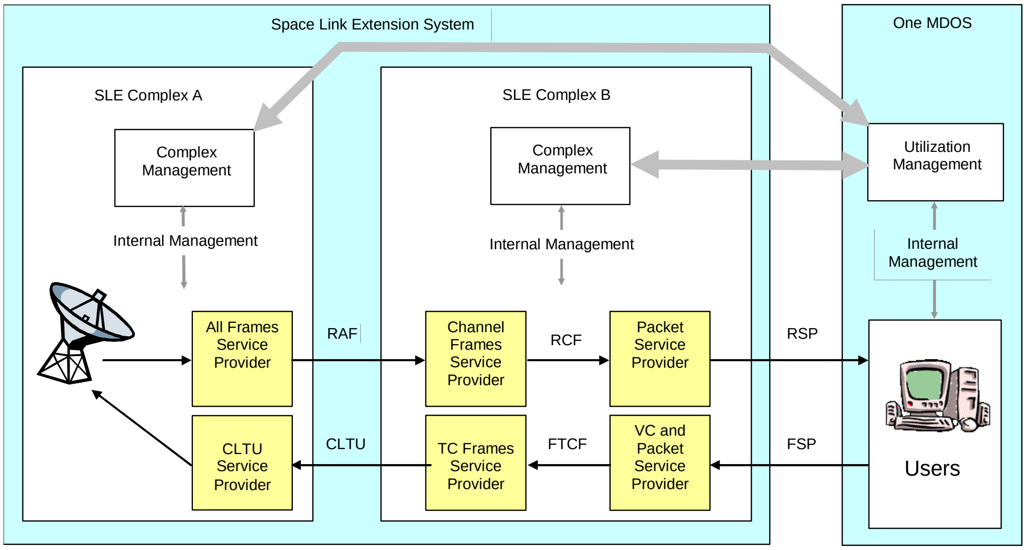


Figure 5-1: SLE Service Management in Context

SLE Complex Management presents the functions performed within the SLE Complex in a standard way to the user, as defined in the SLE Service Management Service Specification (reference [9]). CCSDS does not define the way in which Complex Management interfaces with the equipment that is used to provide the SLE services, which means that the users in the MDOS do not need to be concerned about the internal workings of the SLE Complex. In other words, SLE Complex Management provides an interface to the user that hides the complexity of the provider's SLE Complex. This is depicted in figure 5-2.

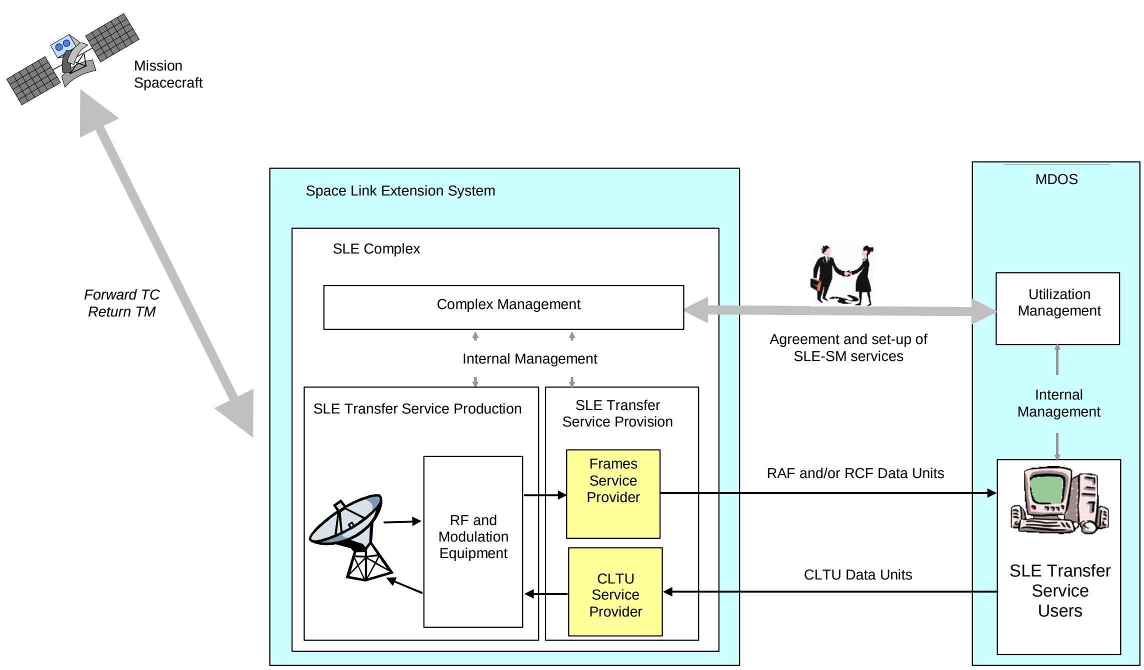


Figure 5-2: Overview of SLE Complex Management Internal Interfaces

5.2 SCOPE OF SLE SERVICE MANAGEMENT

The Open Systems Interconnection *Basic Reference Model* (reference [17]) identifies five ‘management functional areas’: configuration management, accounting management, fault management, security management, and performance management. SLE Service Management includes aspects of the first four functional areas.

– **Configuration management** is the most extensive aspect of SLE Service Management. The configuration management capabilities enable SLE Utilization Management to interact with SLE Complex Management to define collections of SLE transfer service instances grouped into SLE Service Packages. Each Service Package specifies the input and output data channels, processing parameters, and storage requirements to be provided by the SLE Complex to a space mission.

– The **accounting management** aspect of SLE Service Management enables SLE Complex Management to provide SLE Utilization Management with reports, for each Service Package, of the data that are received by, the data that are sent from, and the data that are processed by that particular SLE Complex.

– The **fault management** aspect of SLE Service Management provides SLE Utilization Management with notification of faulty conditions within the SLE Complex that affect services being provided. Actual fault detection, isolation, and correction within the SLE Complex are an internal activity that is not exposed to SLE Utilization Management.

– The **security management** aspect of SLE Service Management protects access to Cross Support services through administration of security credentials, control of access to Cross Support interfaces and authentication of requests for data.

– SLE Service Management does not include **performance management**, because monitoring and controlling an SLE Complex’s performance is an internal matter for the organization that operates the complex.

5.3 SLE SERVICE MANAGEMENT CONSIDERATIONS

The SLE Service Management concept is influenced by the following considerations:

– The user’s mission needs are the drivers of the management concept.

– The user is isolated from the details of the individual organization’s SLE System. The user interacts with SLE Utilization Management to provide service characteristics. The user does not need to know how the service is provided, but only how to interface with the service provider. SLE Utilization Management negotiates with the provider organizations to establish the parameters of the service interface.

– A common management representation of services is provided to users, regardless of the number and combination of services offered by any particular organization. It supports flexibility of an organization to organize internal management domains as it sees fit, while presenting a consistent service view to the user, and it facilitates the ability of organizations to hide system-specific details from users.

– Adequate reporting within the organization's SLE System is provided to the user. The purpose is to provide sufficient accountability for all services performed by the system. The management concept does not specify SLE Complex-internal management requirements or concepts for management data collection and reporting. It only deals with the information that needs to be provided to the mission.

– As far as possible, the management process is designed with automation in mind.

– The management concept has been formulated to minimize operational costs.

– An incremental approach to the implementation of SLE Service Management has been adopted in order to allow providers to evolve their systems towards SLE at a pace that suits their schedules, budgets and commitments to their existing users, some of whom may not use CCSDS protocols on the Space Link.

5.4 SLE SERVICE MANAGEMENT ENVIRONMENT

5.4.1 GENERAL CASE

Figure 5-3 shows the SLE architecture model defined in the *Cross Support Reference Model* (reference [8]).

The focus of *Service Management* (reference [9]) is the interface indicated by the bold dashed line near the top of the figure. SLE Service Management is defined from the perspective of SLE Utilization Management.

Service Management takes place when SLE Utilization Management invokes management operations and SLE Complex Management performs them. *Service Management* (reference [9]) defines the dialog between the two entities, but not the manner in which the operations are performed. This is internal to the SLE Complex.

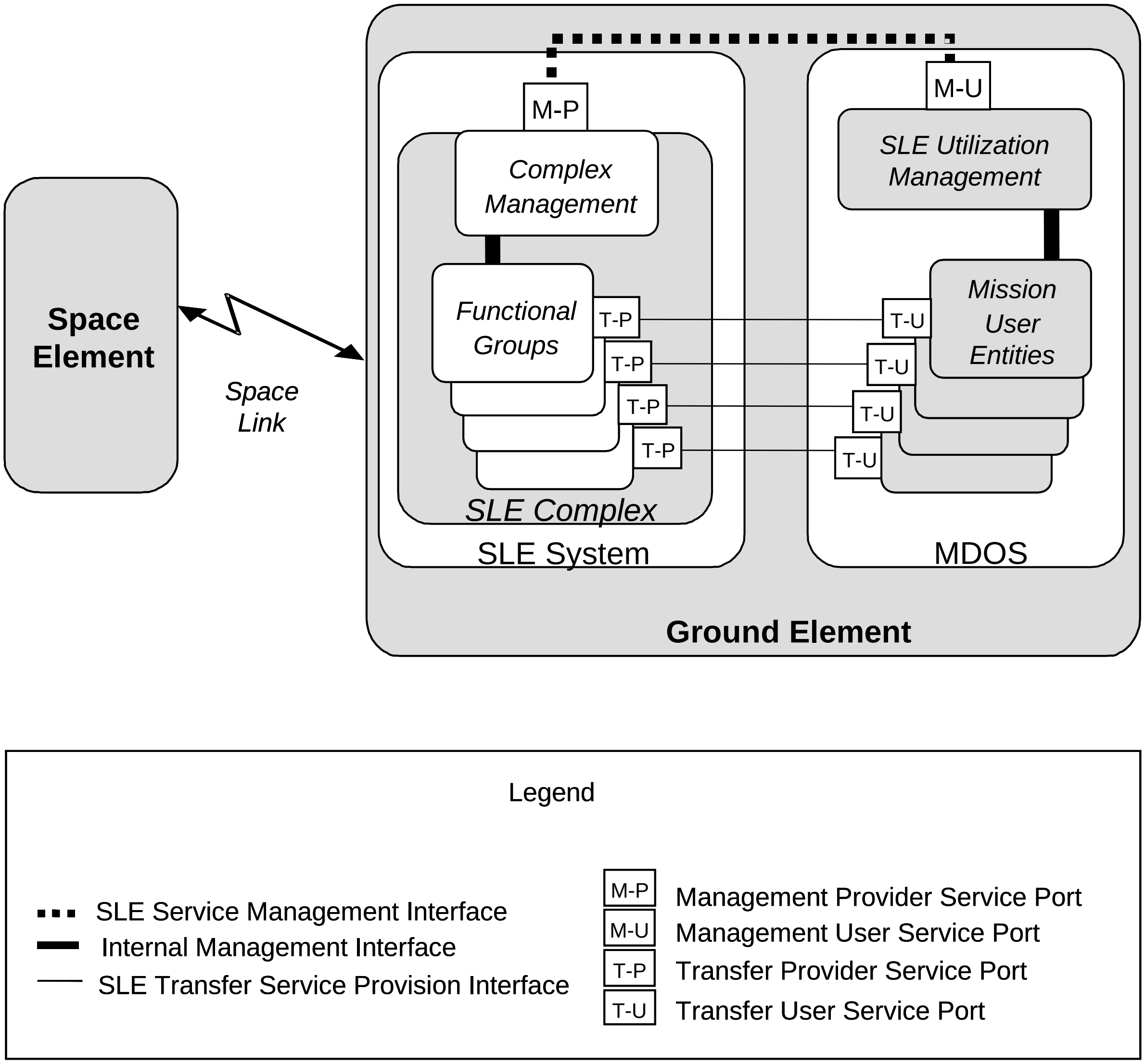


Figure 5-3: SLE Service Management Architecture

All Functional Groups in an SLE Complex are under the authority of a single SLE Complex Management. Figure 5-4 shows an example of an SLE Complex consisting of four Functional Groups managed by a single SLE Complex Management.

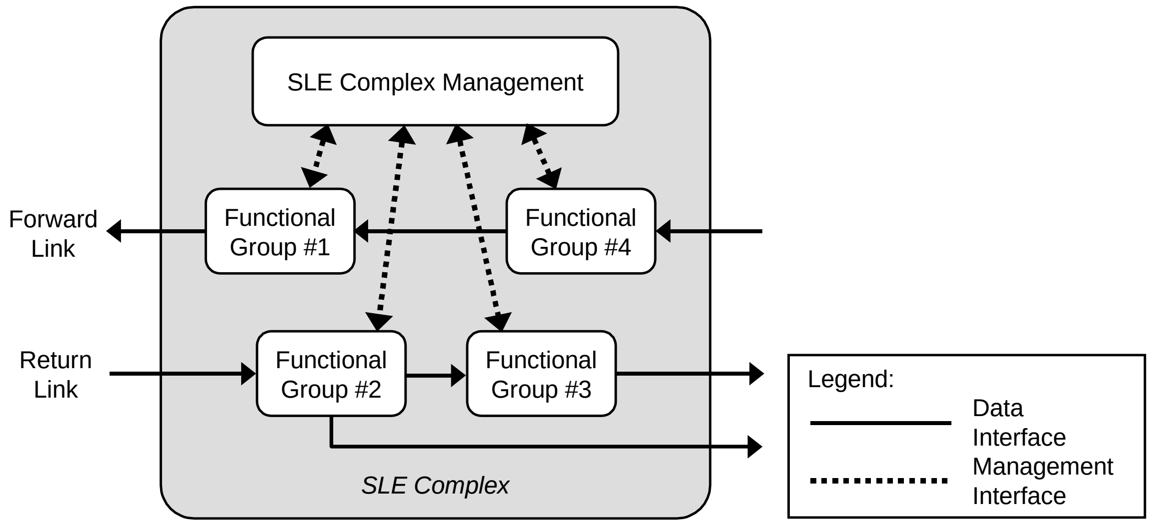


Figure 5-4: Example of Functional Group Configuration within a Complex

SLE Service Management is accomplished through the management of the functions performed by the SLE Complex that provides the SLE services.

If more than one SLE Complex provides a service, SLE Utilization Management coordinates with the SLE Complex Management in each SLE Complex to provide the services the mission requires. SLE Utilization Management must also coordinate and resolve conflicts between multiple service users. These concepts are illustrated in figure 5-5.

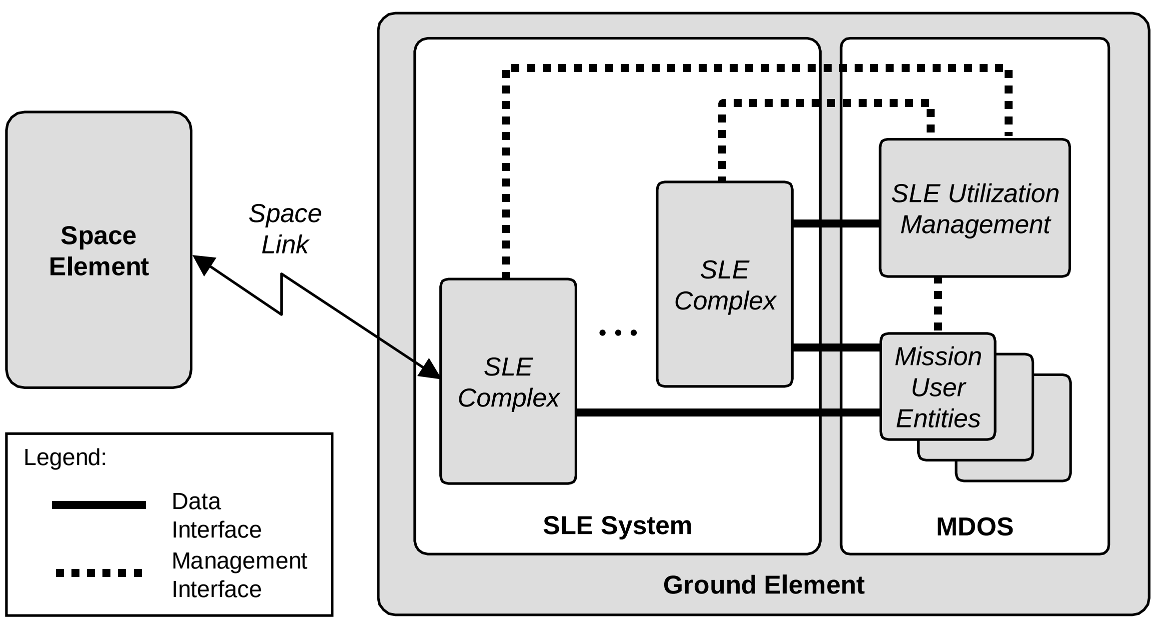


Figure 5-5: SLE Complex/MDOS Interfaces

5.4.2 SLE UTILIZATION MANAGEMENT

SLE Utilization Management is the function within the MDOS that coordinates the requests by users for space link and SLE services from the Complex.

Utilization Management:

a) Requests periods of provision of space link and SLE transfer services;

b) Provides RF, modulation, space link and SLE service configuration information;

c) Provides Trajectory Predictions;

d) Interfaces with Mission User Entities within the MDOS to enable the execution of SLE services and to collect status information.

5.4.3 SLE COMPLEX MANAGEMENT

The SLE Complex is a collection of resources under a single management authority. It may be a single ground station, a network of ground stations or, in the case of packet services provided to payload control centers, a spacecraft control center. The space mission uses the services that the SLE Complex provides to enable the MDOS to communicate with the mission spacecraft.

SLE Complex Management controls the extent to which SLE Utilization Management can affect actual SLE Complex resources. Because SLE Complex Management acts as the intermediary for SLE Utilization Management, only those aspects of the resources of an SLE Complex that SLE Complex Management chooses to expose are visible to SLE Utilization Management for management operations.

Complex Management:

a) Negotiates types of service, number of service instances and the length of the Service Agreements with UM;

b) Responds to requests from the UM for individual space link sessions;

c) Provides configuration information to the internal resources of the Complex in order to enable production and provision of SLE services and monitors their correct functioning.

5.5 TIME SPAN OF SLE AGREEMENTS AND PACKAGES

5.5.1 GENERAL

This subsection presents the terminology used to describe the lifetime of an SLE service. As noted in 1.6, several terms that were defined in the *Cross Support Reference Model* (reference [8]) are used in this Report. These include *Space Link session*, *service instance*, *Service Package* and *Service Agreement*.

The subsection ends by addressing the SLE Service Package definition phase and utilization phase.

5.5.2 PHASES OF SLE SERVICE PACKAGE LIFETIME

The SLE Service Package lifetime consists of two phases: a definition phase and a utilization phase. Both of these phases occur for each SLE Service Package. An example of the SLE Service Package lifetime phases is given in figure 5-6.

The arrangement made between a mission and an SLE Complex is formalized in an SLE Service Agreement that sets out the overall arrangements for cross support between the two parties. The details of the cross support are elaborated in the definition phase, within the capabilities and constraints drawn up in the Service Agreement.

