Draft Recommendation for
Space Data System Standards

|  |
| --- |
| Cross Support Transfer ServiceS - Tracking Data Service |

Draft Recommended Standard

**CCSDS-922.2-W-0.14**

**WHITE BOOK**

June 2016

AUTHORITY

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | Issue: | White Book, Issue 0.14 |  |
|  | Date: | June 2016 |  |
|  | Location: | Not Applicable |  |
|  |  |  |  |

**(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING STATEMENT OF AUTHORITY:)**

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorization of CCSDS documents is detailed in the *Procedures Manual for the Consultative Committee for Space Data Systems*, and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the address below.

This document is published and maintained by:

CCSDS Secretariat

Space Communications and Navigation Office, 7L70

Space Operations Mission Directorate

NASA Headquarters

Washington, DC 20546-0001, USA

STATEMENT OF INTENT

**(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING STATEMENT OF INTENT:)**

The Consultative Committee for Space Data Systems (CCSDS) is an organization officially established by the management of its members. The Committee meets periodically to address data systems problems that are common to all participants, and to formulate sound technical solutions to these problems. Inasmuch as participation in the CCSDS is completely voluntary, the results of Committee actions are termed **Recommended Standards** and are not considered binding on any Agency.

This **Recommended Standard** is issued by, and represents the consensus of, the CCSDS members. Endorsement of this **Recommendation** is entirely voluntary. Endorsement, however, indicates the following understandings:

o Whenever a member establishes a CCSDS-related **standard**, this **standard** will be in accord with the relevant **Recommended Standard**. Establishing such a **standard** does not preclude other provisions which a member may develop.

o Whenever a member establishes a CCSDS-related **standard**, that member will provide other CCSDS members with the following information:

 -- The **standard** itself.

 -- The anticipated date of initial operational capability.

 -- The anticipated duration of operational service.

o Specific service arrangements shall be made via memoranda of agreement. Neither this **Recommended Standard** nor any ensuing **standard** is a substitute for a memorandum of agreement.

No later than five years from its date of issuance, this **Recommended Standard** will be reviewed by the CCSDS to determine whether it should: (1) remain in effect without change; (2) be changed to reflect the impact of new technologies, new requirements, or new directions; or (3) be retired or canceled.

In those instances when a new version of a **Recommended Standard** is issued, existing CCSDS-related member standards and implementations are not negated or deemed to be non-CCSDS compatible. It is the responsibility of each member to determine when such standards or implementations are to be modified. Each member is, however, strongly encouraged to direct planning for its new standards and implementations towards the later version of the Recommended Standard.

FOREWORD

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CCSDS shall not be held responsible for identifying any or all such patent rights.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommended Standard is therefore subject to CCSDS document management and change control procedures, which are defined in the *Procedures Manual for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4). 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.
* Canadian Space Agency (CSA)/Canada.
* Centre National d’Etudes Spatiales (CNES)/France.
* China National Space Administration (CNSA)/People’s Republic of China.
* Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)/Germany.
* European Space Agency (ESA)/Europe.
* Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
* Japan Aerospace Exploration Agency (JAXA)/Japan.
* National Aeronautics and Space Administration (NASA)/USA.
* Russian Federal Space Agency (RFSA)/Russian Federation.
* UK Space Agency/United Kingdom.

