The Consultative Committee for Space Data Systems

Draft Recommendation for
Space Data System Standards

SPACE LINK EXTENSION—RETURN CHANNEL FRAMES SERVICE SPECIFICATION

DRAFT RECOMMENDED STANDARD

CCSDS 911.2-P-1.1

PINK SHEETS

December 2008

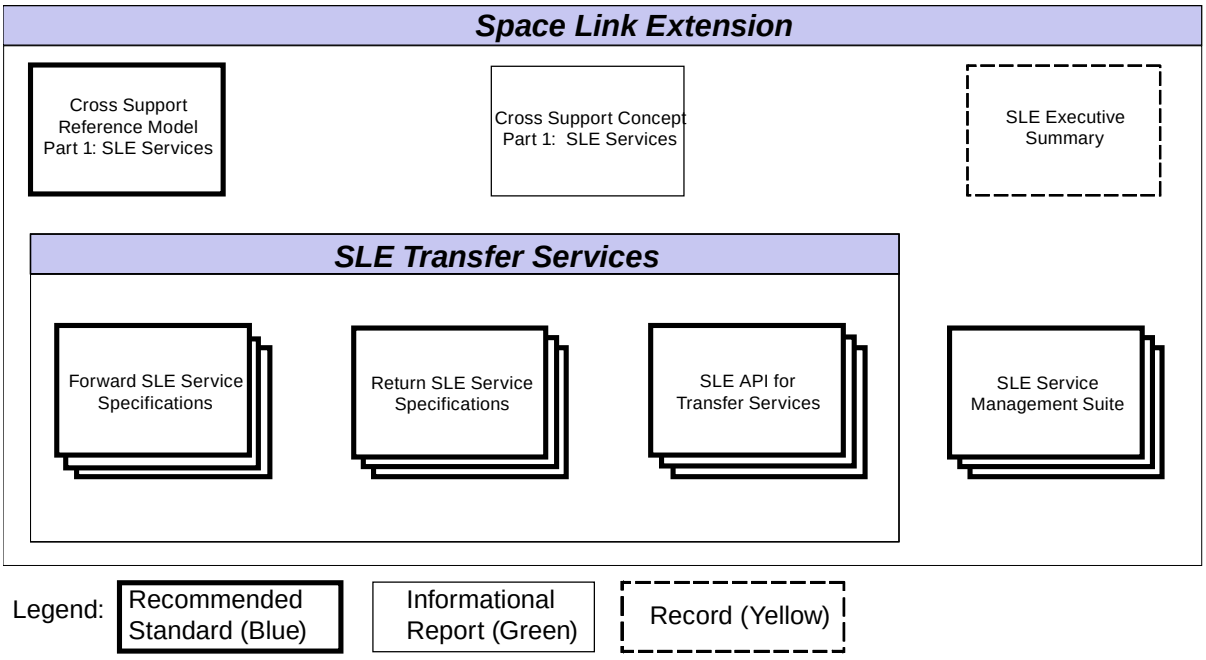

Figure 1-1: SLE Services Documentation

a) *Cross Support Concept—Part 1: Space Link Extension Services* (reference [E2]): a Report introducing the concepts of cross support and the SLE services;

b) *Cross Support Reference Model—Part 1: Space Link Extension Services* (reference [1]): a Recommended Standard that defines the framework and terminology for the specification of SLE services;

c) *SLE Return Service Specifications*: a set of Recommended Standards that will provide specification of all return link SLE services (this Recommended Standard is one of the specifications in that set);

d) *SLE Forward Service Specifications*: a set of Recommended Standards that will provide specification of all forward link SLE services;

e) *SLE API for Transfer Services Specifications*: a set of Recommended Standards that provide specifications of an Application Program Interface and a mapping to TCP/IP as underlying communications service for SLE services;

f) *SLE Service Management Specifications*: a set of Recommended Standards that establish the basis of SLE service management.

n) Return All Frames channel (RAF channel);

o) Return All Frames service (RAF service);

p) Return Master Channel Frame Service (MC service)

q) Return Virtual Channel Frame Service (VC Frame service)

r) service agreement;

s) service provider (provider);

t) service user (user);

u) SLE Complex;

v) SLE Complex Management;

w) SLE data channel;

x) SLE Functional Group (SLE-FG);

y) SLE Protocol Data Unit (SLE-PDU);

z) SLE Service Data Unit (SLE-SDU);

aa) SLE service package;

bb) ~~SLE System;~~

bb) SLE transfer service instance;

cc) SLE transfer service production;

dd) SLE transfer service provision;

ee) SLE Utilization Management;

ff) space link;

gg) space link data channel;

hh) Space Link Data Unit (SL-DU);

ii) space link session.

1.6.1.7 Additional Definitions

1.6.1.7.1 Association

An association is a cooperative relationship between an SLE service-providing application entity and an SLE service-using application entity. An association is formed by the

NOTE – The definitions of frame version number given in references [3] and [4] are equivalent. If a CCSDS-compatible telemetry frame is known to contain no errors, the frame version number enables one to distinguish between a transfer frame and an AOS transfer frame.

1.6.1.7.7 Initiator

The initiator is the object that issues the request to bind to another object (the responder).

NOTE – In other words, the initiator is always the invoker of the request to bind to another object. Therefore, in the context of the request to bind, the terms 'initiator' and 'invoker' refer to the same object and are synonyms.

1.6.1.7.8 Invocation

The invocation of an operation is the making of a request by an object (the invoker) to another object (the performer) to carry out the operation.

1.6.1.7.9 Master Channel

The sequence of all telemetry frames with the same Transfer Frame Version Number (TFVN) and the same SCID on the same physical channel constitutes a master channel.

NOTE – Depending on the TFVN, the definition of SCID is as given in either reference [3] or reference [4].

1.6.1.7.10 Parameter

A parameter of an operation is data that may accompany the operation's invocation or return.

NOTE – The term parameter is also used to refer to mission-dependent configuration information used in the production or provision of the service.

1.6.1.7.11 Performance

The performance of an operation is the carrying out of the operation by an object (the performer).

1.6.1.7.12 Port Identifier

A port identifier identifies a source or a destination in a communications system.

NOTE – See 2.6.4.5 for more information.

1.6.1.7.13 Responder

The responder is the object that receives a request to bind and completes the binding (if possible) with the initiator in order for a service association to exist between the two objects.

NOTE – In other words, the responder is always the performer of the binding. Therefore, in the context of binding, the terms 'responder' and 'performer' refer to the same object and are synonyms.

1.6.1.7.14 Return

The return of an operation is a report, from the performer to the invoker, of the outcome of the performance of the operation.

1.6.1.7.15 Service Instance Provision Period

A 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 scheduled to be provided.

NOTE – Reaching of the beginning of this period constitutes the event 'start of service instance provision period' (see 4.2.2).