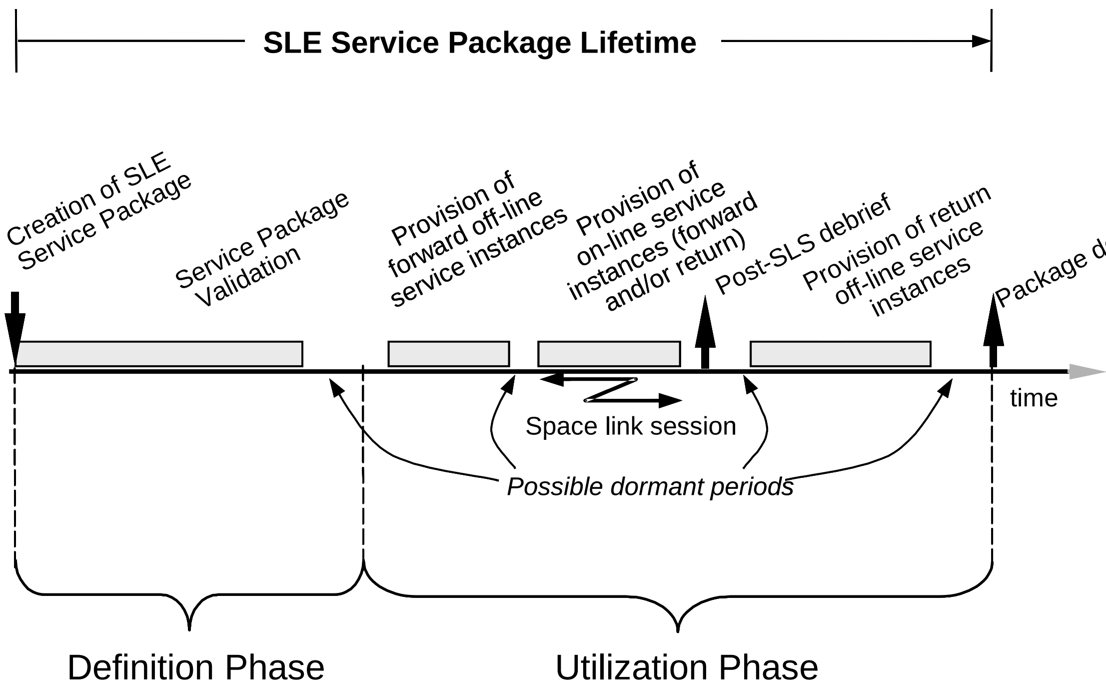


Figure 5-6: Example of Service Package Lifetime Phases

5.5.3 SLE SERVICE AGREEMENT PERIOD

The SLE Service Agreement period is the time during which an SLE Complex provides a space mission's SLE Utilization Management with the capability to create SLE Service Packages. It is also the time during which SLE Complex Management provides the SLE services defined by those packages, within the constraints of the SLE Service Agreement.

As illustrated by figure 5-7, many Service Packages may be provided over the course of an SLE Service Agreement period. Some of these may be related to a single Space Link session, while others may provide offline SLE transfer services related to two or more Space Link sessions.

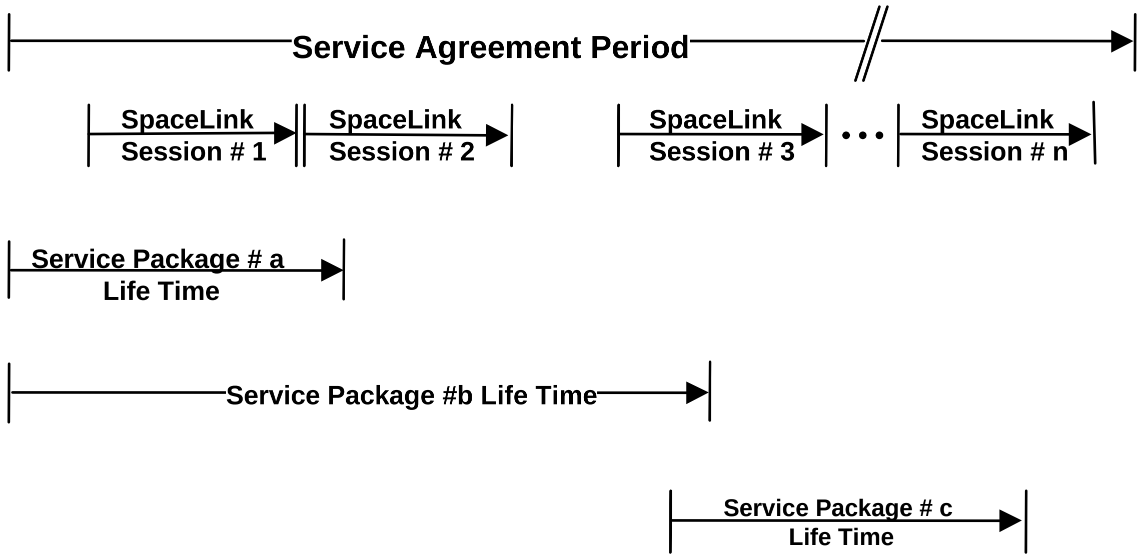


Figure 5-7: Example of Service Agreement Period

The SLE Service Agreement period often encompasses the entire operational lifetime of the supported spacecraft. This is the case when a mission is designed to use ongoing support from an SLE Complex, with a long-term SLE Service Agreement established before launch. A situation such as this may involve several Service Packages per day and hundreds (or thousands) over the agreement period. Other SLE Service Agreements, however, may be much shorter, as when a mission uses the services of a particular SLE Complex only for launch support, during an orbit maneuver, or for emergency operations. Typically, the SLE Service Agreement period spans multiple Space Link sessions.

5.5.4 SLE SERVICE PACKAGE LIFETIME

The SLE Service Package lifetime spans the time during which the SLE Complex provides the mutually agreed-upon services defined in an SLE Service Package and the time during which its contained SLE transfer service instances exist.

5.5.5 SERVICE INSTANCE PROVISION PERIOD

The service instance provision period is the time during which a service instance (i.e., the capability to transfer one or more SLE data channels of a given type) is provided. Figure 5-8 shows examples of three instances of the RSP transfer service.

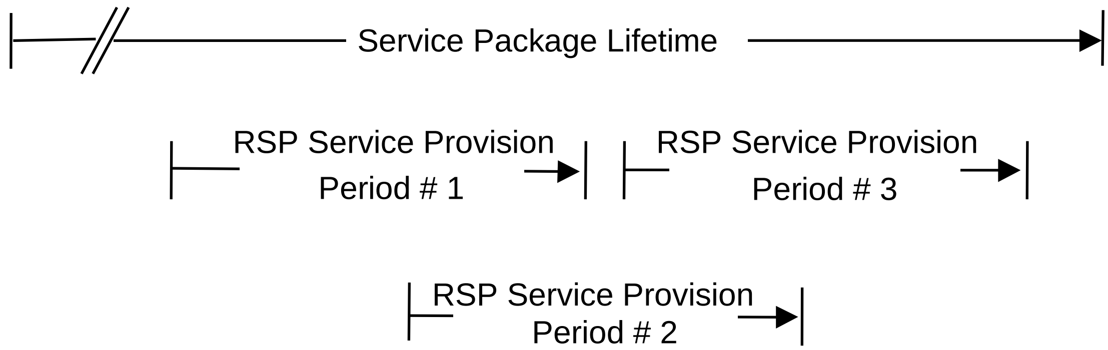


Figure 5-8: Example of Service Provision Period

5.5.6 ONLINE AND OFFLINE SERVICES IN SLE SERVICE PACKAGES

Each online SLE service instance of an SLE Service Package (which may contain several SLE service instances and span several Space Link sessions) is related to a single Space Link session, although it may begin before the start time and extend beyond the end time of the Space Link session.

Offline SLE service instances may exist at any time during the SLE Service Agreement period, subject to the conditions agreed upon in the SLE Service Agreement and in the SLE Service Package definition. Offline SLE service instances of a given package may be related to a Space Link session provided as part of that package or to Space Link sessions provided under one or more other SLE Service Packages.

Figure 5-9 shows an example of an SLE Service Package that provides both online and offline services. In this example, the SLE Service Package #k is a set of one forward online service instance and two return service instances (one online and one offline) associated with Space Link session #n.

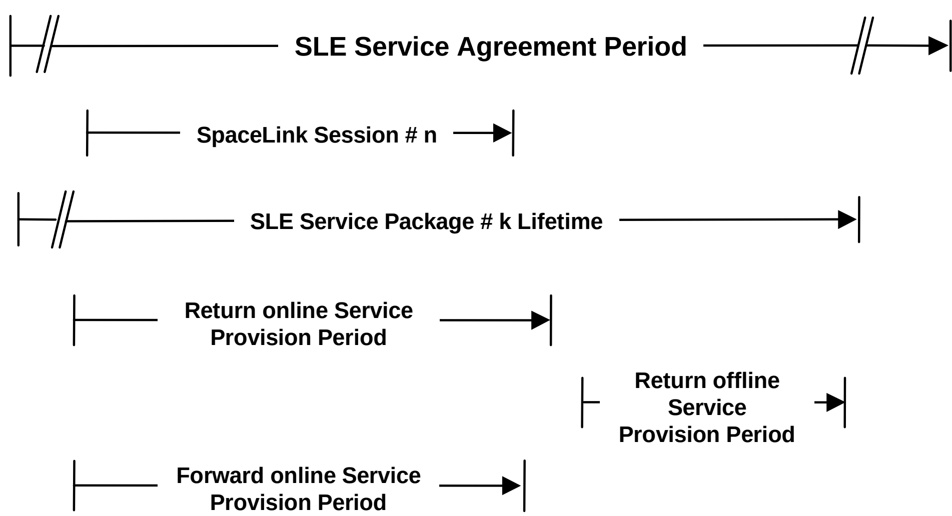


Figure 5-9: Online and Offline Services in One SLE Service Package

5.5.7 SLE SERVICE PACKAGE DEFINITION PHASE

During the definition phase, SLE Utilization Management defines and agrees a Service Package with SLE Complex Management. Parameter values, including schedule parameters, for SLE data transfer service instances are selected within the bounds specified in the SLE Service Agreement and SLE Configuration Profiles. The attributes of the Service Package provide:

a) selection of the SLE data transfer services to be provided by the SLE Complex (within the framework of the SLE Service Agreement);

b) identification of the interfaces at which these SLE data transfer services are to be made available;

c) identification of the SLE data transfer services that must be provided to this SLE Complex by the Space Element or by other SLE Complexes;

d) specification of mission-specific information needed by SLE Complex Management to configure SLE service production within the SLE Complex;

e) identification of the eventual consumers of SLE transfer services (i.e., Mission User Entities or other SLE Complexes);

f) specification of the schedule for provision of

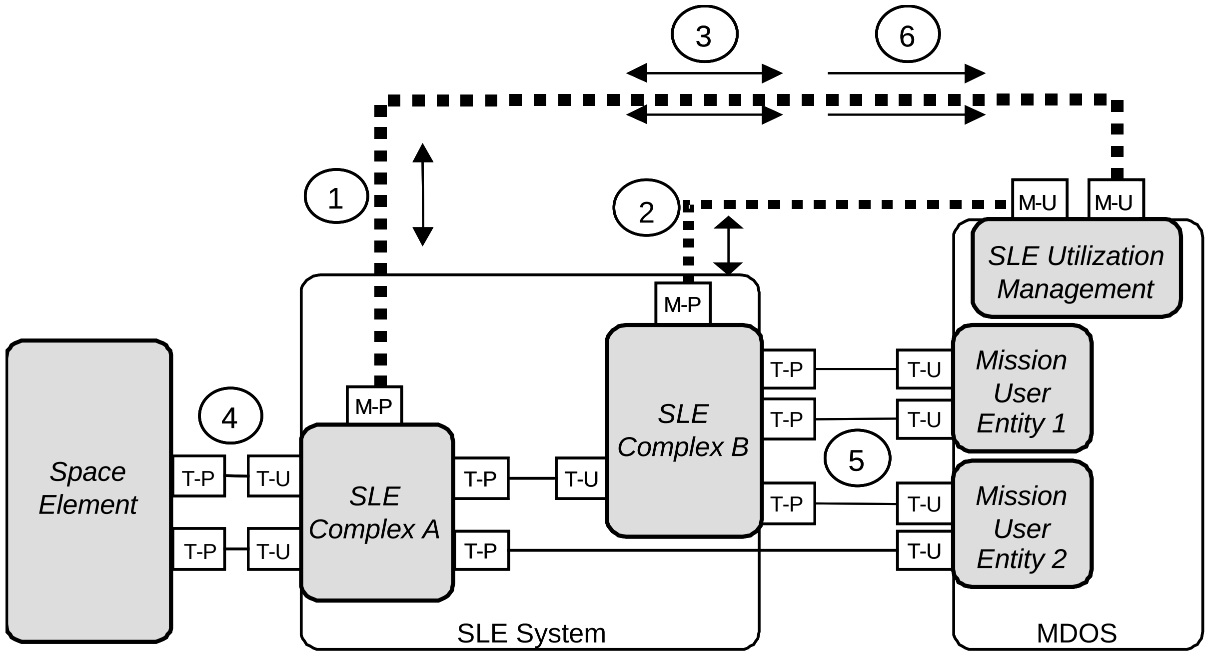
   1) online instances for a Space Link session, and

   2) offline instances of a given SLE data transfer service;

g) specification of reporting by SLE Complex Management to SLE Utilization Management.

A dormant period may occur after creation of the Service Package is complete and before provision of SLE service instances begins.

Figure 5-10 presents an example in which each of two SLE Complexes is requested to provide an SLE Service Package for a mission. The management operations that create these packages are shown as steps 1 and 2 in the figure. The remaining steps in this figure are addressed in the key to circled numbers.



Key to Circled Numbers

1	ServicePackage created at Complex A through management operations
2	A second servicePackage is created at Complex B
3	Management control of SLE transfer services at both Complexes
4	Physical RF channel provided during space link session (this applies to all data transfers between the space element and the Complex)
5	SLE transfer services provided to mission user entities within the MDOS (this applies to all data transfers between the SLE system and the MDOS)
6	Debriefing reports from each Complex to SLE Utilization Management

Figure 5-10: Example of Two Service Packages at Two SLE Complexes

5.5.8 SLE UTILIZATION PHASE

The utilization phase of an SLE Service Package is the period during which the SLE Complex provides the mutually agreed-upon SLE data transfer services. The utilization phase consists of *service instance provision periods*, possibly separated by dormant periods during which no service instances are provided. SLE Complex Management provides SLE Utilization Management with debrief reports following service provision periods. Debriefs may occur at any time following a service provision period.

SLE service provision periods occur as scheduled in the SLE Service Package. During the service provision period, the SLE Complex transfers data to or from the user, according to the particular SLE service being provided. In the case of an SLE Service Package that includes one or more offline SLE transfer service instances, there may be dormant periods between these offline provision periods, or between an online SLE service provision period provided at the beginning of the utilization phase and the first offline SLE service provision period of the SLE Service Package.

The provision of the SLE transfer service instances to the Mission User Entities is controlled in two ways:

a) through configuration operations in the data transfer service, invoked by the MUE;

b) through management operations invoked by SLE Utilization Management.

In the example shown in figure 5-10 above, the management operations that SLE Utilization Management uses to monitor and control these packages are shown as step 3. The physical channel (radio frequency or RF) transfer is shown as step 4, and the provision of SLE transfer services to the mission user entities within the MDOS is shown as step 5.

The utilization phase ends with a debrief to allow the SLE Complex and SLE Utilization Management to exchange information about the SLE Services that were provided throughout the SLE Service Package lifetime. In the example shown in figure 5-10 above, the management operations that provide debrief reports on the two SLE Service Packages that have been provided for the mission are shown as step 6.

5.6 OUTLINE OF SLE SERVICE MANAGEMENT SERVICES

5.6.1 OVERVIEW

5.6.1.1 General

The *SLE Service Management Service Specification* (reference [9]) defines a set of services for the standardized exchange of management information associated with a subset of the SLE data transfer services, those associated with TT&C ground station services. However, the SLE service management services are generic enough to cover all the data transfer services, including those that have not yet been specified formally. The services are:

– *Service Agreement*,

– *Configuration Profile*,

– *Trajectory Prediction*, and

– *Service Package*.

In addition, during the execution of the service package, some service providers may provide status reports (service execution status reporting) and/or allow the UM limited ability to modify minor operational characteristics of the provided services in near-real time.

The general timeline is illustrated in figure 5-11. In this figure, T0 is the start of execution time of the set of services requested, and T1 is the termination time.

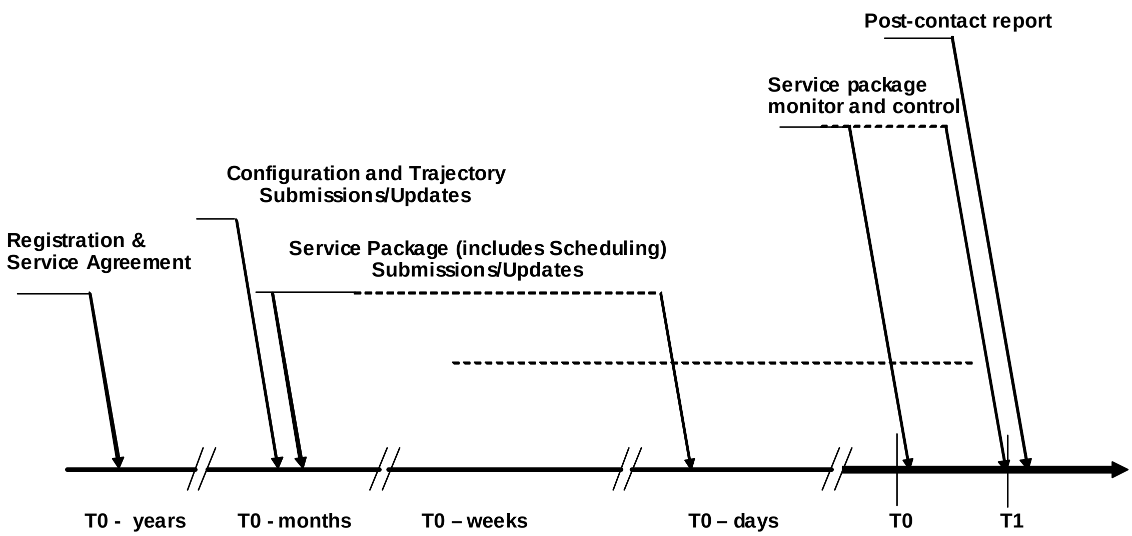


Figure 5-11: Nominal Interaction Timeline

Within this timeline, certain services may occur more than once. It is assumed that the multiple occurrences are still within the general constraints of the timeline. For example, Trajectory Prediction services may occur more than once for a given Service Package. This also applies to Configuration Profile updates. A user can expect that if a subsequent Trajectory Prediction submission is confirmed (in other words, an updated trajectory is to be used), it will be used for the applicable SLE Service Packages.

5.6.1.2 Use Case Example

The use case shown in figure 5-12 assumes that the Service Agreement is already in place.

In this scenario, UM begins by submitting Trajectory Prediction data and one or more Configuration Profiles. Initial versions of these will have been submitted with the Service Agreement, and further iterations will provide increasingly accurate data to prepare for the execution of the mission.

The user will request available schedule time opportunities via a mechanism not covered by SLE service management. These opportunities will be returned to the user by the provider. The UM can then submit a Service Package for one or more SLE services, referring to the Trajectory Prediction and Configuration Profiles submitted already. The Service Package submission includes the spacecraft contact time requested by the user. The start and stop times of the requested contact can be flexible to allow the CM's ground station scheduling system to optimize the schedule for all the missions it is supporting. Once the contact is scheduled, the CM replies with a positive commitment to the UM.