1.6.1.7.16 Spacecraft Identifier

The spacecraft identifier (SCID) of a telemetry frame is as defined in reference [3] if the frame is a TM Transfer Frame or as defined in reference [4] if the frame is an AOS Transfer Frame.

1.6.1.7.17 Telemetry Frame

A telemetry frame is a TM Transfer Frame (as defined in reference [3]) or an AOS Transfer Frame (as defined in reference [4]). In case a distinction of the frame versions is necessary, the full term as per references [3] or [4] is used.

1.6.1.7.18 Transfer Frame Version Number

The Transfer Frame Version Number (TFVN) is either the TFVN as defined in reference [3] or the TFVN as defined in reference [4].

NOTE – The definitions of TFVN given in references [3] and [4] are equivalent. If a CCSDS-compatible telemetry frame is known to contain no errors, the TFVN enables one to distinguish between a TM Transfer Frame and an AOS Transfer Frame.

object closer to the space link is considered to be the provider of the service, and the object further from the space link is considered to be the user.

2.2.6 OPERATION

An operation is a procedure or task that one object (the invoker) can request of another (the performer) through a bound port pair. The terms invoker and performer are used to describe the interaction between two objects as the operations that constitute the service occur. One object invokes an operation that is performed by the other. For most services, each object invokes some operations and performs others.

2.3 SERVICE MANAGEMENT

SLE service management determines the number and schedule of RCF service instances to be provided, the resources required to enable those service instances, and the initial configuration of all service instances and their supporting resources. SLE service management is the subject of separate CCSDS Recommended Standards.

The SLE Reference Model (reference [1]) distinguishes between service provision and service production:

a) service provision makes available to the user the operations necessary to obtain the service;

b) service production transforms a space link channel to an RCF channel, possibly using the service provision and production of another SLE provider or the equivalent capability.

Certain configuration parameters are associated with provision of RCF services, while others are associated with production. Changes to RCF provision configuration parameters (e.g., `requested-global-VCID`) affect only a single service instance; the values of such parameters are initialized by service management when the service instance is created, but may be modified subsequently by the user through RCF service operations specified in this Recommended Standard. Changes to RCF production configuration parameters (e.g., bit rate, frame length, coding type) potentially affect multiple service instances or potentially impact SLE Complex resources; consequently, those parameters may be modified only through service management.

RCF service may be user-initiated (i.e., the user invokes the bind operation) or provider-initiated (i.e., the provider invokes the bind operation). A particular instance of RCF service shall support either user initiation or provider initiation but not both. The form of initiation that applies to a particular service instance is set by service management.

The SLE Reference Model defines two delivery modes: online delivery mode and offline delivery mode. Online delivery mode indicates that the provision of service is generally coincident in time with the space link session, whereas offline delivery mode indicates that the telemetry frames acquired during a space link session are provided to the user some time

involved in the exchange of SLE-PDUs are generally minor. However, the way an association is established (i.e., the binding) tends to vary significantly depending on the communications technology in use. Nonetheless, the RCF-BIND and RCF-UNBIND operations as specified in this document are intended to be 'technology neutral'. This neutrality is achieved as described in the following subsections.

For purposes of the communications mapping, the endpoints of an SLE association are identified by port identifiers, namely, an 'initiator port identifier' and a 'responder port identifier'. The port identifiers represent all the technology-specific addressing information needed to establish communications between the user and provider and to route SLE-PDUs between them. The initiator port identifier identifies the endpoint that will invoke the RCF-BIND operation (initiator). The responder port identifier identifies the endpoint that will perform the RCF-BIND operation (responder). Generally speaking, the information represented by a port identifier consists of:

a) information needed in order to route data between two real systems over a communications channel or network; and

b) information needed in order to route data within a real system to a particular application entity.

For example, the information represented by a port identifier might be the combination of an Internet Protocol (IP) network address and a Transmission Control Protocol (TCP) port number or the combination of an OSI network address and an associated set of Service Access Points (SAPs).

The exact relationship between SLE port identifiers and communications ports provided by the underlying communications service must be specified by the mapping of the RCF service to the underlying communications service. If the underlying communications service is connection-oriented, then the mapping may specify a one-to-one relationship between SLE associations and communications connections; however, that is not required. For example, two SLE associations involving the same pair of SLE endpoints may share a single connection. In that case, it is the responsibility of the mapping of the RCF service to the underlying communications service to specify how the SLE-PDUs of one association are distinguished from the SLE-PDUs of the other association.

One possible mapping of the SLE transfer service to the TCP/IP communications service is specified in [E6]. As part of this mapping, also issues such as sizing of TCP buffers in accordance with the bandwidth-delay product of the communication link and ways to manage relative priority of transfer services concurrently using the same connectivity are to be addressed.

In order for an SLE association to be established, SLE Complex Management and SLE Utilization Management must agree beforehand on the responder port identifier for the association. The responder needs the information represented by the responder port identifier to ensure that resources are allocated to recognize and respond to an RCF-BIND

3.1.5 AUTHENTICATION

NOTE – Requirements for security depend on the application and the ~~SLE system~~ environment of the SLE Complexes and the MDOS (e.g., whether closed or public networks are used or if access is only from physically restricted areas). In many environments, security may be provided by the communications service, transparently to the SLE application. This Recommended Standard does not preclude the use of security features that are provided by the communications service or the local environment, nor does it assume the availability of such features.

3.1.5.1 The RCF service shall provide the following options with respect to the level of authentication of invocations and returns of operations:

a) 'all': all RCF invocations and returns, except the invocation of RCF-PEER-ABORT, shall be authenticated;

b) 'bind': only the RCF-BIND invocation and return shall be authenticated;

c) 'none': no RCF invocations or returns shall be authenticated.

3.1.5.2 SLE Complex Management and SLE Utilization Management shall agree on the level of authentication to be required for an association between a service user and a service provider and shall configure both entities accordingly.

3.1.5.3 SLE Complex Management and SLE Utilization Management shall agree on the algorithm used to generate and check credentials parameters and shall make this algorithm known to the service user and service provider together with associated parameters such as passwords or keys as necessary for the selected algorithm.

NOTES

1 The specification of the algorithms themselves is outside the scope of this Recommended Standard.

2 The `initiator-identifier` and `responder-identifier` parameters of the RCF-BIND operation identify the user and provider and therefore the applicable authentication level and algorithm necessary to generate and check credentials.

3.1.5.4 For operations for which authentication is required by the terms of the agreement between SLE Complex Management and SLE Utilization Management:

a) invocations shall include an `invoker-credentials` parameter to permit the performer to authenticate the invocation;

b) returns shall include a `performer-credentials` parameter to permit the invoker to authenticate the return.

3.1.6.8 Compliance with this Recommended Standard does not require the performer to process invocations concurrently; however, the performer must accept invocations from a non-blocking invoker and buffer and serialize them by local means not visible externally.

3.1.7 TIME

3.1.7.1 The time reference for all parameters containing a time value shall be based on Coordinated Universal Time (UTC).

NOTE – The type of all time parameters is specified in annex A.

3.1.7.2 The `earth-receive-time` parameter (see 3.6.2.3) shall be expressed using the CCSDS Day Segmented (CDS) time code (reference [5]) with ~~a resolution of one microsecond,~~ an epoch of 1958-01-01~~,~~ and a 16-bit day segment. Depending on the RCF service provider capabilities and/or the supported mission requirements, the time tag may have either a resolution of microseconds or a resolution of picoseconds.

3.1.7.3 The `earth-receive-time` parameter shall have a precision of one millisecond or better.

3.1.7.4 The `earth-receive-time` parameter shall be accurate to within one millisecond or better.

3.1.8 SETTING OF PARAMETERS

3.1.8.1 An RCF provider shall permit setting of the service configuration parameters as specified in table 3-1.

3.1.8.2 The range or set of values a parameter may assume is constrained by specification of its data type (see annex A).

3.1.8.3 Service management may further constrain the allowed values for a given service instance.

Table 3-1: Setting of RCF Service Configuration Parameters

Parameter	Service Management	RCF-START Operation	RCF-SCHEDULE-STATUS-REPORT Operation
delivery mode	X		
latency-limit	X		
maximum-delivery-rate	X		
maximum-reporting-cycle	X		
minimum-reporting-cycle	X		
permitted-global-VCID-set	X		
reporting-cycle			X
requested-global-VCID		X	
return-timeout-period	X		
service-instance-provision-period	X		
service-version-number	X		
transfer-buffer-size	X		

NOTES

1 The user can ascertain the current value of the parameters presented in table 3-11 by means of the RCF-GET-PARAMETER operation.

2 This Recommended Standard also refers to parameters that are set by service management, but are not listed in table 3-1. These parameters cannot be ascertained by means of the RCF-GET-PARAMETER operation.

3 The methods used by service management to control service provision and service production parameters are outside the scope of this Recommended Standard.

3.1.9 DELIVERY MODES

3.1.9.1 Timely Online Delivery Mode

3.1.9.1.1 For timely online delivery mode, the RCF service provider shall store frames acquired from the space link and certain information associated with those frames (as per 3.6.2) in a buffer called the transfer buffer. The stored information shall be an RCF-TRANSFER-DATA invocation or the equivalent thereof.

Table 3-2: RCF-BIND Parameters

Parameter	Invocation	Return
~~I~~invoker-credentials	M	
performer-credentials		M
initiator-identifier	M	
responder-identifier		M
responder-port-identifier	M	
service-type	M	
version-number	M	C
service-instance-identifier	M	
result		M
diagnostic		C

3.2.2.3 `performer-credentials`

The **`performer-credentials`** parameter shall provide information that enables the invoker to authenticate the return from the performance of RCF-BIND (see 3.1.4.1).

3.2.2.4 `initiator-identifier`

The **`initiator-identifier`** parameter shall identify the authority on whose behalf the initiating SLE application is initiating the association.

NOTES

1 The `initiator-identifier` parameter permits the responder to determine if the RCF-BIND operation is being invoked by the authorized initiator for this service instance.

2 Each value of the `initiator-identifier` parameter is associated with exactly one authentication level and exactly one authentication scheme.

3 If authentication based on credentials is used, this parameter may be redundant since the `initiator-identifier` value may be one constituent of the `invoker-credentials` parameter. However, the encoding may differ, and it may be convenient to have this parameter available in ‘clear text’ form.

3.2.2.7 `service-type`

3.2.2.7.1 The **`service-type`** parameter shall specify the type of service that will be provided if the bind operation succeeds.

3.2.2.7.2 For RCF service, the value of `service-type` shall be 'Rtn Ch Frames'.[1]

3.2.2.8 `version-number`

3.2.2.8.1 The **`version-number`** parameter shall identify the version number of the RCF service specification that is to govern this association if RCF-BIND succeeds.

3.2.2.8.2 `version-number` is conditionally present in the return based on the `result` parameter:

a) if the value of `result` is 'positive result', `version-number` shall be present in the return;

b) if the value of `result` is 'negative result', `version-number` shall not be present in the return.

3.2.2.8.3 If the value of the `result` parameter is 'positive result', the responder shall either:

a) accept the version proposed by the initiator by putting the same version number into the return; or,

b) if the responder supports version negotiation, propose a lower (earlier) version number by putting the lower number into the return.

3.2.2.8.4 If the responder implementation does not support the requested version and does not support a lower version (or does not support version negotiation), the responder shall reject the bind with the `diagnostic` parameter set to 'version not supported'.

3.2.2.8.5 If the responder proposes a lower version and the initiator implementation does not support version negotiation or does not support the version proposed by the responder, the initiator shall unbind the association.

3.2.2.8.6 The value of the `version-number` parameter for the RCF service defined by this issue of this Recommended Standard shall be '~~1~~2'.

[1] For the RCF-BIND operation, the `service-type` parameter is redundant, because the only valid value of `service-type` is 'Rtn Ch Frames'. However, it is anticipated that future work by CCSDS may result in RCF-BIND being superseded by a generic SLE-BIND operation that is invoked with any one of several SLE service types. The RCF-BIND `service-type` parameter is provided in an attempt to facilitate such a change.

Table 3-4: RCF-START Parameters

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

3.4.2.3 performer-credentials

The **performer-credentials** parameter shall provide information that enables the invoker to authenticate the return from the performance of RCF-START (see 3.1.4.1).

3.4.2.4 invoke-ID

The RCF service provider shall return unchanged the user-supplied value of the invoke-ID parameter (see 3.1.6).

3.4.2.5 start-time

3.4.2.5.1 The value of the **start-time** parameter shall be 'null', or it shall be a time value that indicates that only frames with an ERT equal to or later than start-time shall be delivered.

3.4.2.5.2 For the online delivery mode, only frames acquired during the space link session associated with this service instance shall be delivered, regardless of the value of start-time.

3.4.2.5.3 For the offline delivery mode, the provider shall deliver all available frames that meet the delivery criteria regardless of the space link session in which they were acquired.

3.4.2.5.4 For the online delivery mode, if start-time is 'null', the data transfer shall begin with the next frame that is acquired from the space link.

3.4.2.5.5 For the offline delivery mode, start-time must not be 'null'.

3.4.2.5.6 To be valid, start-time must satisfy the following criteria:

latency is negotiated between SLE Complex Management and SLE Utilization Management.