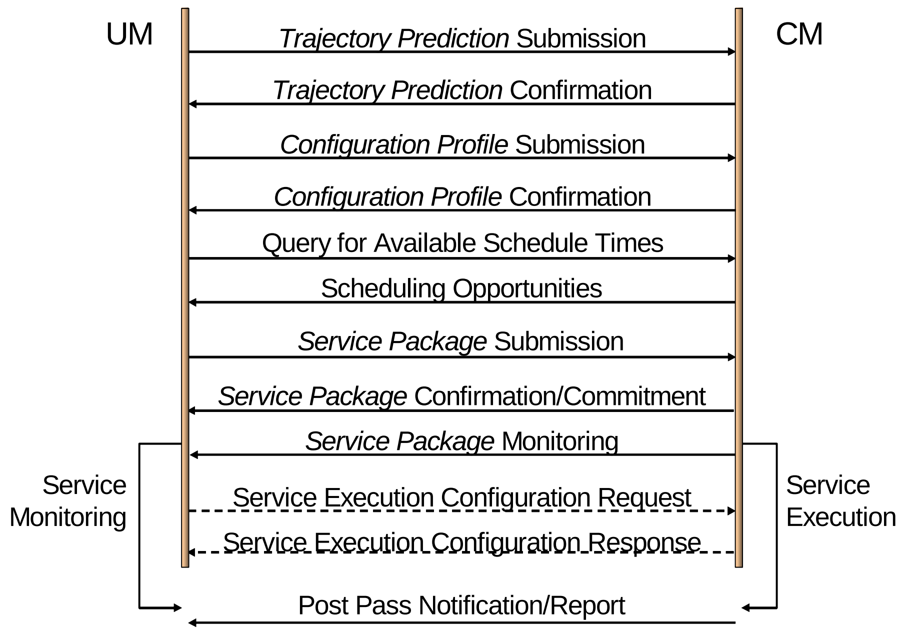


Figure 5-12: Nominal Use Case

At the agreed-upon time, the CM makes the SLE data transfer service ports available and the MUE binds to them. As service execution begins, the CM may provide the UM periodic updates about the execution. After the Service Package has completed its execution, the CM may send a notification and/or a report to the UM for its records and for accounting purposes.

5.6.2 SERVICE AGREEMENT ESTABLISHMENT

A user should first be registered with a service provider as the first step in establishing a service relationship between the two. The information exchanged as part of registration process includes security and authentication credentials (including expiration dates and re-authentication criteria), points of contact, and general bounds and constraints for subsequent Service Agreements. Registration can occur years before the first support period is provided to a mission. The registration process is not currently addressed by SLE service management.

Upon successful registration, the spaceflight mission establishes one or more Service Agreements with service providers. Each Service Agreement is limited in scope to a single mission. Negotiation of a Service Agreement occurs on the order of years in advance of the first support period provided to the mission.

The Service Agreement contains information about the services to be provided, including spacecraft communication characteristics, static and default Configuration Profile parameters, nominal frequency and duration of contacts, nominal trajectory and so on. It also specifies the range in which the exact values of parameters in Service Packages, Configuration Profiles and Trajectory Predictions are allowed to fall. The information contained in the Service Agreement assists the provider to determine the resources needed to support the mission (e.g., RF equipment, data storage, terrestrial network bandwidth).

Service Packages, Configuration Profiles and Trajectory Predictions are submitted within the scope of the Service Agreement. The parameter values for the Service Packages, Configuration Profiles and Trajectory Predictions are validated against the limits and constraints imposed by the applicable Service Agreement. A mission may have several Service Agreements if the level of support is to vary among the different phases of spacecraft operation. For example, a planetary mission might have separate Service Agreements with a provider for launch and early flight, cruise, and target planet exploration phases, if each phase represents a different level of support and/or requires different resources e.g., S-band or Ka-Band antennas.

In addition to the definition of parameter boundaries and constraints, the Service Agreement also contains configuration data that is considered static for the lifetime of the Service Agreement and hence does not need to be defined for each Configuration Profile or Service Package.

It is the responsibility of the CM to ensure that the negotiated parameter values in a Service Agreement are consistent with the capabilities that are available (or will become available) in the Complex during the Service Agreement period.

Registration and Service Agreement establishment are not (yet) standardized processes. The content, format, and media of the document(s) used to register the mission and to establish the Service Agreement(s) are agreed by the SLE service provider(s) and the spaceflight mission organization(s). CCSDS may address standardization of registration and Service Agreement establishment at some time in the future, at which point the concept will be updated to include them.

Although the data format and methods of information interchange are not standardized for the creation of Service Agreements, the *SLE Service Management Service Specification* (reference [9]) defines a standard query that the UM can invoke to obtain the data from the Service Agreement that are relevant to the SLE service. This is needed to validate Configuration Profile and Service Package service invocation against the information contained within the Service Agreement.

5.6.3 CONFIGURATION PROFILE

A Configuration Profile specifies a reusable set of space link and SLE services parameters that are established between UM and CM for supporting a mission spacecraft during the lifetime of the Service Agreement.

Once established, a Configuration Profile can be referenced by any number of Service Packages. Configuration Profiles allow the full set of configuration parameters to be defined independently of the Service Packages. It allows the clear separation between the reusable configuration data of the Configuration Profile from the dynamic schedule information contained in Service Packages.

The use of Configuration Profiles is well-suited to the majority of spacecraft that have one or more well-defined modes of TT&C operation, e.g., housekeeping, high rate instrument operation, tracking, and various combinations thereof.

5.6.4 TRAJECTORY PREDICTION

A TT&C Complex uses trajectory data to derive the pointing angles and Doppler compensation settings needed to acquire and track the mission spacecraft. UM is responsible for providing the trajectory data necessary to support contacts.

A Trajectory Prediction has a unique identifier and contains the trajectory data in one of the following formats:

a) one of the two standard message formats defined by CCSDS for use in transferring spacecraft orbit information between space agencies (see reference [19]):

   1) the Orbit Parameter Message (OPM), either in plain text or in XML format,

   2) the Orbit Ephemeris Message (OEM), either in plain text or in XML format;

b) a bilaterally agreed message format.

The allowed trajectory data formats are restricted to the ones recorded in the Service Agreement.

5.6.5 SERVICE PACKAGE

A *Service Package* holds information about the types of SLE transfer services to be executed and the periods the services that are to be provided, the end users that access the services, and the agreed configuration(s) for the space link (where relevant) and SLE production processes, for specific space link sessions. The services in a Service Package may be related for any number of reasons; e.g., forward and return RF carriers may be paired by coherency relationships.

Among the space link and SLE service capabilities facilitated by the Service Package are:

a) Multiple forward and return space link carriers, each with its own preferred and acceptable start/stop time windows.

b) Multiple SLE transfer service instances, each with a start/stop time specified relative to the start/stop times of the associated space link carrier.

c) Multiple alternate *scenarios*, by which CM is able to reserve resources for several contingency support modes concurrently, and to change between them on relatively short notice from UM. This capability is envisioned for use when circumstances may require different service configurations (e.g., those associated with tracking through not only a nominal spacecraft trajectory change but also any of several preplanned contingency situations such as under burn/over burn/no burn). Because only one of the scenarios can be executing at any given time, the same Complex resources can be allocated concurrently to multiple scenarios of the same Service Package. Not all providers will support alternate scenario reservation, and those that do will likely limit its use to periods of critical operations.

d) Optional time-sequencing of events, by which UM specifies a time sequence of parameter changes to be automatically executed by the Complex at the specified times. Event sequencing allows for proper coordination between the space element and the tracking complex for situations such as the carrier being temporarily unavailable due to occultation by an intervening planetary body, etc. As with alternate scenario reservation, not all providers will support time sequencing of events.

5.7 SERVICE MANAGEMENT CONCEPT EXTENSIBILITY

5.7.1 INCREMENTAL ADOPTION

Defining separate service management services affords gradual adoption of the standard. For example, before an agency adopts the CCSDS formats for spacecraft trajectory or configuration profiles, it will be possible to continue to submit these under the legacy format, utilizing the reference in the standard service package.

In this way, agencies may achieve full SLE interoperability on an incremental basis, suited to the agency’s plans and commitments.

5.7.2 SERVICE PACKAGE EXECUTION STATUS REPORTING

Some SLE service providers will provide status updates once the service package begins execution. The information provided may include status information, performance statistics or updated configuration parameters. The information may be reported periodically, upon request from UM or in a summary report at the completion of service request execution.

Currently, there are no CCSDS standards specifying the content, representation, or timing of service package execution status reports. Each service provider that does provide such reports does so according to its local specification. However, this area is recognized as a candidate for future standardization efforts.

5.7.3 SERVICE CONTROL

Some TT&C service providers give the UM the capability to control certain parameters of the services as they are executing. This capability is normally restricted to parameters that affect only the service that is being provided to that mission (e.g. change of frequency).

Currently, there are no CCSDS standards specifying the content, representation, or timing of service control interactions between UM and CM. Each service provider that does provide such capability does so according to its local specification. However, this area is recognized as a candidate for future standardization efforts.

5.7.4 TRACKING SERVICE

CCSDS is in the process of defining cross support tracking services, which will incorporate the results of multiple agency studies. The specific concepts related to the management of tracking services will be added to SLE service management once the requirements are established.

5.7.5 MISSION PLANNING

A new service management service could be used to help automate mission planning. For example, a planning service request could be submitted to poll the service provider on viewing opportunities and estimated link margins (for example, predicted maximum return carrier symbol rate) given the proposed spacecraft geometry and proposed ephemeris.

The service provider in this case would return information as to what the user could expect given the hypothetical inputs.

In addition to the technical flight support predictions, estimated expenses for providing various scenarios of services could also be returned.

ANNEX A

ACRONYMS

This annex identifies and defines the acronyms that have been adopted in this Report.

AOS	Advanced Orbiting Systems
API	Applications Program Interface
APID	Application Identifier
ASN.1	Abstract Syntax Notation 1
CLCW	Communications Link Control Word
CLTU	Communications Link Transmission Unit
CM	Complex Management
COP-1	Communications Operations Procedure 1
COTS	Commercial Off The Shelf
F-AOSSP	Forward AOS Space Packet
F-AOSVCA	Forward AOS Virtual Channel Access
F-cVCDU	Forward Coded Virtual Channel Data Unit
F-pVCDU	Forward Proto Virtual Channel Data Unit
F-VCDU	Forward Virtual Channel Data Unit
FSH	Frame Secondary Header
FSP	Forward Space Packet
FTCF	Forward Telecommand Frame
FTCVCA	Forward Telecommand Virtual Channel Access
ISDN	Integrated Services Digital Network
ISO	International Organization for Standardization
MC	Master Channel
MCID	Master Channel Identification

MDOS	Mission Data Operation System
MMI	Man Machine Interface
OCF	Operation Control Field
PLOP	Physical Link Operating Procedure
RAF	Return All Frames
RCF	Return Channel Frames
RF	Radio Frequency
RFSH	Return Frame Secondary Header
ROCF	Return Operational Control Field
RSP	Return Space Packet
SCID	Spacecraft Identification
SLE	Space Link Extension
SLE-FG	Space Link Extension Functional Group
TC	Telecommand
TM	Telemetry
TT&C	Telemetry, Tracking and Command
UM	Utilization Management
VC	Virtual Channel
VCA	Virtual Channel Access
VCDU	Virtual Channel Data Unit

ANNEX B

GLOSSARY

Annotation..The extension of the Space Link services requires that information be added to the Space Link data units. The process of adding information to the Space Link data is called annotation.

Cross Support....................................The situation when one organization uses part of another organization’s data-system resources to complement its own system.

Forward Data ....All data sent from the ground element to the space element (e.g., telecommand).

Functional Group ....The fundamental building block of an SLE service; it is not decomposed further. Functional groups are derived from the layered functionality specified in the Space Link protocols.

Ground Element.................................The collection of systems and organizations based on the ground that provides SLE services used by a specified mission.

Mission...For the purposes of this Report, an undertaking to explore and/or utilize fields of interest using one or more spacecraft.

Online..For the purposes of this Report, the transfer of SLE service data through all or part of the SLE System during the time that the associated Space Link session is active.

Offline ....For the purposes of this Report, offline refers to the transfer of SLE service data through all or part of the SLE System at a time other than that during which the associated Space Link session is active.

Operations Phase...............................In the Operations phase, the user defines service requests that are sent to the Provider Organization, which in turn responds by scheduling the support. If the support cannot be scheduled, the fact is communicated to the user, who can in turn formulate new Service Requests.

Payload... The on-board equipment that directly relates to the purpose of the spacecraft's flight.

Return Data .... For the purposes of this Report, all data sent from the space element to the ground element (e.g., telemetry).

Schedule... For the purposes of this Report, a Schedule is a chronological list or timeline of spacecraft activities and allocated resources constituting the daily operations plan for mission support.

Scheduling.. For the purposes of this Report, the task of placing activities onto a timeline and allocating resources.

SLE Complex..................................... The functions that perform SLE System services can be distributed across multiple systems. This distribution is aligned with the layering of the Space Link services. The systems performing individual functions of a service may belong to different organizations and have varying size and structure. The systems performing SLE service are grouped into SLE Complexes by the organizations that implement them. Each SLE Complex has two components, a Service Provision component and a Management component.

SLE Complex Management.............. The component of an SLE Complex that manages space data transfer.

SLE Service .... The set of services that extend one of the CCSDS Space Link Subnetwork services, providing access to the ground termination of that service from a remote ground-based system. An SLE service supplies or consumes one or more channels of the same Space Data Channel type.

SLE Service Agreement..................... The agreement between the agencies that are engaged in Cross Support. The SLE Service Agreement defines the set of Service Packages that are to be supported over the lifetime of the SLE Service Agreement. The resources that will be accessible and the privileges that will be extended are identified.

SLE Service Instance .... The provision by an SLE Complex of the capability to transfer one or more SLE data channels of a given type, all of which are related to the same Space Link session.

SLE Service Package ....An SLE Service Package is the set of service instances, together with the specification of the characteristics of the production of those service instances, that are provided by one SLE Complex to one or more SLE Transfer Service users, with respect to one Space Link session.

SLE System..The global collection of systems and organizations that provide SLE services.

SLE Utilization Management ....The entity within an MDOS that interfaces with the Complex for use of SLE services.

Space Data Channel ....For the purposes of this Report, a virtual stream of space link data units of the same type, with a single unique identification.

Space Element....................................The systems and organizations on board the spacecraft, which provide SLE services used by a specified mission.

Space Link Service ....The Space Link service is the service provided over the Space Link Subnetwork to the user during a contact with the spacecraft.

User...For the purposes of this Report, an entity receiving services.

ANNEX C

CROSS SUPPORT SCENARIOS

C1 CROSS SUPPORT ORGANIZATION

Cross support may occur between the SLE System and an application in the spacecraft or mission domain. The SLE System builds on the Space Link standards by standardizing the SLE and Service Management protocols and services. Such standardization allows a mission to interface with the SLE System, concatenating SLE services by interconnecting SLE Complexes within the SLE System, no matter how or where the services are implemented.

Implementation of Cross Support is each organization's responsibility. In other words, the groupings of Functional Groups into SLE Complexes are at the discretion of each organization. However, it should be noted that there is a limited number of rational groupings that may be selected by an organization.

The following subsections illustrate the SLE service concept for providing Cross Support by presenting a set of scenarios that first illustrate how the functional distribution of SLE Complexes may differ between organizations and then show some examples based on previous implementations of (non-SLE) cross support.

C2 FUNCTIONAL DISTRIBUTION OF COMPLEXES

When discussing Cross Support, a functional rather than a physical view is needed. There are cases where different organizations use the same word for a facility but mean different functions. For example, in figure C-1, 'ground station' contains an extensive processing capability; the control center distributes files of user data as well as performing forward link functions. In figure C-2, 'ground station' has a limited capability; return link processing, offline processing, and command control are performed at different facilities.

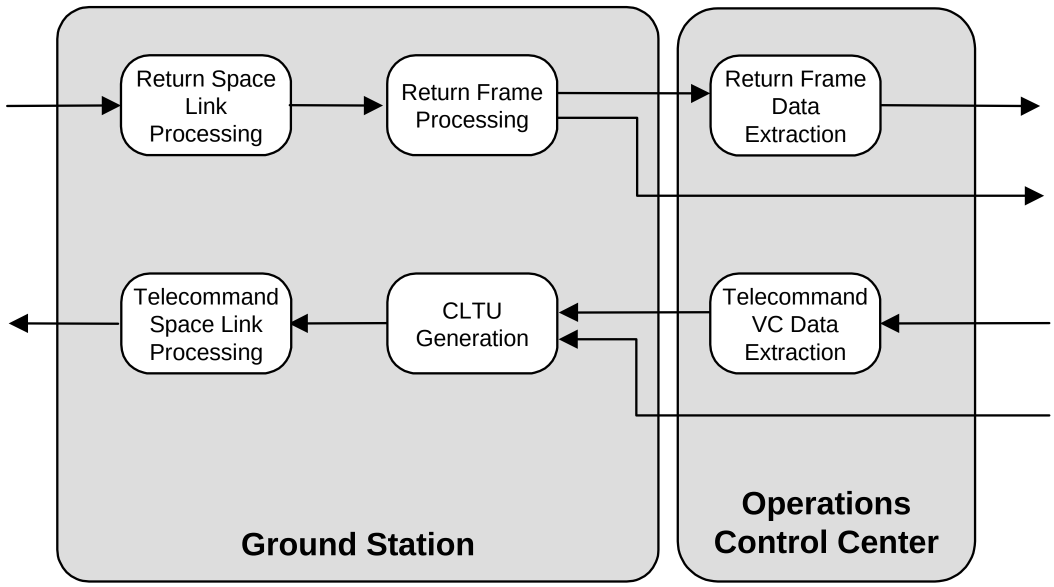


Figure C-1: Example of Ground Station with Extensive Capability

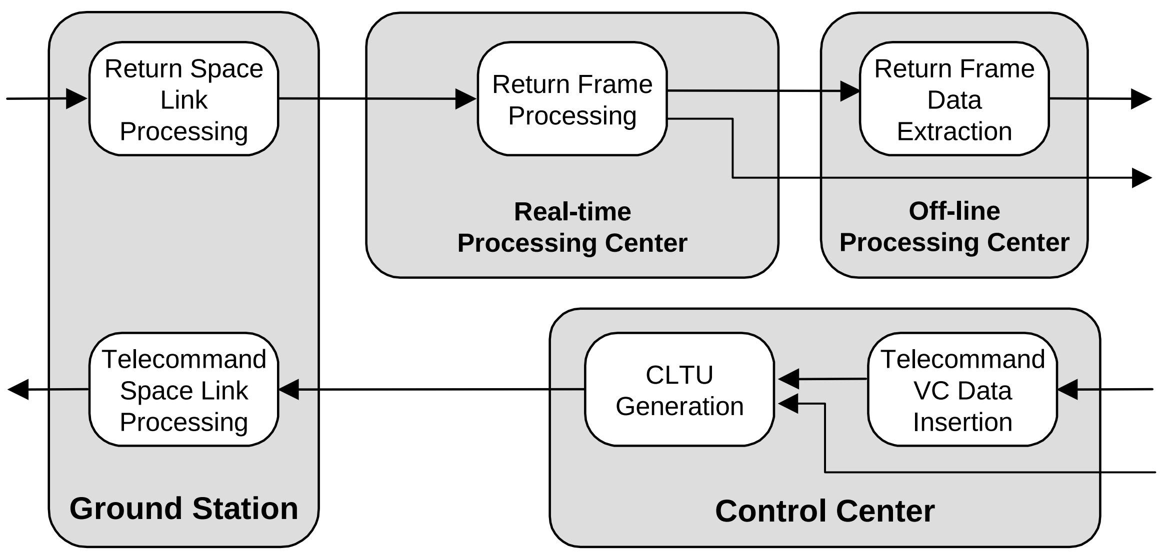


Figure C-2: Example of Ground Station with Limited Capability

An SLE Complex may contain the complete set of functions required by a mission (in which case the SLE Complex and the SLE System are the same), or a subset of functions. Figure C-3 is an example of an SLE System comprising two SLE Complexes. In this example, the Ground Terminal SLE Complex implements the Return Space Link Processing and Forward Telecommand Space Link Processing. The Remote Processing SLE Complex implements the Return Frame Processing, Return Data Extraction, CLTU Generation, and Telecommand VC Data Insertion. In this example, the mission uses only the Space Packet and Insert services.

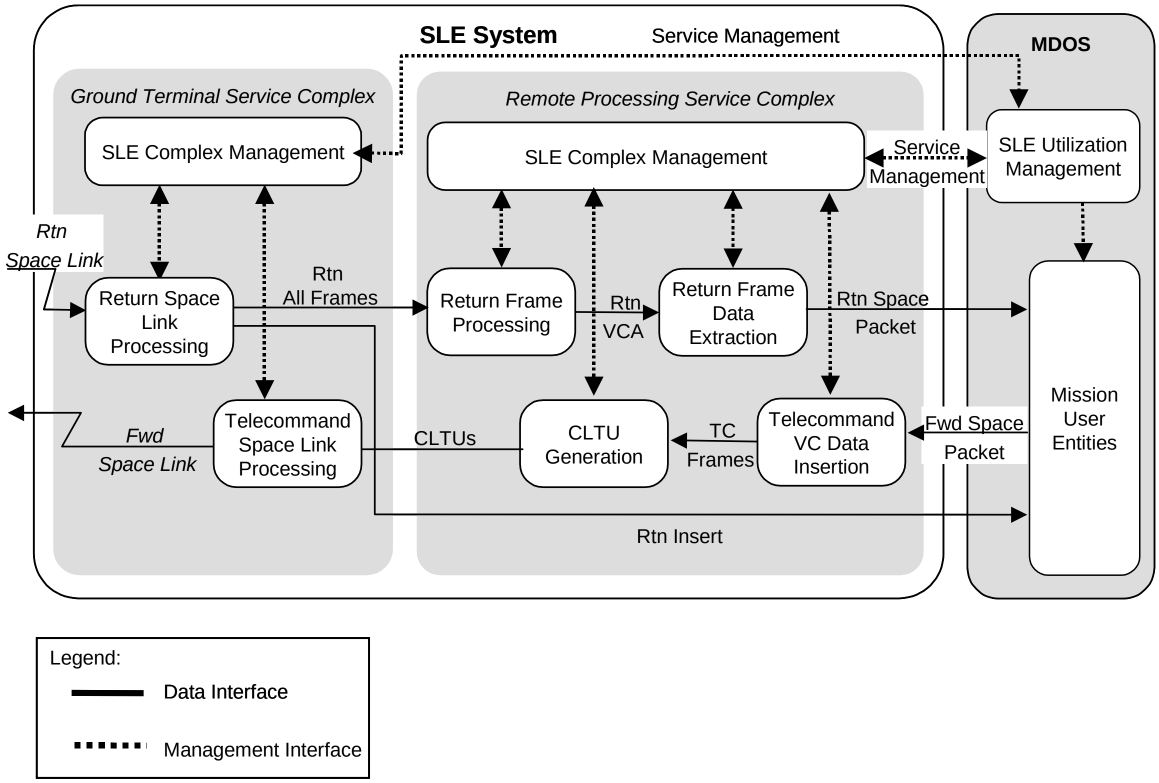


Figure C-3: SLE System with Two SLE Complexes

C3 CROSS SUPPORT SCENARIO ILLUSTRATIONS

This subsection illustrates examples of cross support drawn from the experiences of the space agencies that contribute to CCSDS. Although the agencies have implemented their systems differently, they must all perform common functions. This allows us to illustrate current agency implementations according to the Functional Groups described earlier in this document.

This subsection provides five scenarios of Cross Support that are generalizations and extensions of actual cross support implementations. Between them, these scenarios illustrate different functional allocations and different aspects of Cross Support that have been performed between agencies.

Cross Support Scenario #1, shown in figure C-4, shows the return link for a situation in which Agency B, which owns the spacecraft, may receive the return data either directly from its own SLE Complex, or from the Complex of another organization. The return data are processed and individual data units or files of sorted packets are distributed to users and to other agencies. In the figure, Agency A owns the payload and receives data from it directly. It feeds in processed data to Agency B, which also receives data from the spacecraft directly.

The Agency B complex in the center of the figure sends products to the MDOS, as represented by the three boxes on the right of the figure.

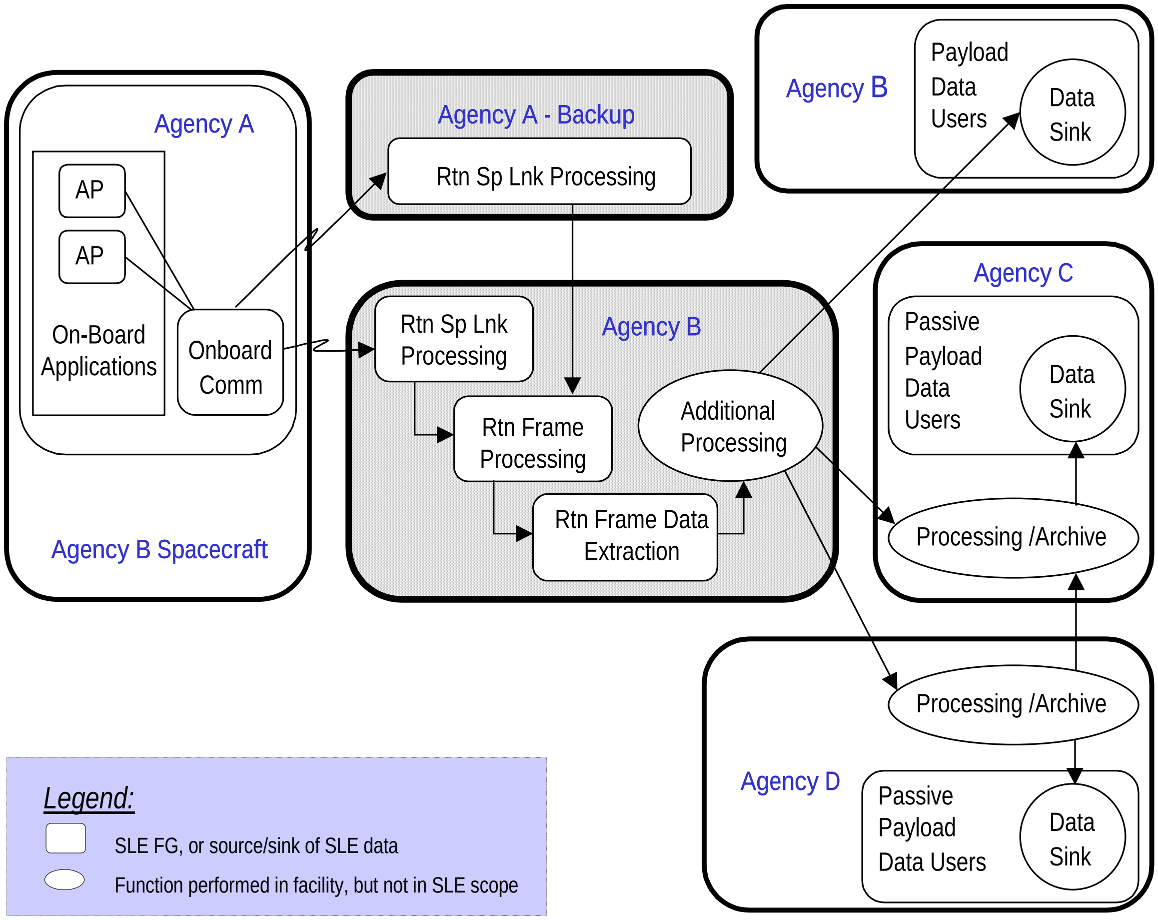


Figure C-4: Cross Support Scenario #1: Multiple User Agencies

Cross Support Scenario #2, shown in figure C-5, illustrates the case in which multiple minimal 'ground stations' each send all frames received during a pass to a single Complex. This Complex performs all the remaining return processing and distributes the data to users. Similarly, this Complex accepts forward data from users, processes it, and transmits it to the ground stations for transmission to the mission spacecraft.

In the figure, the spacecraft is shown on the left, the Complexes are in gray, and the MDOS is on the right.

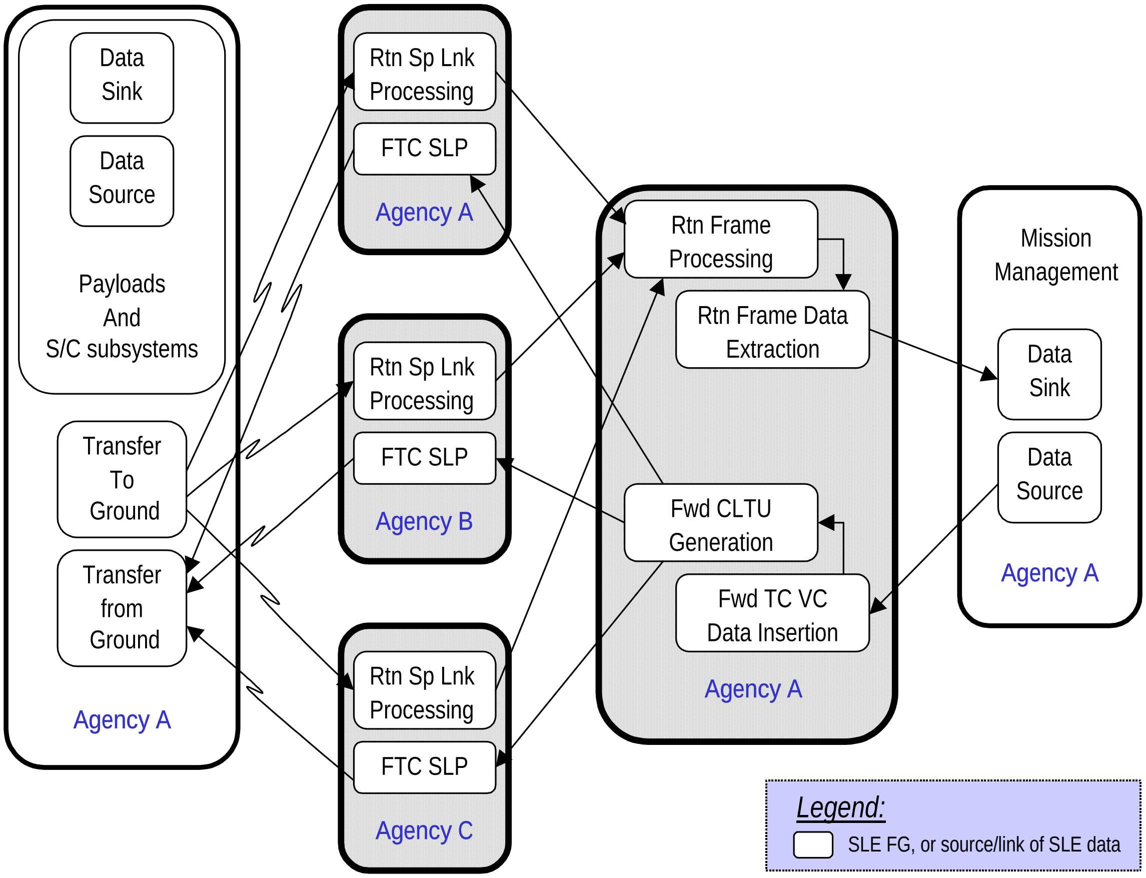


Figure C-5: Cross Support Scenario #2: Multiple 'Limited Capability' Ground Stations

In Cross Support Scenario #3, shown in figure C-6, Agency A typically sends data from its spacecraft to a network of ground stations, each of which performs all data processing functions through Return Frame Data Extraction. However, Agency B must use two facilities to process the data to this level. While this does not affect the service interface, it may affect the management interfaces between the agencies.

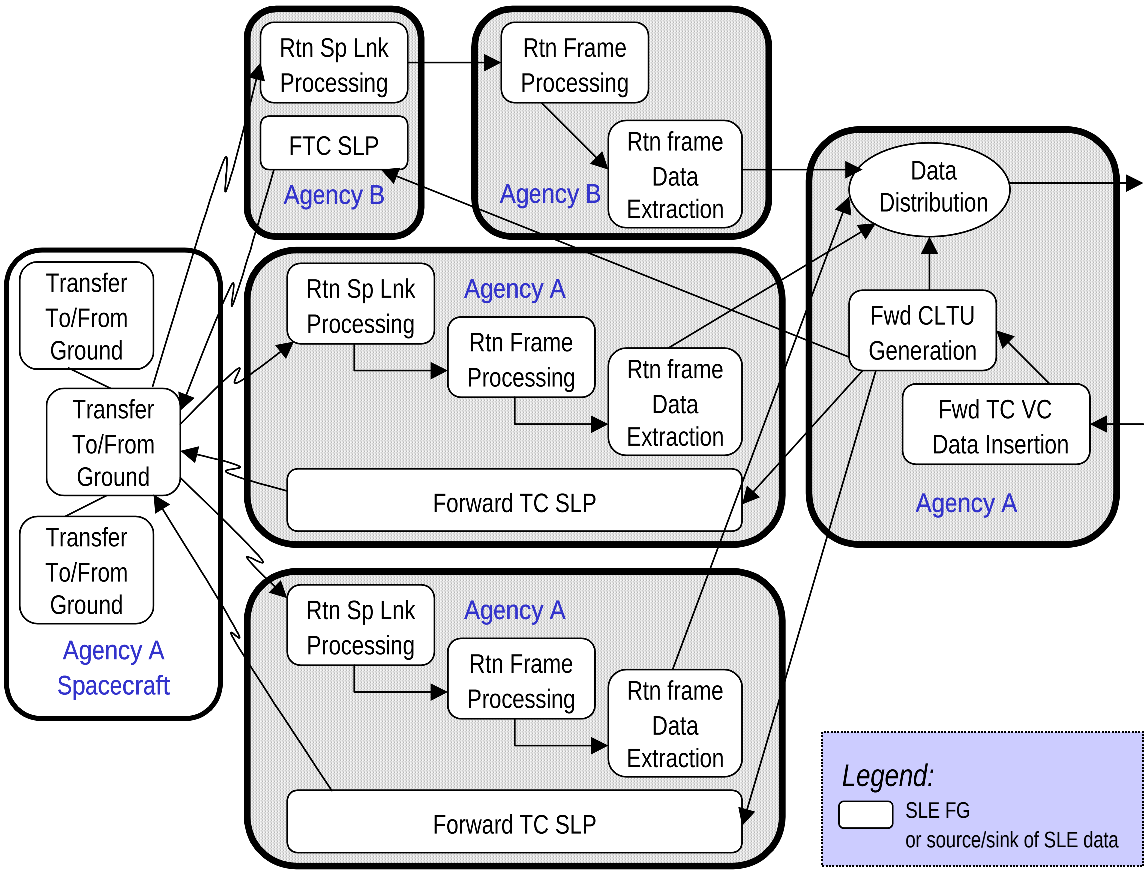


Figure C-6: Cross Support Scenario #3: ‘Extensive’ and ‘Limited’ Capability Ground Stations

In Cross Support Scenario #4, shown in figure C-7, Agency B provides a payload on Agency A’s spacecraft. Agency A is responsible for receiving the return data and forwarding processed data for the Agency B payload to Agency B, and also for accepting forward data from Agency B and forwarding it to the spacecraft. Agency B plans to develop its own return data processing capability in the future, and will be able to receive the return data directly in the future. Agency A will remain responsible for the forward link.

The ‘mission ground applications’ shown in this figure correspond to the SLE MDOS.

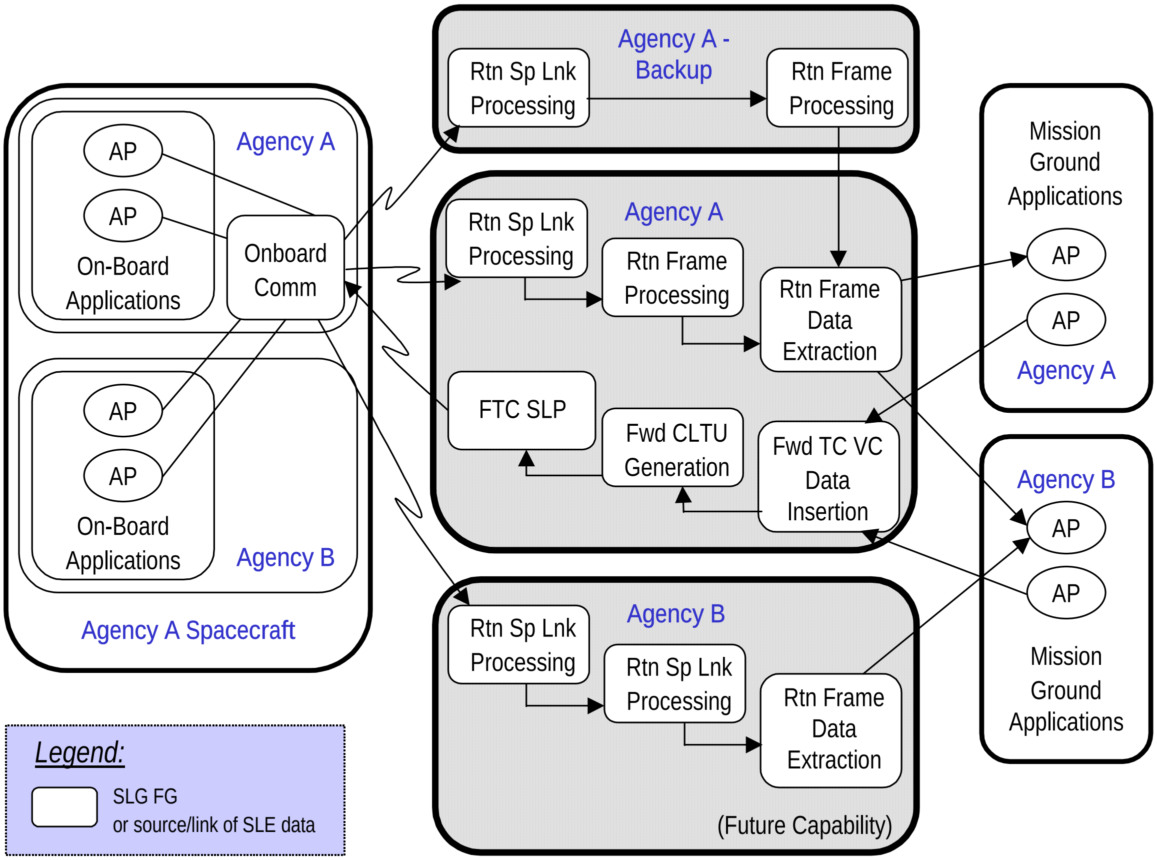


Figure C-7: Cross Support Scenario #4: Multiple 'Extensive' Capability Ground Stations

Cross Support Scenario #5, shown in figure C-8, illustrates Agency B receiving and processing data from the payloads on its spacecraft, including that provided by Agency A. On the ground, Agency B uses Agency A for backup.