2 Offline delivery is only available for frames that already have been acquired when the RCF-START operation is invoked.

3.4.2.7 requested-global-VCID

3.4.2.7.1 The requested-**global-VCID** parameter shall identify the master channel or virtual channel that is to be provided to the user and shall consist of the TFVN, the SCID, and the VCID.

NOTES

1 The definitions of SCID and VCID depend on the TFVN. If the TFVN indicates that the virtual channel consists of TM Transfer Frames, then the definitions of SCID and VCID are as per reference [3]. If the frame version number indicates that the virtual channel consists of AOS Transfer Frames, then the definitions of SCID and VCID are as per reference [4].

2 The physical channel is not specified directly through the RCF service. Rather, the selection of physical channel is determined through the service package, which specifies the RAF service instance that is consumed by the RFP-FG that is producing the RCF service.

3 Depending on the configuration, for a given service instance, the selection of only one master channel or only one VC from a set of VCs (where the set may have a single member) or a single master channel plus a set of VCs is permitted. In case the permitted GVCID list contains a master channel but no virtual channels from that master channel, the service user is not permitted to request a virtual channel from this master channel.

3.4.2.7.2 The TFVN shall be a valid transfer frame version number defined by CCSDS.

NOTE – At the time of issuance of this Recommended Standard, the only valid TFVN were ‘00’ (version 1) and ‘01’ (version 2) (see references [3] and [4]).

3.4.2.7.3 The SCID shall be a valid spacecraft identifier as defined by CCSDS (see references [3] and [4]).

3.4.2.7.4 The VCID shall be a valid virtual channel identifier as defined by CCSDS (see references [3] and [4]) or it shall be the value ‘any’. The value ‘any’ indicates that a master channel, defined by the TFVN and the SCID, shall be provided by the RCF service. Otherwise, a virtual channel shall be provided by the RCF service.

3.4.2.8 `result`

The **`result`** parameter shall specify the result of the RCF-START operation and shall contain one of the following values:

a) 'positive result'—the RCF-START operation has been performed by the provider, and the provider shall henceforth invoke RCF-TRANSFER-DATA operations as needed to transfer to the user all available frames that meet the specified delivery criteria;

b) 'negative result'—the RCF-START operation has not been performed by the provider, and the provider shall not invoke any RCF-TRANSFER-DATA operations even if frames are available.

3.4.2.9 `diagnostic`

3.4.2.9.1 If `result` is 'negative result', the `diagnostic` parameter shall be present in the return, and its value shall be one of the following:

a) 'duplicate Invoke-ID'—the value of the `invoke-ID` parameter is the same as the `invoke-ID` of a previous, outstanding operation;

b) 'out of service'—the provider has been taken out of service for an indefinite period by management action;

c) 'unable to comply'—the provider is unable to transfer data at this time because of a fault affecting the provider;

d) 'invalid start time'—the value of the `start-time` provided in the invocation is not valid;

e) 'invalid stop time'—the value of the `stop-time` provided in the invocation is not valid;

f) 'missing time value'—for the offline delivery mode, the value of `start-time` and/or `stop-time` was 'null';

g) 'invalid global-VCID'—the value specified for the `requested-global-VCID` parameter is not valid;

h) 'other reason'—the reason for the negative result will have to be found by other means.

3.4.2.9.2 If `result` is 'positive result', the `diagnostic` parameter shall not be present in the return.

3.6 RCF-TRANSFER-DATA

3.6.1 PURPOSE

3.6.1.1 The provider shall invoke the RCF-TRANSFER-DATA operation to deliver a telemetry frame to the user.

3.6.1.2 The RCF-TRANSFER-DATA operation shall be an unconfirmed operation.

NOTE – Although RCF-TRANSFER-DATA is an unconfirmed operation, it is assumed that the communications service provides certain guarantees, as described in 1.3.1.

3.6.1.3 RCF-TRANSFER-DATA is valid only in state 3 (‘active’) and shall be invoked only by the provider.

3.6.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.6.2.1 General

The parameters of the RCF-TRANSFER-DATA operation shall be present in the invocation as specified in table 3-6.

Table 3-6: RCF-TRANSFER-DATA Parameters

Parameters	Invocation
`invoker-credentials`	M
`earth-receive-time`	M
`antenna-ID`	M
`data-link-continuity`	M
`private-annotation`	M
`data`	M

3.6.2.2 `invoker-credentials`

The **`invoker-credentials`** parameter shall provide information that enables the user to authenticate the RCF-TRANSFER-DATA invocation (see 3.1.4.1).

3.6.2.3 `earth-receive-time`

The **`earth-receive-time`** parameter shall contain the UTC time at which the signal event corresponding to the leading edge of the first symbol which has been influenced by the last bit of the attached sync marker that immediately preceded this telemetry frame was presented at the phase center of the antenna used to acquire the frame.

~~NOTE The first bit of the frame is the first bit following the attached sync marker.~~

NOTE – In case of punctured coding, the number of symbols influenced by each information bit is variable, depending on the puncture pattern. To minimize the resulting jitter of the `earth-receive-time` annotation with respect to the beginning of the frame, the end of the attached sync marker is used as the reference event.

3.6.2.4 `antenna-ID`

3.6.2.4.1 The **`antenna-ID`** parameter shall indicate which antenna of the SLE Complex was used to acquire the frame.

NOTE – `antenna-ID` is provided specifically to identify the physical location used as the reference point for the `earth-receive-time` parameter.

3.6.2.4.2 SLE Complex Management and SLE Utilization Management shall mutually agree upon the allowable values for `antenna-ID` and their interpretation.

NOTE – It is assumed that the value of the `antenna-ID` parameter is a reference to the actual location information, which is provided outside the scope of this service.

3.6.2.5 `data-link-continuity`

3.6.2.5.1 The **`data-link-continuity`** parameter shall indicate whether the frame from which the RCF was extracted was the direct successor of the previous frame on the master or virtual channel selected by means the RCF-START operation.

3.6.2.5.2 The **`data-link-continuity`** parameter shall contain an integer value:

a) a value of '-1' shall indicate that this is the first frame after the start of production or the selected channel is a master channel carrying AOS Transfer Frames and therefore no information regarding a discontinuity on the channel can be provided.

NOTE – AOS Transfer Frames do not contain a master channel frame counter.

b) a value of '~~(([MCFC$_n$ – MCFC$_{n-1}$] modulo 255) – 1)~~([MCFC$_n$ – MCFC$_{n-1}$ – 1] modulo 256)' if the selected channel is a master channel carrying TM Transfer Frames; MCFC$_n$ is the master channel frame count of the frame and MCFC$_{n-1}$ is the master channel frame count of the previous frame delivered by the production process for the given master channel.

c) a value of '~~(([VCFC$_n$ – VCFC$_{n-1}$] modulo 255) – 1)~~([VCFC$_n$ – VCFC$_{n-1}$ – 1] modulo 256)' if the selected channel is a virtual channel carrying TM Transfer Frames; VCFC$_n$ is the virtual channel frame count of the frame and VCFC$_{n-1}$ is the virtual channel frame count of the previous frame delivered by the production process for the given virtual channel.

d) a value of '~~(([VCFC$_n$ – VCFC$_{n-1}$] modulo 16777215) – 1)~~([VCFC$_n$ – VCFC$_{n-1}$ – 1] modulo 16777216)' if the selected channel is a virtual channel carrying AOS Transfer Frames; VCFC$_n$ is the virtual channel frame count of the frame and VCFC$_{n-1}$ is the virtual channel frame count of the previous frame delivered by the production process for the given virtual channel.

NOTE – The number of missing TM Transfer Frames reported is correct as long as the gap is less than 256 frames. For longer gaps it will normally be possible to resolve the ambiguity resulting from the modulo ~~255~~256 count based on the ERT of the frames and the nominal frame rate on the given master channel or virtual channel. For AOS Transfer Frames, the likelihood of an incorrectly reported gap size is much lower.

3.6.2.6 `private-annotation`

The **`private-annotation`** parameter shall be used to convey additional information that may be associated with a frame:

a) it may be set to 'null' to indicate that there is no private annotation;

b) if not 'null', there must be a prior arrangement between SLE Complex Management and SLE Utilization Management regarding the contents and interpretation of this parameter.

3.6.2.7 `data`

The value of the **`data`** parameter shall be the telemetry frame acquired by the provider from the RAF channel for delivery to the user. The frame (i.e., a AOS transfer frame) is delivered to the user, without any check symbols.

NOTE – The value of the `data` parameter does not include the attached sync marker.

3.7 RCF-SYNC-NOTIFY

3.7.1 PURPOSE

3.7.1.1 The RCF service provider shall invoke the RCF-SYNC-NOTIFY operation to notify the user of the occurrence of an event affecting the production of the RCF service.

NOTE – Notification of events may be of value to the user in understanding specific provider behavior, such as an interruption in frame delivery.

3.7.1.2 The RCF-SYNC-NOTIFY operation shall be an unconfirmed operation.

3.7.1.3 The order in which the RCF-SYNC-NOTIFY and RCF-TRANSFER-DATA operations are invoked shall reflect the actual chronology of events.

NOTE – For example, if an RCF-SYNC-NOTIFY operation is invoked after one RCF-TRANSFER-DATA operation but before another, then the event indicated by the notification occurred after the ERT of the frame associated with the preceding RCF-TRANSFER-DATA but before the ERT of the frame associated with the following RCF-TRANSFER-DATA.

3.7.1.4 RCF-SYNC-NOTIFY is valid only in state 3 (‘active’) and shall be invoked only by the provider.

3.7.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.7.2.1 General

The parameters of the RCF-SYNC-NOTIFY operation shall be present in the invocation as specified in table 3-7.

Table 3-7: RCF-SYNC-NOTIFY Parameters

Parameter	Invocation
`invoker-credentials`	M
`notification-type`	M
`notification-value`	C

3.7.2.2 `invoker-credentials`

The **`invoker-credentials`** parameter shall provide information that enables the user to authenticate the RCF-SYNC-NOTIFY invocation (see 3.1.4.1).

3.9 RCF-STATUS-REPORT

3.9.1 PURPOSE

3.9.1.1 The provider shall invoke the RCF-STATUS-REPORT operation to send a status report to the user.

3.9.1.2 RCF-STATUS-REPORT shall be an unconfirmed operation.

3.9.1.3 Status reports shall be sent (or not sent) in accordance with user requests conveyed by means of the RCF-SCHEDULE-STATUS-REPORT operation (see 3.8).

3.9.1.4 The RCF-STATUS-REPORT operation is valid only in states 2 (‘ready’) and 3 (‘active’) and shall be invoked only by the provider.

3.9.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.9.2.1 General

The parameters of the RCF-STATUS-REPORT operation shall be present in the invocation as specified in table 3-9.

Table 3-9: RCF-STATUS-REPORT Parameters

Parameters	Invocation
`invoker-credentials`	M
`number-of-frames-delivered`	M
`frame-sync-lock-status`	M
`symbol-sync-lock-status`	M
`subcarrier-lock-status`	M
`carrier-lock-status`	M
`production-status`	M

3.9.2.2 `invoker-credentials`

The **`invoker-credentials`** parameter shall provide information that enables the performer to authenticate the RCF-STATUS-REPORT invocation (see 3.1.5).

3.9.2.3 `number-of-frames-delivered`

The **`number-of-frames-delivered`** parameter shall specify the total number of telemetry frames with the ~~selected~~ `requested-global-VCID` value that have been delivered to the user since the start of the service instance provision period.

NOTE – This parameter is equivalent to the number of frames that an RCF service instance with the same ~~selected~~ `requested-global-VCID` value would deliver to the user while the service instance is in the active state.

3.9.2.4 `frame-sync-lock-status`

The **`frame-sync-lock-status`** parameter shall specify the current lock status of the frame synchronization process, the value of which shall be 'in-lock', 'out-of-lock', or 'unknown'.

3.9.2.5 `symbol-sync-lock-status`

The **`symbol-sync-lock-status`** parameter shall specify the current lock status of the symbol (or bit) synchronization process, the value of which shall be 'in-lock', 'out-of-lock', or 'unknown'.

3.9.2.6 `subcarrier-lock-status`

The **`subcarrier-lock-status`** parameter shall specify the current lock status of the subcarrier demodulation process, the value of which shall be 'in-lock', 'out-of-lock', or 'unknown'.

3.9.2.7 `carrier-lock-status`

The **`carrier-lock-status`** parameter shall specify the current lock status of the carrier demodulation process, the value of which shall be 'in-lock', 'out-of-lock', 'not in use', or 'unknown'.

3.9.2.8 `production-status`

The **`production-status`** parameter shall specify the current status of RCF production, the value of which shall be 'running', 'halted', or 'interrupted'.

NOTE – See 3.7.2.4 for a description of the `production-status` values.

3.10 RCF-GET-PARAMETER

3.10.1 PURPOSE

3.10.1.1 The user shall invoke the RCF-GET-PARAMETER operation to ascertain the value of an RCF service parameter.

3.10.1.2 The provider shall return a report of the outcome of the performance of the RCF-GET-PARAMETER operation to the user.

3.10.1.3 If the operation is successful, the current value of the specified RCF service parameter shall be provided to the user in the return from the operation.

3.10.1.4 RCF-GET-PARAMETER is valid in state 2 (‘ready’) and state 3 (‘active’) and shall be invoked only by the user.

3.10.2 INVOCATION, RETURN, AND PARAMETERS

3.10.2.1 General

The parameters of the RCF-GET-PARAMETER operation shall be present in the invocation and return as specified in table 3-10.