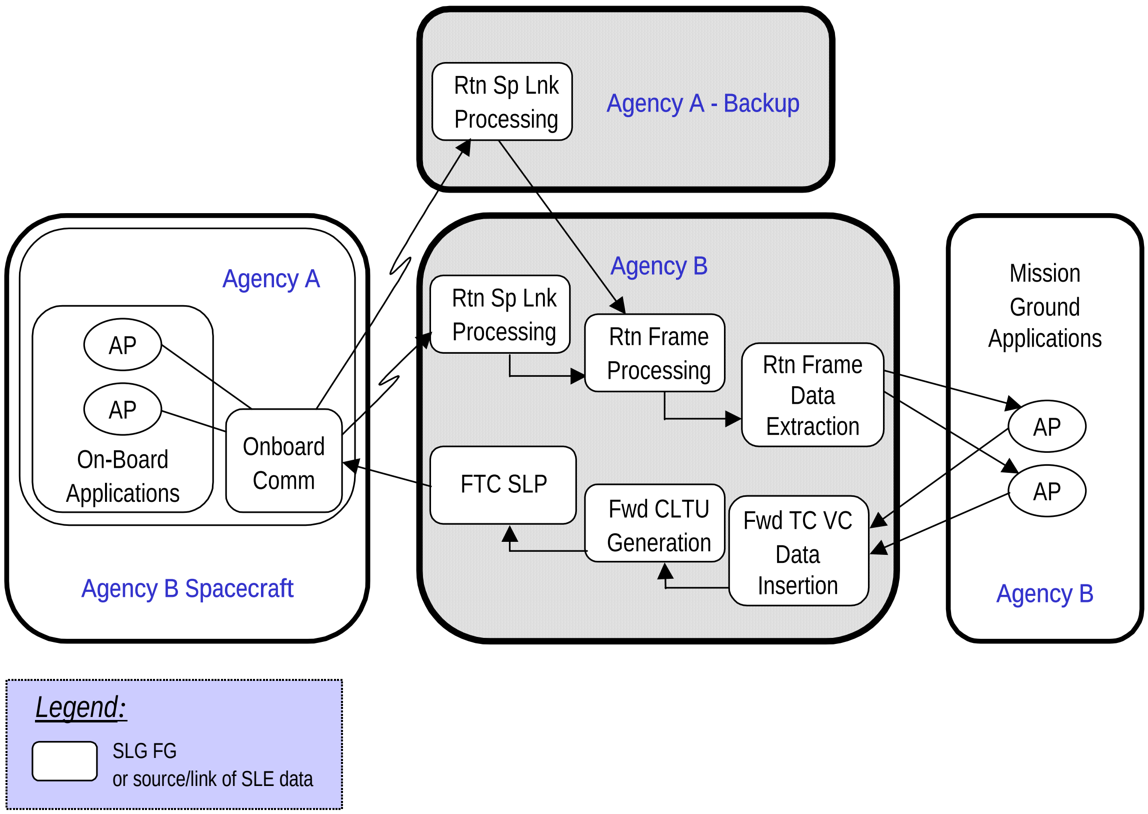


Figure C-8: Cross Support Scenario #5: Simple Cross Support Backup

ANNEX D

SLE SERVICE MANAGEMENT:
ANTENNA POINTING SCENARIOS

D1 GENERAL

The basic prerequisite for any space link processing at a ground station is the ability to point an antenna to a location in space where a spacecraft is expected to come into sight.

This, of course, requires that the antenna pointing subsystem in the ground station has timely access to Trajectory Prediction information for the target spacecraft.

Before we see step by step how the antenna pointing is performed, let us lay out a simple yet typical scenario where handling of trajectory information is vital:

SCENARIO

An SLE Complex is entrusted with acquiring the space link of a spacecraft during launch, early orbit operations, and routine phase.

– The launch is successful but it has slipped for a few minutes.

– Over the next hours, several subsequent orbit adjustment-manoeuvres establish the target orbit.

– The spacecraft undergoes commissioning and then enters its routine operations phase.

In order to arrive at widely acceptable standards, the authors of the Recommendations have formed an interaction model capturing current practices in several agencies and it is assumed that SLE Utilization Management and SLE Complex Management behave according to the following general pattern:

INTERACTION MODEL

Trajectory information is computed by the spacecraft operations organization, i.e., the MDOS in the abstract terms of the *Cross Support Reference Model* (reference [8]). It predicts the trajectory of the target spacecraft in a ground station independent manner, for a certain time interval, relative to a certain reference system. The computed trajectory information is encoded, packed into a file and transferred in time from the MDOS to the destination SLE Complex Management. There it is deposited in a mission dedicated trajectory file repository. When antenna pointing becomes due, SLE Complex Management decodes the trajectory information, transforms it into some complex-internal reference system and translates it into micro-commands to be executed by the antenna drive.

Now we explore step by step how antenna pointing is performed.

D2 STEP 1: ESTABLISH THE INTERACTION FRAMEWORK

The currently available CCSDS Recommendations do not cover all elements of the interaction model and a number of them must be left subject to bilateral agreement between SLE Utilization Management and SLE Complex Management. In a phase, typically one year before the actual launch of the spacecraft, negotiations between mission customer and the service providing SLE Complex take place. Among many other topics, the prerequisite agreements for antenna pointing operations must be settled. These agreements concern the following:

- The specification of a common format for the Trajectory Prediction files. As long as there are no pertinent CCSDS Recommendations, this format specification covers: the choice of the commonly understood reference system for a trajectory specification, the general layout of the file (header, body) and specification of trajectory information elements (name, data type, engineering unit), and the encoding of trajectory information for transfer.
- The maximum size of the spacecraft dedicated repository for trajectory information.
- The path to the spacecraft dedicated repository for trajectory information.
- A protocol for file transfer and file management in the repository (add file, remove file), and security measures for the protection of the trajectory file repository. It is expected that a reference to an existing standard file transfer protocol (e.g., FTP or FTAM) and the agreement about suitable user identifications and passwords is sufficient.

D3 STEP 2: PRODUCE TRAJECTORY PREDICTIONS

Service Management (reference [9]) does not make any assumption about the actual production of Trajectory Prediction files by the mission organization. However, the file that will ultimately be shipped to the supporting Complex must adhere to the format conventions as negotiated in step 1 and, of course, SLE Complex Management must be prepared to read and interpret this format correctly.

D4 STEP 3: TRANSFER AND DEPOSIT TRAJECTORY FILES

Using the file transfer protocol, the directory path negotiated, and applying the security measures (step 1), the file is transferred and stored in the directory under a unique name (unique relative to the directory).

If the new file cannot be accommodated in the negotiated space, the transfer is rejected.

Before a retry, directory housekeeping (e.g., remove of outdated files) may be required to free sufficient space. This may be affected via the file transfer (and management) protocol or by other suitable means which are not covered in the SLE Service Management Recommendations.

It shall be noted that several Trajectory Prediction files may well co-exist in the directory.

D5 STEP 4: SELECT AN OPERATIONALLY APPLICABLE TRAJECTORY FILE FROM THE REPOSITORY

In order to support space link acquisition correctly during the execution of a Service Package, antenna pointing must be driven by an operationally applicable Trajectory Prediction.

Whenever SLE Utilization Management sets the applicable Trajectory Prediction file, a few tests are performed on the file: is it really deposited?, can it be read?, has it the agreed format and encoding?, does it fit with the time bounds of the Service Package?, can the antenna move along the desired trajectory arc? If something is discovered wrong with the prediction file the setting is rejected and an indicative diagnostic is returned to SLE Utilization Management. SLE Complex Management has no other choice than using the ‘old’ file for eventual antenna pointing.

If the trajectory file is OK, SLE Complex Management will convert it into a Complex-internal format, process it as required, and ensure that this information will be used with the next upcoming antenna pointing activity.

If an antenna pointing activity is going on at the moment when a new trajectory is selected, the new trajectory may be prepared, but it will be actively used only for the next upcoming antenna pointing, i.e., in general for the subsequent spacecraft pass (for the major exception see D8).

D6 STEP 5: DRIVE THE ANTENNA

So, what does ‘antenna pointing activity’ mean?

For pointing an antenna to a target spacecraft, SLE Complex Management must assign the antenna to that spacecraft.

Eventually, the antenna begins a timely move into a start position, which has been computed from the Trajectory Prediction for the time when space link acquisition is supposed to begin. SLE Utilization Management is notified when the antenna has reached its starting position.

Then, depending on the tracking mode, the antenna moves along the trajectory arc (programmed tracking) or follows the spacecraft, once it is locked on it (auto-tracking).

At the end of the pass, the antenna stops tracking the target spacecraft. At this point in time, a notification is sent to Utilization Management. Thereafter, the antenna may be reassigned to another spacecraft.

D7 STEP 6: ADJUST THE TIME TAG OF A TRAJECTORY PREDICTION

In our scenario, we assume that the launch of the target spacecraft slips for a couple of minutes. This means that the validity time of the launch-Trajectory Prediction must be adjusted in order to compensate for the delay.

SLE Complex Management must take this correction into account when it eventually converts the selected prediction into antenna specific commands.

If an antenna pointing activity is going on, the correction will become effective only at the next pointing exercise (for the big exception see D8).

D8 STEP 7: ENFORCED SWITCHING BETWEEN TRAJECTORY PREDICTIONS

Following our scenario, we enter the Early Orbit Operations phase. In this phase, one must be prepared for orbit maneuvers which may fail or perform with certain degradations. Usually, several alternative orbit predictions are prepared for the time after a maneuver. These alternative predictions are transferred into the directory (in fact, that is why this directory can hold several prediction files).

One of the effects of the maneuver may have been to move the spacecraft to a new orbit. In order to best track the spacecraft in its new orbit, the antenna must be switched immediately to use the best available file from the prepared Trajectory Predictions.

On the resource level, this invocation causes the antenna to move immediately (or as fast as possible without breaking apart) from the current position to a new position, which has been calculated on the basis of the new trajectory file and the applicable time-offset correction. Then, the antenna continues the movements with the new trajectory information.

D9 STEP 8: ROUTINE UPDATING OF TRAJECTORY PREDICTIONS

Typically, in the routine phase of spacecraft operations, Trajectory Predictions are gradually refined.

Such refined Trajectory Predictions files can be downloaded to the repository as in step 3 and then used for upcoming antenna pointing activities as described in step 4.

D10 STEP 9: DEBRIEFING

As an antenna is a scarce and expensive resource, antenna-time may be the limiting factor for cross support costs; therefore it is important to account for the usage.

ANNEX E

SLE SECURITY MECHANISMS

E1 GENERAL

SLE transfer services have two security mechanisms that are integral to the services: authentication and access control. This annex describes these two mechanisms, and provides descriptions of how the mechanisms are applied to the SLE transfer service operations.

E2 AUTHENTICATION

Authentication is the process of verifying that an entity (person, computer, etc.) is what it identifies itself to be. In the case of SLE transfer services, the entities of interest are the SLE service user and provider applications that associate with each other in a transfer service instance, for which authentication is applied to the operation invocations and returns. SLE transfer services provide for rudimentary authentication through the use of credentials. The use of credentials relies on the existence of a 'secret' (such as a password) that is shared by two entities (in this case, the SLE service user and provider), such that the presentation of the secret in a communication from one entity to the other authenticates the sending entity to the receiver. There are any number of authentication methods (that is, schemes for generating credentials), with the better, more powerful ones offering better protection against the compromise of the secret to a third party.

The ability to completely authenticate a peer SLE entity will most likely involve not only the authentication performed by the SLE entities themselves, but also capabilities provided by the underlying data transfer and management technologies used. Some of these underlying technologies may provide strong authentication capabilities, whereas others may provide little or no such capability. In order to complement the varying degree of authentication provided by underlying technologies, each SLE service instance is specified to have one of three *authentication levels*: 'all', 'bind', and 'none'. When 'all' is in force, all SLE operation invocations and returns carry credentials. When 'bind' is specified, only the BIND invocation and its return carry credentials. When no authentication is needed at the SLE application level, 'none' is specified.

E3 ACCESS CONTROL

Access control is the process of permitting access only to other entities have been explicitly granted access privileges. In the context of SLE, each transfer service instance is configured by Service Management to allow only one specified user to initiate that service instance. Access control is applied to every SLE transfer service instance.

Access control specifies which user may initiate a service instance, but it does not guarantee that the entity attempting to initiate the service instance is actually the user that it identifies itself to be. Authentication, combined with access control, ensures that it is indeed the privileged entity that is accessing the service.

E4 SECURITY-RELATED INFORMATION

E4.1 GENERAL

Figure E-1 illustrates the security-related information that is associated with the SLE transfer service user and provider entities. For the purpose of this description, an SLE transfer service entity is the set of user or provider resources allocated to the support of transfer service instance for the duration of that service instance. Some of these security-related parameters serve other, non-security purposes, but their description in this annex is limited to their security roles.

The SLE security-related information is conceptually distributed across two types of data structures: entity authentication structures and service instance structures. This organization of the information is conceptual because there is not a standard management interface defined for the exchange of this information, and so the actual organization is implementation-dependent. Real implementations may organize this information in any number of ways.

E4.2 ENTITY AUTHENTICATION STRUCTURES

The SLE transfer services make use of pre-defined SLE transfer service entity authentication structures, each of which encapsulates a particular profile of security characteristics. The SLE entity authentication structures are established between a Complex and MDOS to be used in service instances: one set for the entity identifiers that can be used to identify the MUEs of the MDOS, and another to identify SLE entities that can be used by the Complex when supporting the MDOS. Each system in the Complex and MDOS is configured with a set of *local* entity authentication structures (designating the identifiers that the system can use to identify itself) and *peer* entity authentication structures (designating the identifiers that can be used by peer systems supporting the SLE transfer services under the Service Agreement).

The entity authentication structures are managed separately from service instances: they are created, destroyed, and updated (e.g., change of password associated with an entity identifier) outside of the process by which Service Packages are created.

The local entity authentication structure has the following elements:

– the *localEntityIdentifier* that is assigned by Service Management to an SLE transfer service;

– the *authenticationLevel* associated with the entity identifier: ‘all’, ‘bind’, or ‘none’.

The *authenticationMethod* that is associated with the entity identifier when the authenticationLevel is ‘all’ or ‘bind’. There are currently no standard authentication methods defined for SLE transfer services. Selection and implementation of authentication methods is by bilateral agreement between the spaceflight mission and the Complex. Each authentication method has one or more *methodParameters*, the number and definition of which depend on the authentication method. Authentication methods may also incorporate external inputs—for example, time of creation of the data may be included to reduce the ability of an attacker to record and ‘replay’ a message back into the system at a later time.

NOTE – For the purposes of this description, the authentication method and its set of parameters are presented as configurable parameters, in anticipation of implementations that can support several methods. Implementations that support a single method may, of course, be hard-coded for that method, with no need to configure it on a service-instance-by-service-instance basis.

The peer entity authentication structure has the same elements as the local entity authentication structure, except that in place of the localEntityIdentifier element it has a *peerEntityIdentifier* that is assigned to an SLE transfer service entity authentication structure that is to be used by the peer system for service instances user the Service Agreement.

E4.3 SERVICE INSTANCE STRUCTURES

A service instance structure conceptually encapsulates the configuration information associated with a single service instance. It is created as part of the Service Package. The user and provider entities each have a copy of the structure for each service instance that they support. The service instance elements that are of interest to security processing are:

The *serviceInstanceIdentifier*, which identifies the transfer service instance. The serviceInstanceIdentifier is unique to a single Service Package.

The *initiatorIdentifier*, which contains the entity identifier that the initiator of the service instance must use. Whether the initiatorIdentifier contains a local or peer identifier depends on the role of the entity in the service instance (e.g., the entity that will initiate the service instance will use the localEntityIdentifier, whereas the entity that will respond to the BIND invocation will use a peerEntityIdentifier in this element).

The *responderIdentifier*, which contains the entityIdentifier that the responder of the service instance must use. Whether the responderIdentifier contains a local or peer identifier depends on the role of the entity in the service instance.

The *responderPortIdentifier*, which identifies the port on the responder system that is to be configured to accept the BIND invocation for the service instance.

Each user and provider entity has these security parameters configured by Service Management prior to the start of the service instance. How Service Management configures these parameters is outside the scope of this annex.

Figure E-1 illustrates the relationships between the entity authentication and service instance structures for an SLE entity configured to participate in a service instance.

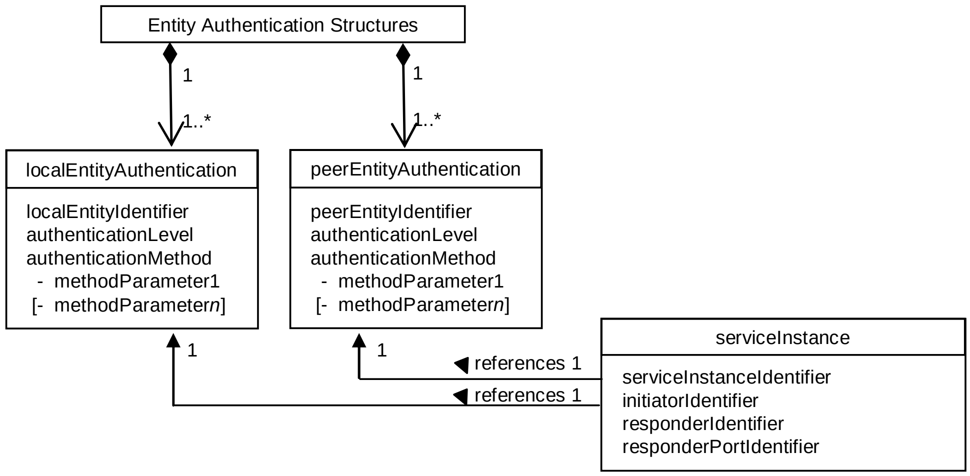


Figure E-1: Relationships among Entity Authentication and Service Instance Structures

E5 SECURITY MANAGEMENT INFORMATION IN SLE OPERATION INVOCATIONS AND RETURNS

E5.1 GENERAL

This subsection describes the use of the security information in generating and processing SLE transfer service operation invocations and returns.

– Subsection E5.2 presents the process flow for binding when the authentication level is ‘all’ or ‘bind’.

– Subsection E5.3 presents the process flow for other (non-BIND) SLE operations when the authentication level is ‘all’.

– Subsection E5.4 presents the process flow for binding when the authentication level is ‘none’.

– Subsection E5.5 presents the process flow for other (non-BIND) SLE operations when the authentication level is ‘none’ or ‘bind’.

All combinations of authentication and access control for SLE transfer service operations can be covered by some combination of these four process flows. The process flows are accompanied by illustrated scenarios. However, the illustrations depict only the ‘success path’ for the processing flows. That is, the error/exception cases are not illustrated (although they are addressed in the text).

E5.2 BIND INVOCATION/RETURN WITH ‘ALL’ OR ‘BIND’ AUTHENTICATION

The following steps describe the process flow for binding when the authentication level is ‘all’ or ‘bind’.

E5.2.1 Initiator invokes BIND:

Figure E-2 illustrates the BIND invocation generation process.

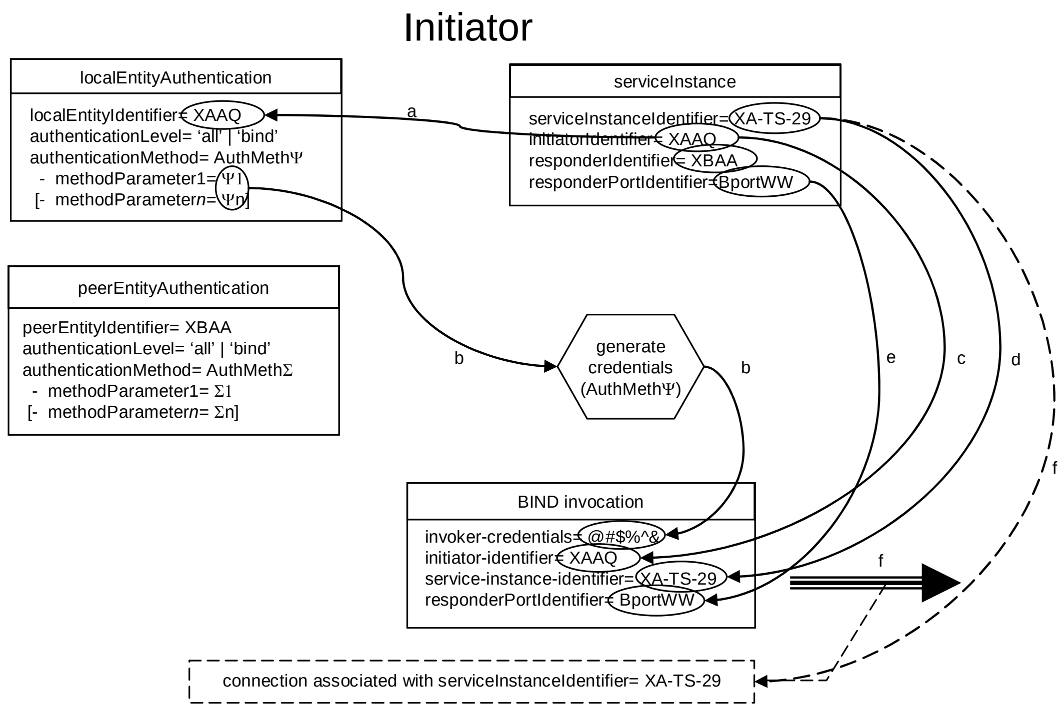


Figure E-2: BIND Invocation Generation for ‘All’ or ‘Bind’ Authentication Level

a) The Initiator uses the initiatorIdentifier element of the serviceInstance structure to identify the appropriate localEntityAuthentication structure (flow ‘a’ in the figure).

b) Because the authenticationLevel associated with the local entity identifier is either ‘all’ or ‘bind’, the Initiator generates the contents of `invoker-credentials` parameter of the BIND invocation using the designated authentication method and the parameter values stored for that entity identifier (b).

c) The Initiator sets the `initiator-identifier` parameter of the BIND invocation to the contents of initiatorIdentifier (c).

d) The Initiator sets the `service-instance-identifier` parameter of the BIND invocation to the contents of serviceInstanceIdentifier (d).

e) The Initiator sets the `responder-port-identifier` parameter of the BIND invocation to the contents of responderPortIdentifier (e).

f) The Initiator establishes a connection to the Responder, associates the serviceInstanceIdentifier with that connection, and invokes the BIND operation on the responder port named by the responderPortIdentifier (f).