Table 3-10: RCF-GET-PARAMETER Parameters

Parameters	Invocation	Return
`invoker-credentials`	M	
`performer-credentials`		M
`invoke-ID`	M	M
`rcf-parameter`	M	C
`parameter-value`		C
~~R~~`result`		M
~~D~~`diagnostic`		C

3.10.2.2 `invoker-credentials`

The **`invoker-credentials`** parameter shall provide information that enables the performer to authenticate the RCF-GET-PARAMETER invocation (see 3.1.5).

b) 'negative result'—the RCF-GET-PARAMETER operation has not been performed for the reason specified in the `diagnostic` parameter.

Table 3-11: RCF Parameters

Parameter	Description
delivery-mode	The delivery mode for this instance of RCF service, which is set by service management (see 3.1.9): its value shall be 'timely online delivery mode', 'complete online delivery mode', or 'offline delivery mode'
latency-limit	The maximum allowable delivery latency time (in seconds) for the online delivery mode, as defined in 3.1.9.1 (i.e., the maximum delay from when the frame is acquired by the provider until the RCF extracted from it is delivered to the user): the value of this parameter shall be 'null' if the delivery mode is offline.
permitted-global-VCID-set	The set of global VCIDs permitted for this RCF service instance (see 3.4.2.7).
reporting-cycle	The current setting of the reporting cycle for status reports (see 3.8 and 3.9): the value is 'null' if cyclic reporting is off, otherwise it is the time (in seconds) between successive RCF-STATUS-REPORT invocations (see 3.8).
requested-global-VCID	If the provider is in state 3 ('active', the GVCID set by the RCF-START operation, used to determine which frames are selected for delivery. If the provider is not in state 3 ('active'), the GVCID value returned shall be 'undefined'.
return-timeout-period	The maximum time period (in seconds) permitted from when a confirmed RCF operation is invoked until the return is received by the invoker (see 4.1.3).
transfer-buffer-size	The size of the transfer buffer ~~(see 3.1.9)~~: the value of this parameter shall indicate the number of RCF-TRANSFER-DATA and RCF-SYNC-NOTIFY invocations that can be stored in the transfer buffer. The precise specification of the transfer buffer size may be found in 3.1.9.

3.10.2.8 `diagnostic`

3.10.2.8.1 If `result` is 'negative result', the `diagnostic` parameter shall be present in the return, and its value shall be one of the following:

a) 'duplicate Invoke-ID'—the value of the `invoke-ID` parameter is the same as the `invoke-ID` of a previous, outstanding operation;

3.11 RCF-PEER-ABORT

3.11.1 PURPOSE

3.11.1.1 The user or provider shall invoke the RCF-PEER-ABORT operations to notify the peer system that the local application detected an error that requires that the association between them be terminated abnormally.

3.11.1.2 RCF-PEER-REPORT shall be an unconfirmed operation.

3.11.1.3 RCF-PEER-ABORT is valid only in states 2 (‘ready’) and 3 (‘active’) and may be invoked by either the user or the provider.

3.11.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.11.2.1 General

The parameters of the RCF-PEER-ABORT operation shall be present in the invocation as specified in table 3-12.

Table 3-12: RCF-PEER-ABORT Parameters

Parameters	Invocation
`diagnostic`	M

3.11.2.2 `diagnostic`

The **`diagnostic`** parameter shall specify why the RCF-PEER-ABORT is being invoked, and its value shall be one of the following:

a) ‘access denied’—a responder with an identity as presented in the `responder-identifier` parameter of the RCF-BIND return is not known to the initiator (e.g., the value of the `responder-identifier` parameter does not match the authorized responder for any service instance known to the initiator);

b) ‘unexpected responder ID’—the value of the `responder-identifier` parameter in the RCF-BIND return does not match the identity of the authorized responder for this service instance as specified by service management;

c) ‘operational requirement’—the local system had to terminate the association to accommodate some other operational need;

d) ‘protocol error’—the local application detected an error in the sequencing of RCF service operations;

4 RCF PROTOCOL

4.1 GENERIC PROTOCOL CHARACTERISTICS

NOTE – This section specifies the handling of invalid SLE-PDUs and other failures affecting the protocol.

4.1.1 UNEXPECTED PROTOCOL DATA UNIT

If the peer application invokes an operation not allowed in the current state of the performer, the performer shall abort the association by invoking the RCF-PEER-ABORT operation with the `diagnostic` parameter set to 'protocol error'.

4.1.2 INVALID PROTOCOL DATA UNIT

If the application receives an invocation or return that contains an unrecognized operation type, contains a parameter of the wrong type, or is otherwise not decodable, the application shall abort the association by invoking the RCF-PEER-ABORT operation with the `diagnostic` parameter set to 'encoding error'.

4.1.3 MISSING RETURN

For confirmed operations, if the invoker does not receive the return from the performer within a timeout period specified by service management, the invoker shall abort the association by invoking the RCF-PEER-ABORT operation with the `diagnostic` parameter set to 'return timeout'.

NOTES

1 The timeout period shall be chosen taking into account performance of user and provider applications as well as the delays introduced by the underlying communications service.

2 In order to provide responsive service and short timeout periods, the generation of the return from an operation must not depend on any human interaction.

3 After invoking the RCF-UNBIND operation, the initiator must not invoke any further operations with the exception of the case addressed in 3.3.1.4 nor send any returns. The responder is not required to send any pending returns after having received the RCF-UNBIND invocation. Therefore, following an RCF-UNBIND invocation, the 'missing return' event may occur.

Table 4-2: Event Description References

Event	Reference
'data available'	3.1.9.1.2, 3.1.9.2.2, 3.1.9.3.2
'end of data'	3.7.2.3
'end of service instance provision period'	3.11.2.2
'invalid protocol data unit'	4.1.2
'loss of frame synchronization'	3.7.2.3
'not authenticated SLE-PDU'	4.1.7
'production status change'	3.7.2.3
'release timer expired'	3.1.9.1.4, 3.1.9.2.6
'reporting-cycle timer expired'	3.8.2.6
'return SLE-PDU with unsolicited Invoke-ID'	4.1.4
'return <n> timer expired'	4.1.3
'start of service instance provision period'	1.6.1.7.15