NOTE – How the communication connection is established depends on the communication technology employed. In any case, this underlying connection is assumed to exist unless and until it is terminated as a result of an UNBIND, PEER-ABORT, or PROTOCOL-ABORT.

E5.2.2 Responder processes the BIND invocation:

Figure E-3 illustrates the BIND invocation authentication and access control process.

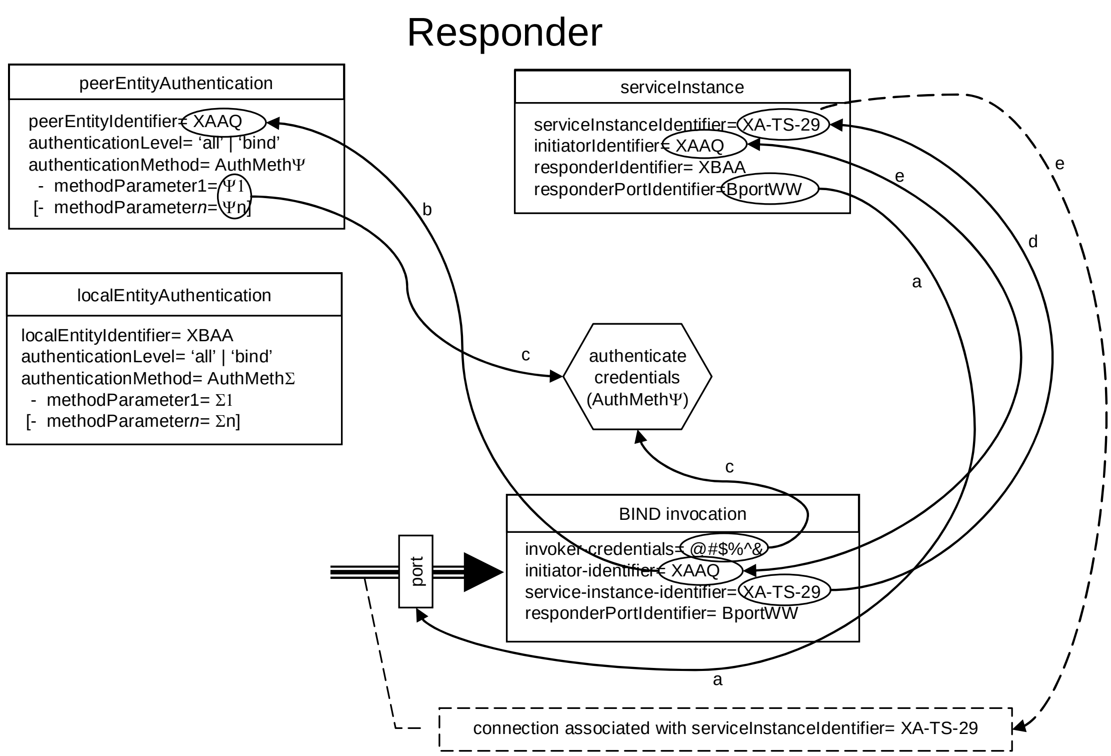


Figure E-3: BIND Invocation Authentication and Access Control for 'All' or 'Bind' Authentication Level

a) At the scheduled start time for the service instance identified by the serviceInstanceIdentifier, the Responder listens for a BIND invocation at the responder port identified in the responderPortIdentifier (a).

b) When the BIND invocation is received at the designated responder port, the Responder verifies that the `initiator-identifier` parameter of the BIND invocation matches a peerEntityIndentifier supported by the Responder (b).

   If the initiator-identifier parameter of the BIND invocation does not match any peerEntityIdentifier for the Responder, the Responder remains unbound, returns a negative result with `diagnostic` value 'access denied' and `performer-credentials` set to 'unused' (not shown in Figure E-3), and ceases processing the invocation.

c) The Responder uses the authenticationLevel value in the peerEntityAuthentication structure to determine that authentication is required, and uses the designated authentication method and parameters from the peerEntityAuthentication structure, and `invoker-credentials` from the BIND invocation, to authenticate the invocation (c).

   If authentication is not successful, the Responder ignores the invocation, remains unbound (not shown in Figure E-3), and ceases processing the invocation.

d) The Responder uses the `service-instance-identifier` parameter from the authenticated BIND invocation to identify the appropriate serviceInstance structure (d).

e) If the `initiator-identifier` matches the initiatorIdentifier in the serviceInstance structure, the Responder (1) grants access to the initiator, (2) attempts to execute the BIND on the service instance, (3) associates the service instance with the connection on which the invocation was received (e), and (4) and (if the BIND is successful) returns a positive result (below).

   If the `initiator-identifier` is not equal to the initiatorIdentifier in the serviceInstance structure, the Responder remains unbound, returns a negative result with `diagnostic` value 'service instance not accessible to this initiator' (not shown in Figure E-3), and ceases processing the invocation.

E5.2.3 Responder returns a positive result:

Figure E-4 illustrates the BIND invocation positive result generation process.

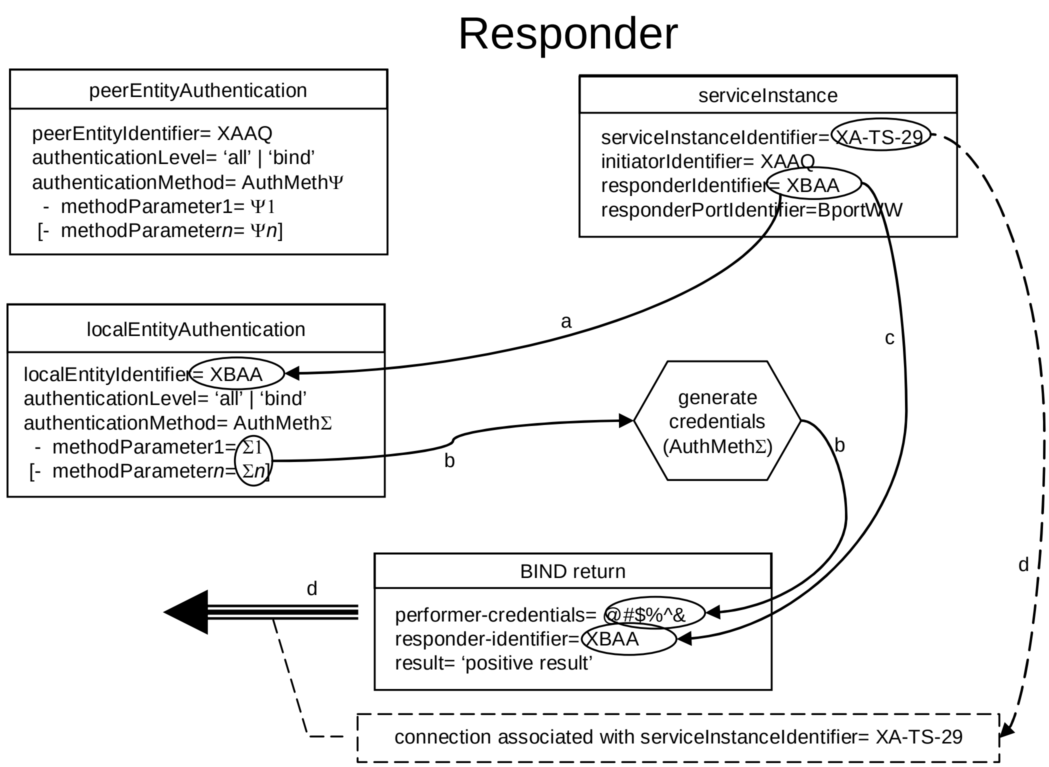


Figure E-4: Positive-Result BIND Return Generation for 'All' or 'Bind' Authentication Level

a) The Responder uses the responderIdentifier from the serviceInstance structure to identify the localEntityAuthentication structure to apply to the returns (a).

b) Because the authenticationLevel associated with the local entity identifier is either 'all' or 'bind', the Responder generates the contents of the `performer-credentials` parameter of the positive-result BIND return using the designated authentication method and the parameters associated with that responder identifier (b).

NOTE – It is technically possible for two paired SLE entities to use different authentication modes, although it seems that in almost all cases matched modes would be desirable.

c) The Responder sets the `responder-identifier` in the return equal to the responderIdentifier in the serviceInstance structure (c).

d) The Responder dispatches the return to the Initiator using the connection associated with the service instance (d).

NOTE – Consistent with the assumption stated in a previous Note, the underlying communication connection 'knows' how to get the BIND return to the Initiator.

E5.2.4 Initiator processes the BIND return:

If the Initiator does not receive an authenticatable BIND return within a specified time, the Initiator invokes a PEER-ABORT with `diagnostic` 'time-out'.

Positive result:

Figure E-5 illustrates the positive-result BIND return authentication and access control process.

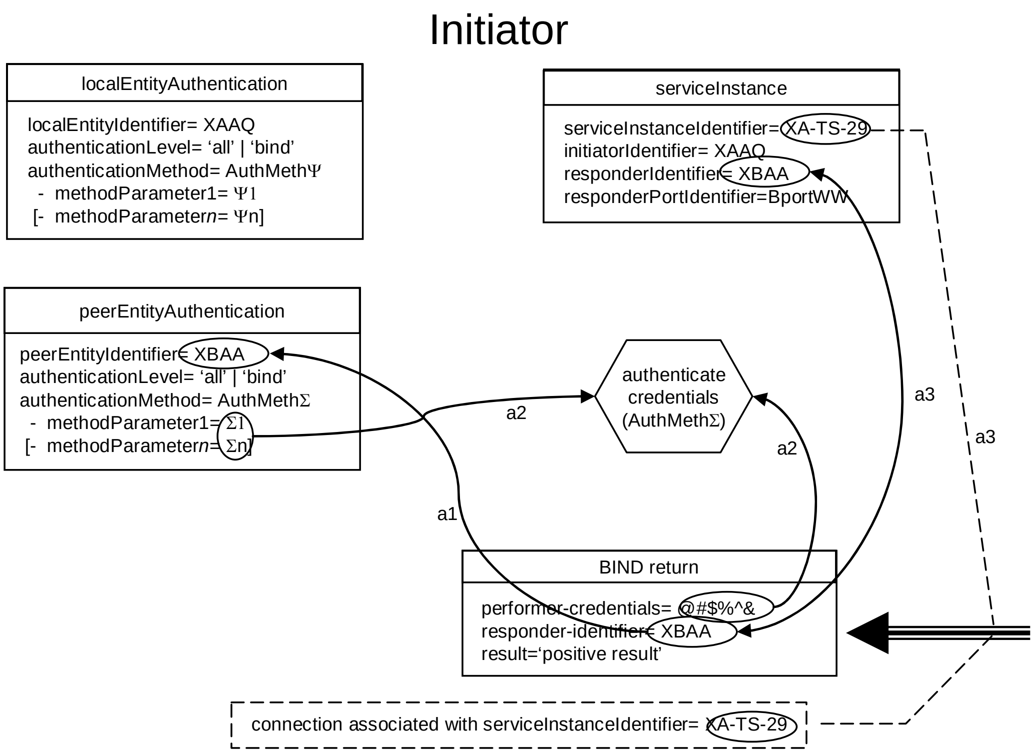


Figure E-5: Positive-Result BIND Return Authentication and Access Control for 'All' or 'Bind' Authentication Level

When a BIND return with a `result` parameter with value 'positive result' is received by the Initiator:

a) The Initiator verifies that the `responder-identifier` parameter of the BIND return matches a peerEntityIdentifier known to the Initiator (a1).

   If a match for the `responder-identifier` does <u>not</u> match any peerEntityIdentifier known to the Initiator, the Initiator invokes a PEER-ABORT with `diagnostic` 'access denied' (not shown in the figure) and ceases processing the return.

b) The Initiator uses the authenticationLevel in the associated peerEntityAuthentication structure to determine that authentication is required, and uses the designated authentication method and parameters specified in the peerEntityAuthentication structure and the `performer-credentials` from the BIND return, to authenticate the return (a2).

If authentication is not successful, the Initiator ignores and ceases processing the return (not shown in the figure).

c) The Initiator verifies that the `responder-identifier` matches the responderIdentifier in the serviceInstance structure associated with the connection over which the return was received (a3).

If the `responder-identifier` does not match the responderIdentifier in the serviceInstance structure, the Initiator invokes a PEER-ABORT with `diagnostic` 'unexpected responder ID' (not shown in the figure) and ceases processing the return.

d) If the service instance is in the ‘unbound’ state, then the state of the service instance becomes ‘ready’ (not shown in the figure).

If the service instance is in the state ‘ready’ or ‘active’, the Initiator invokes a PEER-ABORT with `diagnostic` ‘protocol error’ (not shown in the figure) and ceases processing the return.

Negative result with diagnostic ‘access denied’:

When a BIND return with a `result` parameter with value ‘negative result’ is received by the Initiator, if the value of the `diagnostic` parameter of the BIND return is 'access denied':

NOTE – In the case of a `diagnostic` of ‘access denied’, `performer-credentials` has been set to 'unused' and authentication is not attempted.

a) If the `responder-identifier` matches the responderIdentifier in the serviceInstance structure, the Initiator remains unbound and ceases waiting for a BIND return. The Initiator may attempt to re-invoke the BIND.

b) If the `responder-identifier` does not match the responderIdentifier in the serviceInstance structure, the Initiator invokes a PEER-ABORT with `diagnostic` 'unexpected responder ID'. The Initiator may attempt to re-invoke the BIND.

Negative result with diagnostic other than ‘access denied’:

When a BIND return with a `result` parameter with value ‘negative result’ is received by the Initiator, if the value of the `diagnostic` parameter of the BIND return is not 'access denied':

a) The Initiator verifies that the `responder-identifier` parameter of the BIND return matches a peerEntityIdentifier known to the Initiator.

If a match for the `responder-identifier` does not match any peerEntityIdentifier known to the Initiator, the Initiator invokes a PEER-ABORT with `diagnostic` 'access denied' and ceases processing the return.

b) Initiator uses the authenticationLevel in the associated peerEntityAuthentication structure to determine that authentication is required, and uses the designated authentication method and parameters specified in the peerEntityAuthentication structure and the `performer-credentials` from the BIND return to authenticate the return.

   If authentication is not successful, the Initiator ignores and ceases processing the return.

c) If the Initiator verifies that the `responder-identifier` matches the responderIdentifier in the serviceInstance structure, the Initiator remains unbound and ceases waiting for a BIND return.

   If the `responder-identifier` does not match the responderIdentifier in the serviceInstance structure, the Initiator invokes a PEER-ABORT with `diagnostic` 'unexpected responder ID' and ceases processing the return.

E5.3 XXX-OPERATION INVOCATION/RETURN WITH ‘ALL’ AUTHENTICATION

The following steps describe the process flow for performing an arbitrary (designated as ‘XXX’) SLE operation when the authentication level is ‘all’. Figure E-6 illustrates the BIND invocation generation portion of this process, and Figure E-4 illustrates the return portion.

E5.3.1 Invoker invokes XXX-operation:

Figure E-6 illustrates the XXX invocation generation process for an operation with authentication level of ‘all’.

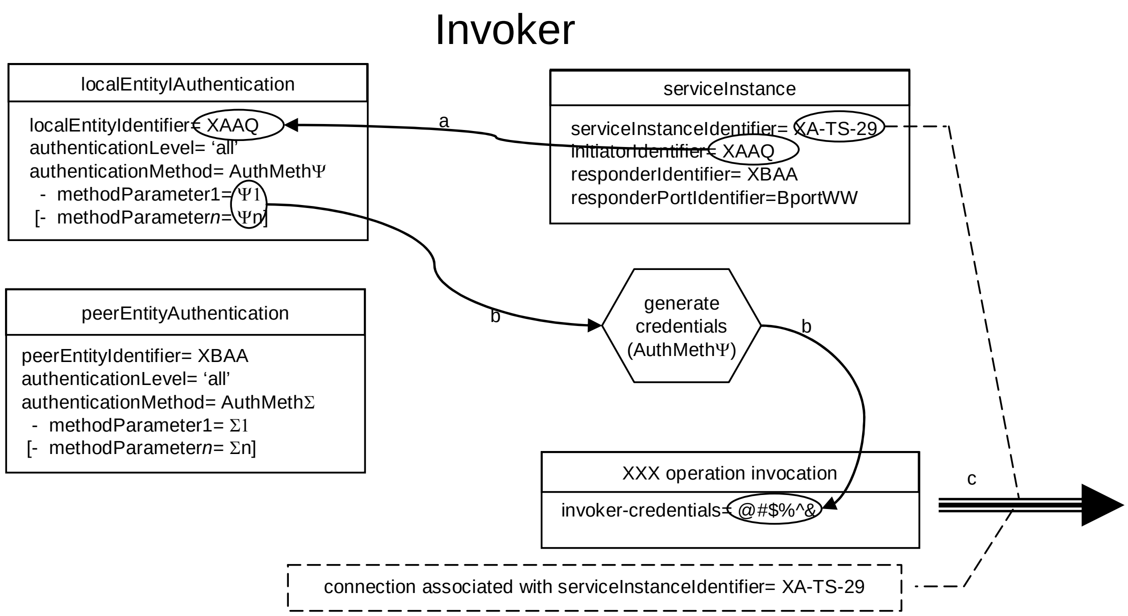


Figure E-6: XXX Operation Invocation Generation for ‘All’ Authentication Level

a) The Invoker uses the initiatorIdentifier element of the serviceInstance structure to identify the appropriate localEntityAuthentication structure (a).

b) Because the authenticationLevel associated with the local entity identifier is ‘all’, the Invoker generates the contents of `invoker-credentials` parameter of the BIND invocation using the designated authentication method and the parameter values stored for that entity identifier (b).

c) The Invoker invokes the XXX operation on the performer, using the connection associated with the service instance (c).

E5.3.2 Performer processes the XXX invocation:

Figure E-7 illustrates the XXX operation invocation authentication and access control process.