Table 4-3: Predicate Descriptions

Predicate	Evaluates to TRUE if
"buffer empty"	There are no RCF SLE-PDUs in the transfer buffer
"buffer full"	The transfer buffer cannot accommodate the currently available annotated frame or synchronous notification
"compatible"	The version number contained in (+rcfBindReturn) is supported by the provider
"complete online"	Delivery mode is complete online
~~"congested"~~	~~The underlying communications service cannot accept the contents of the transfer buffer because of congestion~~
"done"	The unbind-reason parameter value in the provider-initiated BIND invocation was 'end'
"end"	All checks on the UNBIND invocation are passed and the unbind-reason parameter value is 'end'
"immediately"	All parameter checks on the RCF-SCHEDULE-STATUS-REPORT are passed and the `report-request-type` value is 'immediately'
"offline"	Delivery mode is offline
"online"	Delivery mode is timely online or complete online
"periodically"	All parameter checks on the RCF-SCHEDULE-STATUS-REPORT are passed and the `report-request-type` value is 'periodically'
"positive result"	All checks on the invocation are passed
"provider initiated"	The RCF-BIND operation is specified to be initiated by the provider for this service instance

Predicate	Evaluates to TRUE if
“provision period”	Current time is inside the service instance provision period
“retry permitted”	The diagnostic value contained in the (-rcfBindReturn) is ‘unable to comply’ or ‘other’, and the service instance provision period is still active
“timely online”	Delivery mode is timely online

Table 4-4: Boolean Flags

Flag Name	Initial Value
“bind pending”	FALSE
“congested”	FALSE
“unbind pending”	FALSE

Table 4-5: Compound Action Definitions

Name	Actions Performed
{clean up}	stop release timer stop all return timers stop reporting-cycle timer reinitialize transfer buffer reset parameter values to those specified in service package
{immediate report}	(rcfStatusReportInvocation) stop reporting-cycle timer
{insert annotated frame}	annotate the available frame with the parameters of the RCF-TRANSFER-DATA operation insert the annotated frame into the transfer buffer
{invoke bind}	(rcfBindInvocation) set “bind pending” to TRUE start return <n> timer
{invoke unbind}	(rcfUnbindInvocation) stop reporting-cycle timer set “unbind pending” to TRUE start return <n> timer
{pass buffer contents}	stop release timer submit contents of transfer buffer to underlying communications service IF successful THEN set “congested” to FALSE ELSE set “congested” to TRUE reinitialize transfer buffer using the nominal size

Name	Actions Performed
{peer abort 'xxxx'}	stop release timer stop all return timers stop reporting-cycle timer reinitialize transfer buffer (rcfPeerAbortInvocation) with diagnostic set to 'xxxx'
{periodic report}	(rcfStatusReportInvocation) set reporting-cycle timer to the `reporting-cycle` value in the most recent SCHEDULE-STATUS-REPORT invocation start reporting-cycle timer
{provider unbind}	set "unbind pending" to FALSE stop all return timers
{return timeout}	(rcfPeerAbortInvocation) with diagnostic 'return timeout' set "bind pending" to FALSE set "unbind pending" to FALSE
{start release timer}	set release timer to latency limit start release timer
{sync notify 'xxxx'}	create an RCF synchronous notification with ~~diagnostic~~`notification-type` set to 'xxxx' insert the notification into the transfer buffer
{transmit buffer}	stop release timer submit the contents of transfer buffer to underlying communications service until accepted by that service reinitialize transfer buffer using nominal size
{user unbind}	stop reporting-cycle timer stop all return timers (rcfUnbindReturn)

```
ParameterName                          ::=   INTEGER
{  apidList                                  (2)
,  bitLockRequired                           (3)
,  blockingTimeoutPeriod                     (0)
,  blockingUsage                             (1)
,  bufferSize                                (4)
,  deliveryMode                              (6)
,  directiveInvocation                       (7)
,  directiveInvocationOnline                 (108)
,  expectedDirectiveIdentification           (8)
,  expectedEventInvocationIdentification     (9)
,  expectedSlduIdentification                (10)
,  fopSlidingWindow                          (11)
,  fopState                                  (12)
,  latencyLimit                              (15)
,  mapList                                   (16)
,  mapMuxControl                             (17)
,  mapMuxScheme                              (18)
,  maximumFrameLength                        (19)
,  maximumPacketLength                       (20)
,  maximumSlduLength                         (21)
,  modulationFrequency                       (22)
,  modulationIndex                           (23)
,  permittedControlWordTypeSet               (101)
,  permittedGvcidSet                         (24)
,  permittedTcVcidSet                        (102)
,  permittedTransmissionMode                 (107)
,  permittedUpdateModeSet                    (103)
,  plopInEffect                              (25)
,  reportingCycle                            (26)
,  requestedControlWordType                  (104)
,  requestedFrameQuality                     (27)
,  requestedGvcid                            (28)
,  requestedTcVcid                           (105)
,  requestedUpdateMode                       (106)
,  returnTimeoutPeriod                       (29)
,  rfAvailable                               (30)
,  rfAvailableRequired                       (31)
,  segmentHeader                             (32)
,  subcarrierToBitRateRatio                  (34)
,  timeoutType                               (35)
,  timerInitial                              (36)
,  transmissionLimit                         (37)
,  transmitterFrameSequenceNumber            (38)
,  vcMuxControl                              (39)
,  vcMuxScheme                               (40)
,  virtualChannel                            (41)
}

SlduStatusNotification  ::=   INTEGER
{  produceNotification           (0)
,  doNotProduceNotification      (1)
}

SpaceLinkDataUnit       ::=   OCTET STRING (SIZE (41 .. 65536))
```

```
Time                        ::=   CHOICE
{  ccsdsFormat         [0]   TimeCCSDS
,  picoFormat          [1]   TimeCCSDSpico
}

TimeCCSDS                   ::=   OCTET STRING (SIZE(8))
-- P-field is implicit (not present, defaulted to 41 hex
-- T-field:
-- 2 octets: number of days since 1958/01/01 00:00:00
-- 4 octets: number of milliseconds of the day
-- 2 octets: number of microseconds of the millisecond
--        (set to 0 if not used)
-- This definition reflects exactly the format of the CCSDS defined
-- time tag as used in spacelink data units (see Time Code Formats.
-- Recommendation for Space Data System Standards, CCSDS 301.0-B-3.
-- Blue Book.  Issue 3.  Washington, D.C.: CCSDS, January 2002).

TimeCCSDSpico               ::=   OCTET STRING (SIZE(10))
-- P-field is implicit (not present, defaulted to 42 hex
-- T-field:
-- 2 octets: number of days since 1958/01/01 00:00:00
-- 4 octets: number of milliseconds of the day
-- 4 octets: number of picoseconds of the millisecond
--        (set to 0 if not used)
-- This definition reflects exactly the format of the CCSDS defined
-- time tag as used in spacelink data units (see Time Code Formats.
-- Recommendation for Space Data System Standards, CCSDS 301.0-B-3.
-- Blue Book.  Issue 3.  Washington, D.C.: CCSDS, January 2002).

END
```

```
{   invokerCredentials                  Credentials
,   invokeId                            InvokeId
,   startTime                           ConditionalTime
,   stopTime                            ConditionalTime
,   grequestedGvcId                             GvcId
}

END
```

A2.7 SLE TRANSFER SERVICE—RCF OUTGOING PDUS

```
CCSDS-SLE-TRANSFER-SERVICE-RCF-OUTGOING-PDUS
{iso identified-organization(3) standards-producing-organization(112)
    ccsds(4) space-link-extension(3) sle-transfer-services(1)
    modules(1) return-channel-frames-service(13) version-one(1)
    asn1-outgoing-pdu(3)}

DEFINITIONS
IMPLICIT TAGS
::= BEGIN

IMPORTS     Credentials
,           IntUnsignedLong
,           InvokeId
,           SpaceLinkDataUnit
,           Time
    FROM CCSDS-SLE-TRANSFER-SERVICE-COMMON-TYPES
            SleAcknowledgement
,           SleScheduleStatusReportReturn
    FROM CCSDS-SLE-TRANSFER-SERVICE-COMMON-PDUS
            AntennaId
,           CarrierLockStatus
,           DiagnosticRcfGet
,           DiagnosticRcfStart
,           FrameSyncLockStatus
,           LockStatus
,           Notification
,           RcfGetParameter
,           RcfProductionStatus
,           SymbolLockStatus
    FROM CCSDS-SLE-TRANSFER-SERVICE-RCF-STRUCTURES
            SleBindInvocation
,           SleBindReturn
,           SlePeerAbort
,           SleUnbindInvocation
,           SleUnbindReturn
    FROM CCSDS-SLE-TRANSFER-SERVICE-BIND-TYPES
;

-- ============================================================
-- The first part of the module definition contains the RCF type
-- that contains all the possible PDUs the provider may send.
-- ============================================================

RcfProviderToUserPdu        ::= CHOICE
{ rcfBindInvocation                 [100]   SleBindInvocation
, rcfBindReturn                     [101]   SleBindReturn
, rcfUnbindInvocation               [102]   SleUnbindInvocation
, rcfUnbindReturn                   [103]   SleUnbindReturn
```

```
, rcfStartReturn                    [1]     RcfStartReturn
, rcfStopReturn                     [3]     SleAcknowledgement
, rcfTransferBuffer                 [8]     RcfTransferBuffer
, rcfScheduleStatusReportReturn     [5]     SleScheduleStatusReportReturn
, rcfStatusReportInvocation         [9]     RcfStatusReportInvocation
, rcfGetParameterReturn             [7]     RcfGetParameterReturn
, rcfPeerAbortInvocation            [104]   SlePeerAbort
}

-- ============================================================
-- The second part of the module definition contains the types
-- used by the RCF-PDUs declared in the first part.
-- ============================================================

FrameOrNotification             ::= CHOICE
{   annotatedFrame                      [0] RcfTransferDataInvocation
,   syncNotification                    [1] RcfSyncNotifyInvocation
}

RcfGetParameterReturn           ::= SEQUENCE
{   performerCredentials                Credentials
,   invokeId                            InvokeId
,   result                              CHOICE
    {   positiveResult                      [0] RcfGetParameter
    ,   negativeResult                      [1] DiagnosticRcfGet
    }
}

RcfStartReturn                  ::= SEQUENCE
{   performerCredentials                Credentials
,   invokeId                            InvokeId
,   result                              CHOICE
    {   positiveResult                      [0] NULL
    ,   negativeResult                      [1] DiagnosticRcfStart
    }
}

RcfStatusReportInvocation       ::= SEQUENCE
{   invokerCredentials                  Credentials
,   deliveredFrameNumber                IntUnsignedLong
,   frameSyncLockStatus                 FrameSyncLockStatus
,   symbolSyncLockStatus                SymbolLockStatus
,   subcarrierLockStatus                LockStatus
,   carrierLockStatus                   CarrierLockStatus
,   productionStatus                    RcfProductionStatus
}

RcfSyncNotifyInvocation         ::= SEQUENCE
{   invokerCredentials                  Credentials
,   notification                        Notification
}

RcfTransferBuffer               ::= SEQUENCE OF FrameOrNotification

RcfTransferDataInvocation   ::= SEQUENCE
{   invokerCredentials              Credentials
,   earthReceiveTime                Time
,   antennaId                       AntennaId
,   dataLinkContinuity              INTEGER (-1 .. 65535 16777215)
,   privateAnnotation               CHOICE
```

Term	Reference
SLE data channel	reference [1]
SLE Functional Group (SLE-FG)	reference [1]
SLE Protocol Data Unit (SLE-PDU)	reference [1]
SLE Service Data Unit (SLE-SDU)	reference [1]
SLE service package	reference [1]
~~SLE System~~	~~reference [1]~~
SLE transfer service instance	reference [1]
SLE transfer service production	reference [1]
SLE transfer service provision	reference [1]
SLE Utilization Management	reference [1]
spacecraft identifier (SCID)	subsection 1.6.1.7.16
space link	reference [1]
space link data channel	reference [1]
Space Link Data Unit (SL-DU)	reference [1]
space link session	reference [1]
telemetry frame	subsection 1.6.1.7.16
timely (online delivery mode)	subsections 2.3, 3.1.9.1
TM Transfer Frame	reference [3]
transfer buffer	subsections 2.6.4.6.2, 3.1.9
Transfer Frame Version Number	subsection 1.6.1.7.18
unbound (state)	subsection 2.6.4.2
unconfirmed operation	subsection 1.6.1.7.19
user-initiated	subsections 2.3, 3.2.1
virtual channel	subsection 1.6.1.7.20
virtual channel identifier (VCID)	subsection 1.6.1.7.21

ANNEX E

INFORMATIVE REFERENCES

(INFORMATIVE)

[E1] *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.

[E2] *Cross Support Concept — Part 1: Space Link Extension Services*. Report Concerning Space Data System Standards, CCSDS 910.3-G-3. Green Book. Issue 3. Washington, D.C.: CCSDS, March 2006.

[E3] *Telemetry Channel Coding*. Recommendation for Space Data System Standards, CCSDS 101.0-B-6-S. Historical Recommendation. Issue 6-S. Washington, D.C.: CCSDS, (October 2002) August 2005.

[E4] *Packet Telemetry*. Recommendation for Space Data System Standards, CCSDS 102.0-B-5-S. Historical Recommendation. Issue 5-S. Washington, D.C.: CCSDS, (November 2000) August 2005.

[E5] *Advanced Orbiting Systems, Networks and Data Links: Architectural Specification*. Recommendation for Space Data System Standards, CCSDS 701.0-B-3-S. Historical Recommendation. Issue 3-S. Washington, D.C.: CCSDS, (June 2001) August 2005.

[E6] *Space Link Extension—Internet Protocol for Transfer Services*. Recommendation for Space Data System Standards, CCSDS 913.1-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2008.