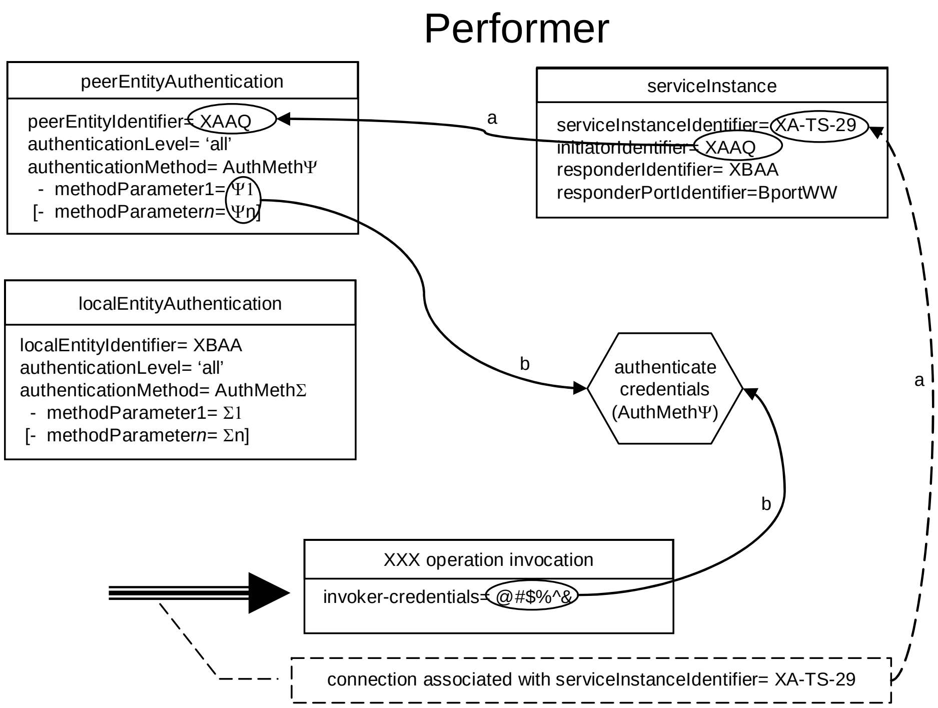


Figure E-7: XXX Operation Invocation Authentication and Access Control for 'All' Authentication Level

a) The Performer uses the initiatorIdentifier associated with the connection on which the invocation is received to select the appropriate peer entity authentication structure (a).

b) Because the authenticationLevel for the peer entity identifier is 'all', the Performer uses the designated authentication method and parameters to authenticate the `invoker-credentials` parameter of the invocation.

   If authentication is <u>not</u> successful, the Performer ignores and ceases processing the invocation.

c) The Performer executes the operation and (if successful) returns a positive result (below).

NOTE – Access privilege is assumed because of successful access verification during the BIND.

   If the operation is <u>not</u> successful, the Performer returns a negative result with the appropriate `diagnostic` value, and may abort the service instance (depending on the nature of the failure).

E5.3.3 Performer generates XXX operation return:

Figure E-8 illustrates the XXX operation return generation process.

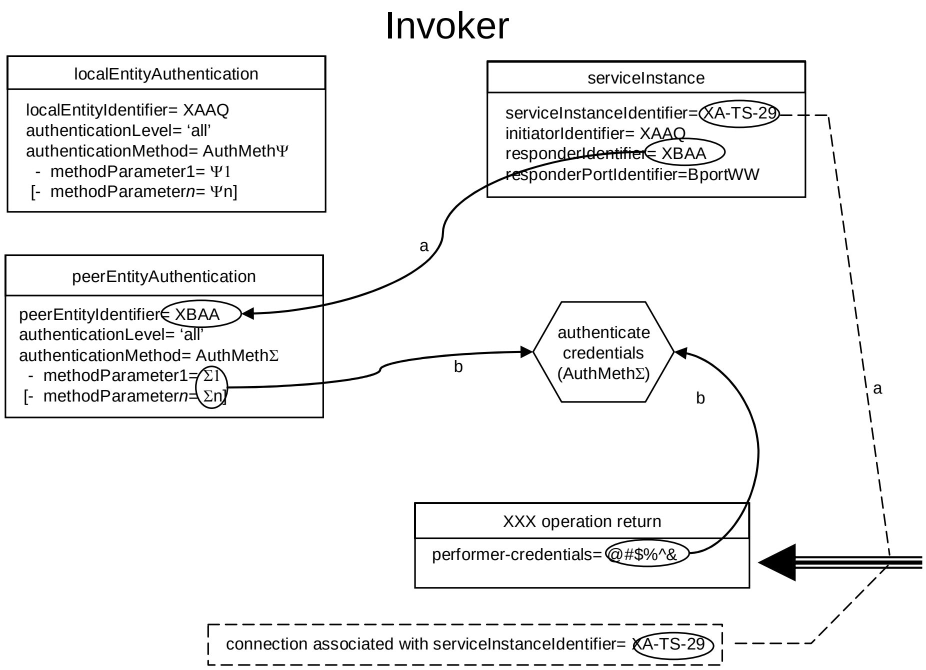


Figure E-8: XXX Operation Return Generation for ‘All’ Authentication Level

a) The Performer uses the responderIdentifier for the service instance to select the appropriate local entity authentication structure (a).

b) Because the authenticationLevel for the local entity identifier is ‘all’, the Performer uses the designated authentication method and parameters to generate the `performer-credentials` parameter of the return.

c) The Responder dispatches the XXX return on the communication connection associated with the service instance.

E5.3.4 Invoker processes the XXX operation return:

Figure E-9 illustrates the XXX operation return authentication and access control process.

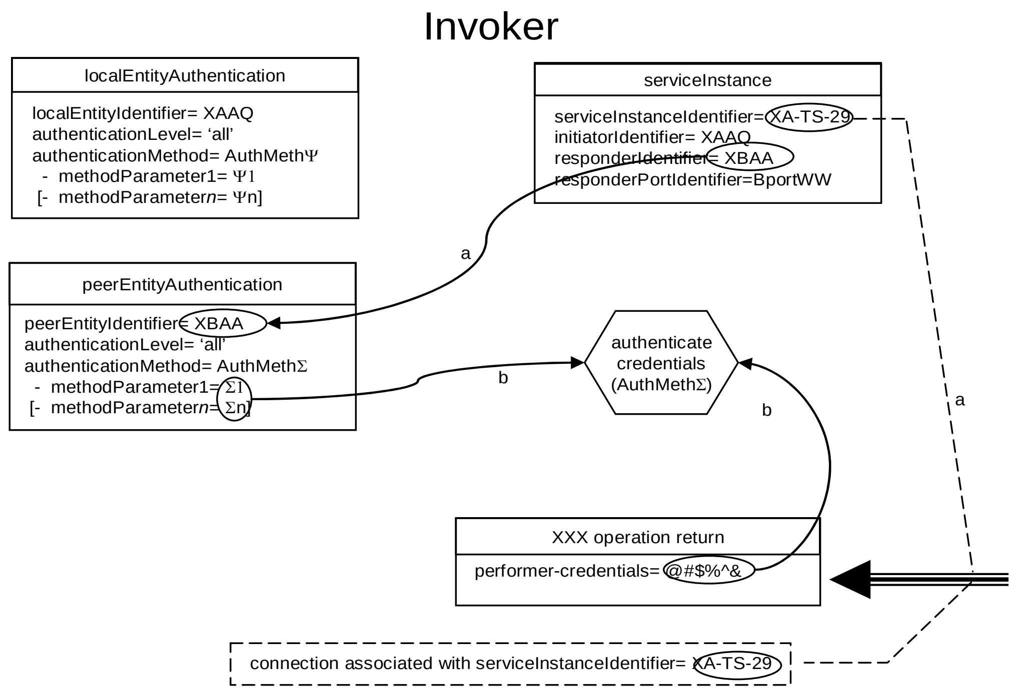


Figure E-9: XXX Operation Return Authentication and Access Control for 'All' Authentication Level

If the Invoker does not receive an authenticatable XXX operation return within a specified time, the Invoker invokes a PEER-ABORT with `diagnostic` 'time-out'.

a) The Invoker uses the responderIdentifier associated with the connection on which the return is received to select the appropriate peer entity authentication structure (a).

b) Because the authenticationLevel for the peer entity identifier is 'all', the Invoker uses the designated authentication method and parameters to authenticate the `performer-credentials` parameter of the return (b).

   If authentication is <u>not</u> successful, the Invoker ignores and ceases processing the return.

c) The Invoker accepts the positive or negative result conveyed by the return.

NOTE – Access privilege is assumed because of successful access verification during the BIND.

E5.4 BIND INVOCATION AND RETURN WITH NO AUTHENTICATION

The following steps describe the process flow for binding when the authentication level is 'none'.

E5.4.1 Initiator invokes BIND:

Figure E-10 illustrates the BIND invocation generation process.

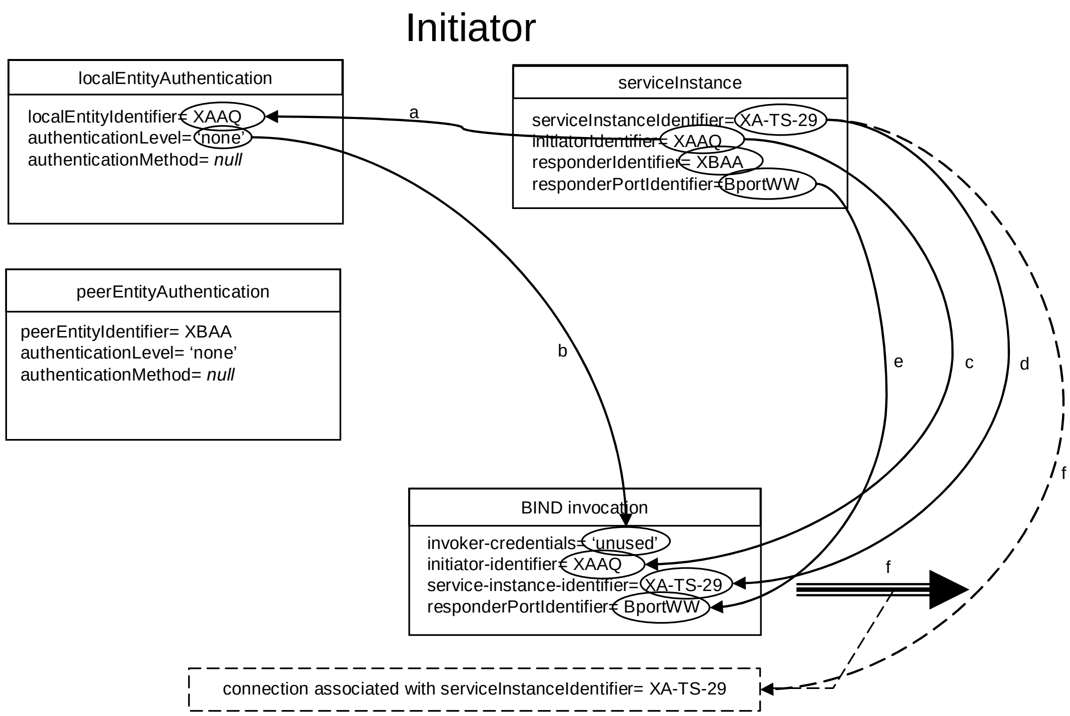


Figure E-10: BIND Invocation Generation for 'None' Authentication Level

a) The Initiator uses the initiatorIdentifier element of the serviceInstance structure to identify the appropriate localEntityAuthentication structure (a).

b) Because the authenticationLevel associated with the local entity identifier is 'none', the contents of `invoker-credentials` parameter of the BIND invocation is set to 'unused' (b).

c) The Initiator sets the `initiator-identifier` parameter of the BIND invocation to the contents of initiatorIdentifier (c).

d) The Initiator sets the `service-instance-identifier` parameter of the BIND invocation to the contents of serviceInstanceIdentifier (d).

e) The Initiator sets the `responder-port-identifier` parameter of the BIND invocation to the contents of responderPortIdentifier (e).

f) The Initiator establishes a connection to the Responder, associates the serviceInstanceIdentifier with that connection, and invokes the BIND operation on the responder port named by the responderPortIdentifier (f).

NOTE – How the communication connection is established depends on the communication technology employed. In any case, this underlying connection is assumed to exist unless and until it is terminated as a result of an UNBIND, PEER-ABORT, or PROTOCOL-ABORT.

E5.4.2 Responder processes the BIND invocation:

Figure E-11 illustrates the BIND invocation authentication and access control process.

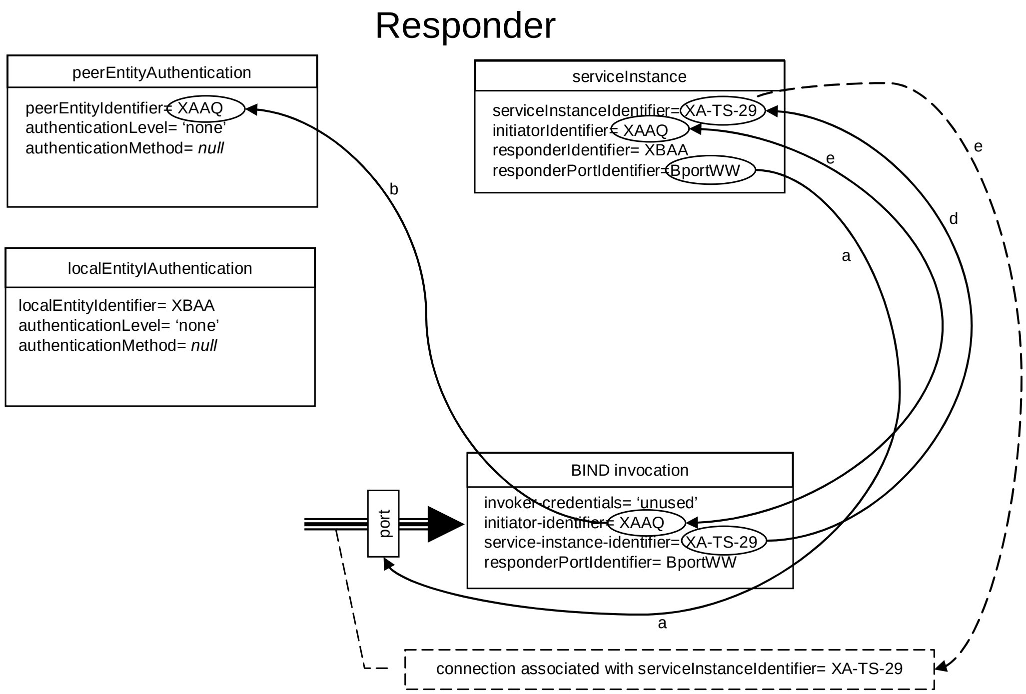


Figure E-11: BIND Invocation Authentication and Access Control for 'None' Authentication Level

a) At the scheduled start time for the service instance identified by the serviceInstanceIdentifier, the Responder listens for a BIND invocation at the responder port identified in the responderPortIdentifier (a).

b) When the BIND invocation is received at the designated responder port, the Responder verifies that the `initiator-identifier` parameter of the BIND invocation matches a peerEntityIndentifier supported by the Responder (b).

   If the initiator-identifier parameter of the BIND invocation does not match any peerEntityIdentifier for the Responder, the Responder remains unbound, returns a negative result with `diagnostic` value 'access denied' and `performer-credentials` set to 'unused' (not shown in the figure), and ceases processing the invocation.

c) The Responder uses the authenticationLevel value ('none') in the peerEntityAuthentication structure to determine that no authentication is required.

NOTE – If a match for the `initiator-identifier` exists in the peerEntityIdentifier of a peer entity authentication structure that has an authenticationLevel that is 'all' or 'bind', the Responder attempts to authenticate the credentials using the authenticationMethod associated with peer entity authentication structure. Since the `invoker-credentials` parameter has the value 'unused', the authentication fails and the Responder ignores the invocation and remains unbound. Eventually, the Initiator will time-out waiting for a valid return and PEER-ABORT with `diagnostic` value 'time out'.

d) The Responder uses the `service-instance-identifier` parameter from the BIND invocation to identify the appropriate serviceInstance structure (d).

e) If the `initiator-identifier` matches the initiatorIdentifier in the serviceInstance structure, the Responder (1) grants access to the initiator, (2) attempts to execute the BIND on the service instance, (3) associates the service instance with the connection on which the invocation was received (e), and (4) and (if the BIND is successful) returns a positive result (below).

   If the `initiator-identifier` is not equal to the initiatorIdentifier in the serviceInstance structure, the Responder remains unbound, returns a negative result with `diagnostic` value 'service instance not accessible to this initiator' (not shown in the figure), and ceases processing the invocation.

E5.4.3 Responder returns a positive result:

Figure E-12 illustrates the BIND invocation positive result generation process.

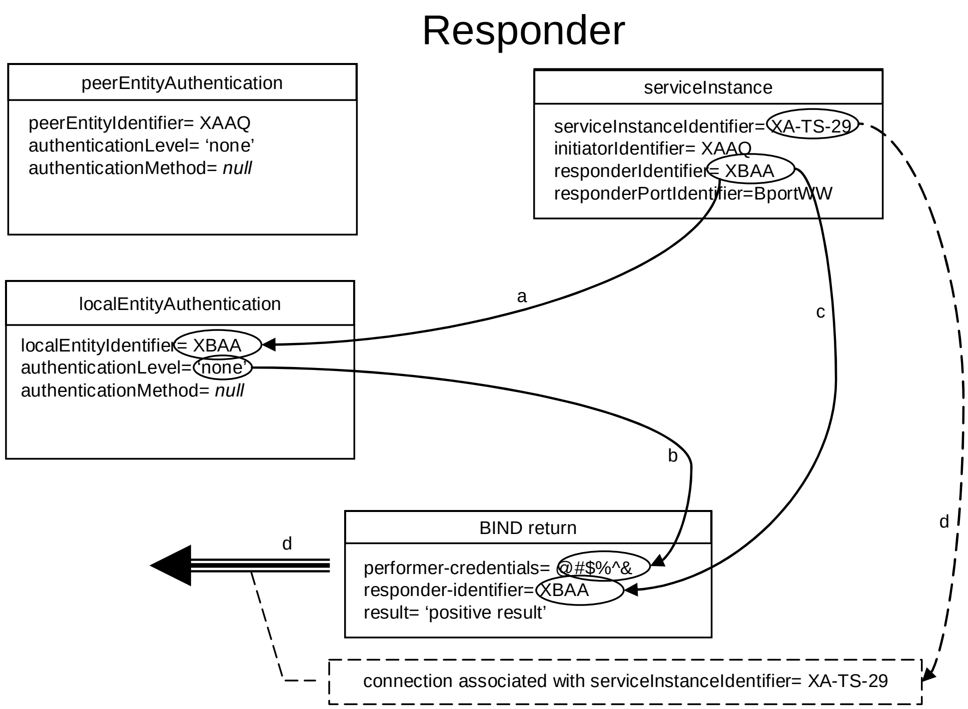


Figure E-12: Positive-Result BIND Return Generation for 'None' Authentication Level

a) The Responder uses the responderIdentifier from the serviceInstance structure to identify the localEntityAuthentication structure to apply to the returns (a).

b) Because the authenticationLevel associated with the local entity identifier is either 'none', the Responder sets the contents of the `performer-credentials` parameter of the positive-result BIND return to 'unused' (b).

NOTE – It is technically possible for two paired SLE entities to use different authentication levels for invocations and returns, but the assumption in this scenario is that the authentication levels match—in this case, both are 'none'.

c) The Responder sets the responder-identifier in the return set equal to the responderIdentifier in the serviceInstance structure (c).

d) The Responder dispatches the return to the Initiator using the connection associated with the service instance (d).

NOTE – Consistent with the assumption stated in a previous Note, the underlying communication connection 'knows' how to get the BIND return to the Initiator.

E5.4.4 Initiator processes the BIND return:

If the Initiator does not receive a BIND return within a specified time, the Initiator invokes a PEER-ABORT with `diagnostic` 'time-out'.

Positive result:

Figure E-13 illustrates the positive-result BIND return authentication and access control process.

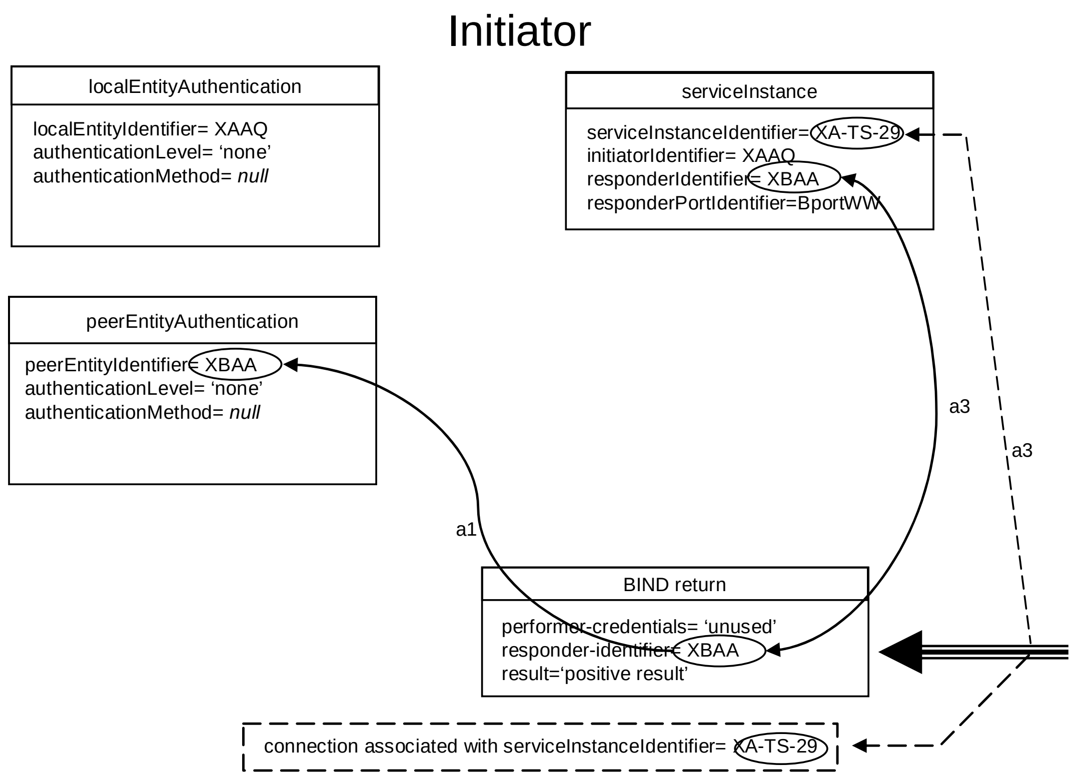


Figure E-13: Positive-Result BIND Return Authentication and Access Control for 'None' Authentication Level

When a BIND return with a `result` parameter with value 'positive result' is received by the Initiator:

a) The Initiator verifies that the `responder-identifier` parameter of the BIND return matches a peerEntityIdentifier known to the Initiator (a1).

   If a match for the `responder-identifier` does <u>not</u> match any peerEntityIdentifier known to the Initiator, the Initiator invokes a PEER-ABORT with `diagnostic` 'access denied' (not shown in the figure) and ceases processing the return.

b) The Initiator uses the authenticationLevel in the associated peerEntityAuthentication structure to determine that no authentication is required.

NOTE – If a match for the `responder-identifier` exists in the peerEntityIdentifier of a peer entity authentication structure that has an authenticationLevel that is ‘all’ or ‘bind’, the Initiator attempts to authenticate the credentials using the authenticationMethod associated with peer entity authentication structure. Since the `performer-credentials` parameter has the value 'unused', the authentication fails and the Initiator ignores the invocation and remains unbound. Eventually, the Initiator will time-out waiting for a valid return and PEER-ABORT with `diagnostic` value 'time out'.

c) The Initiator verifies that the `responder-identifier` matches the responderIdentifier in the serviceInstance structure associated with the connection over which the return was received (a3).

   If the `responder-identifier` does not match the responderIdentifier in the serviceInstance structure, the Initiator invokes a PEER-ABORT with `diagnostic` 'unexpected responder ID' and ceases processing the return (not shown in the figure).

d) If the service instance is in the ‘unbound’ state, then the state of the service instance becomes ‘ready’ (not shown in the figure).

   If the service instance is in the state ‘ready’ or ‘active’, the Initiator invokes a PEER-ABORT with `diagnostic` ‘protocol error’ and ceases processing the return (not shown in the figure).

Negative result with diagnostic ‘access denied’:

When a BIND return with a `result` parameter with value ‘negative result’ is received by the Initiator, if the value of the `diagnostic` parameter of the BIND return is 'access denied':

NOTE – In the case of a diagnostic of ‘access denied’, performer-credentials has been set to 'unused' and authentication is not attempted.

a) If the `responder-identifier` matches the responderIdentifier in the serviceInstance structure, the Initiator remains unbound and ceases waiting for a BIND return. The Initiator may attempt to re-invoke the BIND.

b) If the `responder-identifier` does not match the responderIdentifier in the serviceInstance structure, the Initiator invokes a PEER-ABORT with `diagnostic` 'unexpected responder ID'. The Initiator may attempt to re-invoke the BIND.

Negative result with diagnostic other than ‘access denied’:

When a BIND return with a `result` parameter with value ‘negative result’ is received by the Initiator, if the value of the `diagnostic` parameter of the BIND return is not 'access denied':

a) The Initiator verifies that the `responder-identifier` parameter of the BIND return matches a peerEntityIdentifier known to the Initiator.

   If a match for the `responder-identifier` does not match any peerEntityIdentifier known to the Initiator, the Initiator invokes a PEER-ABORT with `diagnostic` 'access denied' and ceases processing the return.

b) The Initiator uses the authenticationLevel in the associated peerEntityAuthentication structure to determine that no authentication is required.

NOTE – If a match for the `responder-identifier` exists in the peerEntityIdentifier of a peer entity authentication structure that has an authenticationLevel that is ‘all’ or ‘bind’, the Initiator attempts to authenticate the credentials using the authenticationMethod associated with peer entity authentication structure. Since the `performer-credentials` parameter has the value 'unused', the authentication fails and the Initiator ignores the invocation and remains unbound. Eventually, the Initiator will time-out waiting for a valid return and PEER-ABORT with `diagnostic` value 'time out'.

c) If the Initiator verifies that the `responder-identifier` matches the responderIdentifier in the serviceInstance structure, the Initiator remains unbound and ceases waiting for a BIND return.

   If the `responder-identifier` does not match the responderIdentifier in the serviceInstance structure, the Initiator invokes a PEER-ABORT with `diagnostic` 'unexpected responder ID'.

E5.5 XXX-OPERATION INVOCATION/RETURN WITH ‘NONE’ OR ‘BIND’ AUTHENTICATION

The following steps describe the process flow for performing an arbitrary (designated as ‘XXX’) SLE operation when the authentication level is 'none' or 'bind'.

E5.5.1 Invoker invokes XXX-operation:

Figure E-14 illustrates the XXX invocation generation process for an operation with authentication level of ‘none’ or ‘bind’.

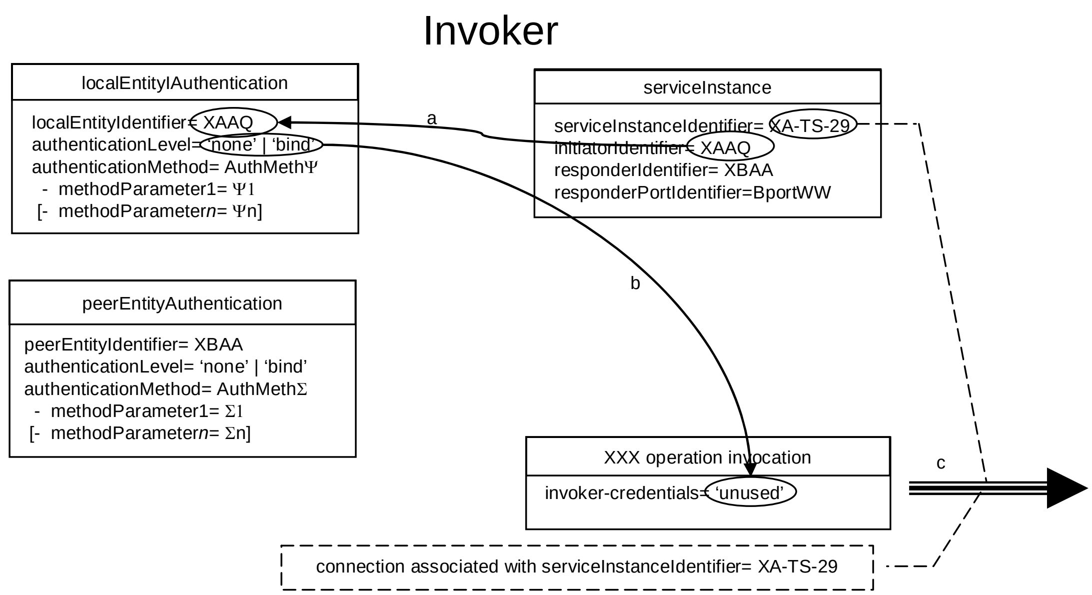


Figure E-14: XXX Operation Invocation Generation for 'None' or 'Bind' Authentication Level

a) The Invoker uses the initiatorIdentifier element of the serviceInstance structure to identify the appropriate localEntityAuthentication structure (a).

b) Because the authenticationLevel associated with the local entity identifier is 'none' or bind', the Invoker sets the contents of `invoker-credentials` parameter of the XXX operation invocation to 'unused' (b).

c) The Invoker invokes the XXX operation on the performer, using the connection bound to the service instance (c).

E5.5.2 Performer processes the XXX invocation:

Figure E-15 illustrates the XXX operation invocation authentication and access control process.

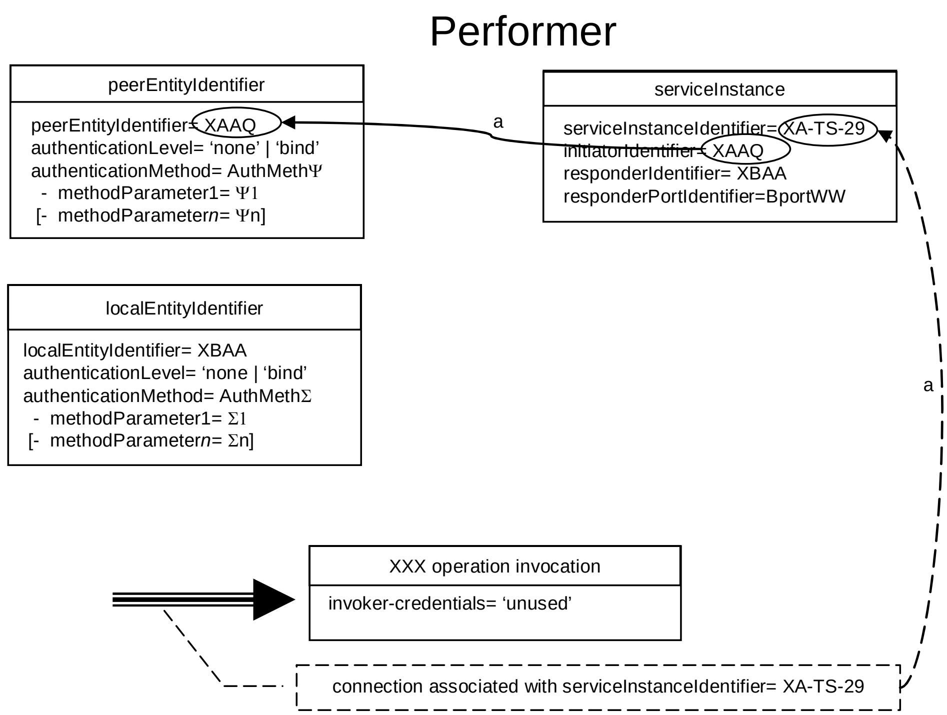


Figure E-15: XXX Operation Invocation Authentication and Access Control for 'None' or 'Bind' Authentication Level

a) The Performer uses the initiatorIdentifier associated with the connection on which the invocation is received to select the appropriate peer entity authentication structure (a).

b) Because the authenticationLevel for the peer entity identifier is 'none' or 'bind', the Performer does not authenticate the invocation. Access privilege is assumed because of successful access verification during the BIND.

NOTE – If a match for the `initiator-identifier` exists in the peerEntityIdentifier of a peer entity authentication structure that has an authenticationLevel that is 'all', the Responder attempts to authenticate the credentials using the authenticationMethod associated with peer entity authentication structure. Since the `invoker-credentials` parameter has the value 'unused', the authentication fails and the Responder ignores the invocation and remains unbound. Eventually, the Initiator will time-out waiting for a valid return and PEER-ABORT with `diagnostic` value 'time out'.

c) The Performer executes the operation and (if successful) returns a positive result (below).

If the operation is not successful, the Performer returns a negative result with the appropriate `diagnostic` value, and may abort the service instance (depending on the nature of the failure).

E5.5.3 Performer generates XXX operation return:

Figure E-16 illustrates the XXX operation return generation process.

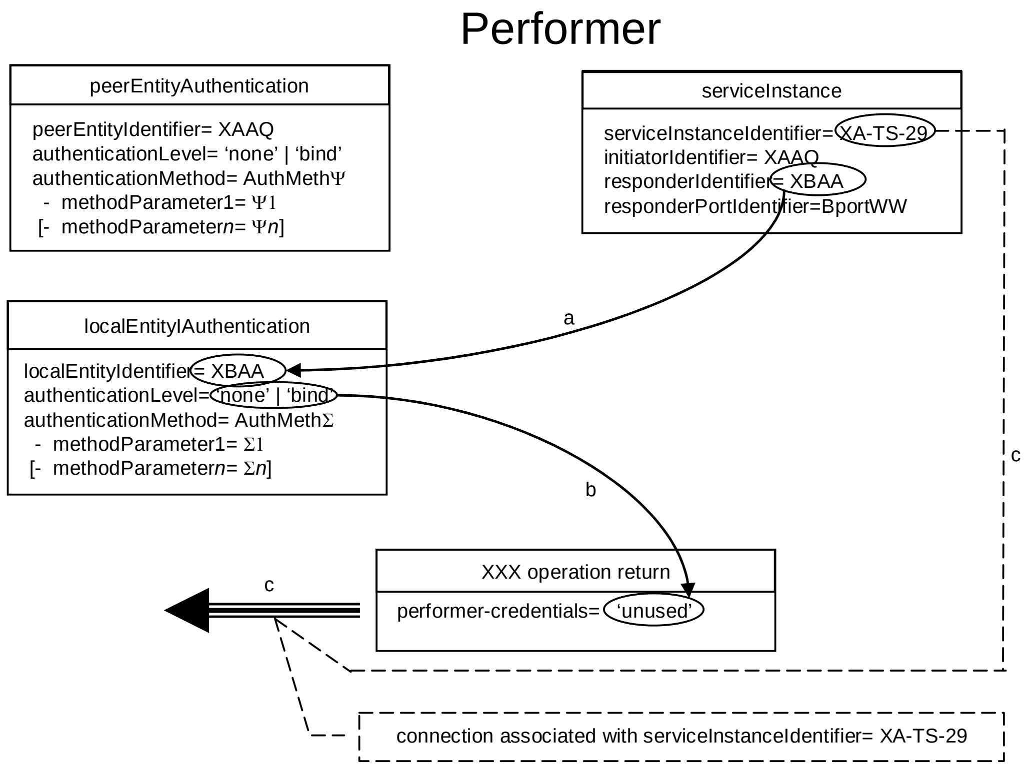


Figure E-16: XXX Operation Return Generation for 'None' or 'Bind' Authentication Level

a) The Performer uses the responderIdentifier for the service instance to select the appropriate local entity authentication structure (a).

b) Because the authenticationLevel for the local entity identifier is 'none' or 'bind', the Performer sets the `performer-credentials` parameter of the return to 'unused'.

c) The Responder dispatches the XXX return on the communication connection associated with the service instance.

E5.5.4 Invoker processes the XXX operation return:

Figure E-17 illustrates the positive-result XXX operation return authentication and access control process.

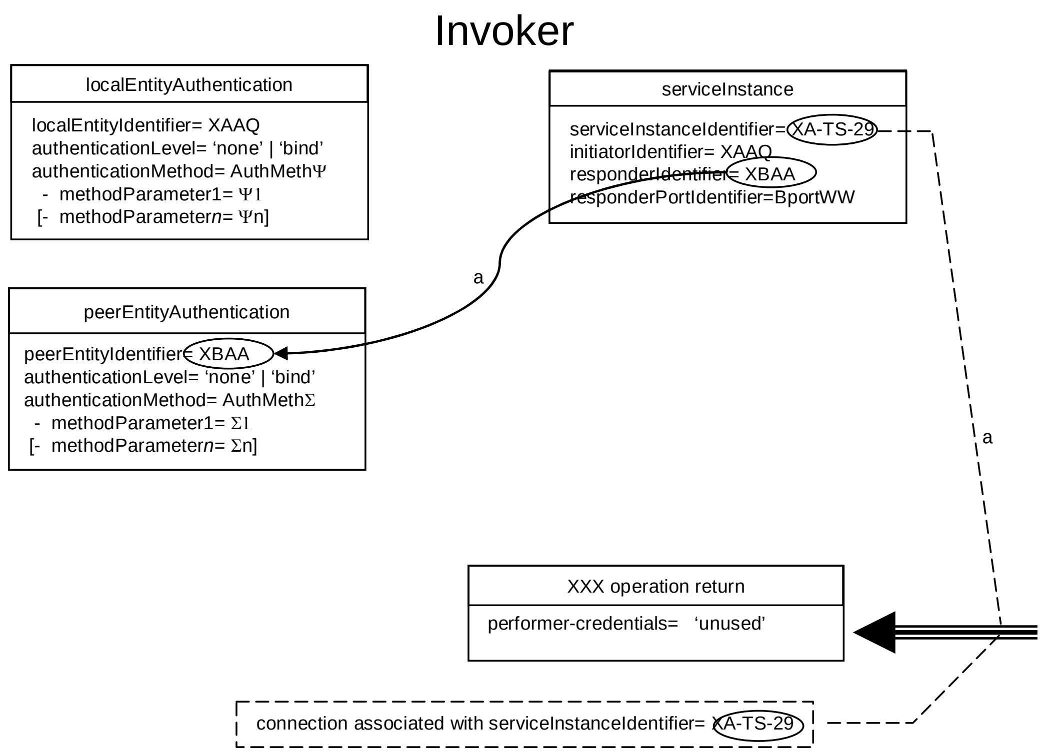


Figure E-17: XXX Operation Return Authentication and Access Control for 'None' or 'Bind' Authentication Level

If the Invoker does not receive a XXX operation return within a specified time, the Invoker invokes a PEER-ABORT with `diagnostic` 'time-out'.

a) The Invoker uses the responderIdentifier associated with the connection on which the return is received to select the appropriate peer entity authentication structure (a).

b) Because the authenticationLevel for the peer entity identifier is 'none' or 'bind', the Invoker does not authenticate the `performer-credentials` parameter of the return.

NOTE – If a match for the `responder-identifier` exists in the peerEntityIdentifier of a peer entity authentication structure that has an authenticationLevel that is 'all', the Initiator attempts to authenticate the credentials using the authenticationMethod associated with peer entity authentication structure. Since the `performer-credentials` parameter has the value 'unused', the authentication fails and the Initiator ignores the invocation and remains unbound. Eventually, the Initiator will time-out waiting for a valid return and PEER-ABORT with `diagnostic` value 'time out'.

c) The Invoker accepts the positive or negative result conveyed by the return.

NOTE – Access privilege is assumed because of successful access verification during the BIND (b).