Observer Agencies

* Austrian Space Agency (ASA)/Austria.
* Belgian Federal Science Policy Office (BFSPO)/Belgium.
* Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
* China Satellite Launch and Tracking Control General, Beijing Institute of Tracking and Telecommunications Technology (CLTC/BITTT)/China.
* Chinese Academy of Sciences (CAS)/China.
* Chinese Academy of Space Technology (CAST)/China.
* Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
* CSIR Satellite Applications Centre (CSIR)/Republic of South Africa.
* Danish National Space Center (DNSC)/Denmark.
* Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
* European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
* European Telecommunications Satellite Organization (EUTELSAT)/Europe.
* Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
* 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.
* Ministry of Communications (MOC)/Israel.
* National Institute of Information and Communications Technology (NICT)/Japan.
* National Oceanic and Atmospheric Administration (NOAA)/USA.
* National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
* National Space Organization (NSPO)/Chinese Taipei.
* Naval Center for Space Technology (NCST)/USA.
* Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
* 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 922.2-W-0.1 | Tracking Data Cross Support Transfer Service | June 2008 | First Draft. |
| CCSDS 922.2-W-0.2 | Tracking Data Cross Support Transfer Service | August 2009 | * Uses simpler approach whereby each time sample has both metadata and data.
* Removes clock, media, and meteorological data.
* Specifies object identifiers for the extension parameters.
* Specifies an XML schema for the extension parameters.
 |
| CCSDS 922.2-W-0.2A | Tracking Data Cross Support Transfer Service | September 2009 | * Changes the transfer of the TDM Header from the NOTIFY invocation to the START positive return.
* Cleans up multiple typographical errors.
 |
| CCSDS 922.2-W-0.4 | Tracking Data Cross Support Transfer Service | February 2010 | * Compliance with Guidelines
* VLBI and Delta-DOR removed per D. Berry’s suggestion
* Functional groups in section 2 replaced with functional resources
* Further extension of START positive return prohibited (simplifies syntax)
 |
| CCSDS 922.2-R-0.5 | Tracking Data Cross Support Transfer Service | July 2010 | * Reflects comments made by Yves Doat and David Berry, and CSTSWG members at Portsmouth
* ASN.1 restructured into Annexes.
* References to specific subsections of referenced documents have been changed to references to section/subsection titles for robustness.
 |
| CCSDS 922.2-R-0.5a | Tracking Data Cross Support Transfer Service | August 2010 | * Corrects errors in ASN.1 module annexes
 |
| CCSDS 922.2-R-0.6 | Tracking Data Cross Support Transfer Service | March 2013 | * Responded to comments from David Berry
* Removed normative references to and dependencies on the Guidelines.
* Updates Section 2 for current Functional Resource usage.
* Reworked Managed Information section to identify which configured parameters from CSTS SFW are to be managed.
* Moved Tracking Data Production to a normative annex. Added Tracking Data Production Managed Information section and Tracking Data Production Monitored Parameters, Notifiable Events, and Directives section.
* Added Monitored Parameters and Notifiable Events of TD-CSTS Provider section.
 |
| CCSDS 922.2-R-0.6 (concluded) |  |  | * Changed normative references to SCCS-SM Blue-1 parameters into informative notes in Annex A.
* Extended Annex B (Service Object Identifiers Module) to include OIDs of the TD-CSTS Provider, TDM Segment Generation, and Tracking Data Recording Buffer FR Types
 |
| CCSDS 922.2-R-0.7 | Tracking Data Cross Support Transfer Service | March 2014 | Added Info Query and Notification procedures, procedure parameters and eventsAdjusted terminology to match latest CSTS SFW (e.g., Return Buffer) |
| CCSDS 922.2-R-0.8 | Tracking Data Cross Support Transfer Service | May 2014 | * Removed the Notification procedure
* Limited scope of Info Query procedure to configuration parameters only.
* Added a new Refinement of Service-Generic Parameters and Events section (section 7).
* Made NOTIFY operation of Buffered Tracking Data Message Delivery procedure adopted (it was refined, but that’s now addressed by the new section 7).
* Removed references to SCCS-SM –B-1; changed to Extensible SCCS-SM suite.
* Default list of parameters is set to all parameters that are accessible by the Info Query procedure.
* Added Configuration Parameters and Procedure-Specific Events section to the BTDMD procedures section.
* Authority, Statement of Intent, Foreword, and list of CCSDS agencies added.
* Security, SANA, and Patent Considerations annex added.
 |
| CCSDS 922.2-R-0.8(concluded) | Tracking Data Cross Support Transfer Service  | May 2014 | * Defined the default list
* Added clarification that the selection of tracking data types to be delivered by BTDMD is made through Service Management and \*not\* through the START operation.
 |
| CCSDS 922.2-R-0.9 | Cross Support Transfer Services - Tracking Data Service | September 2014 | * Changed name of document to conrform with Secretariat style
* Added PICS Proforma Annex.
 |
| CCSDS 922.2-W-0.10 | Cross Support Transfer Services - Tracking Data Service | July 2015 | * Updated to address decisions made at Spring 2015 meeting
* Examples of TDM Header and TDM Atomic Segments added to ANNEX A.
* Information Query is now directly adopted – the IQ procedure section has been deleted.
* The tracking-data-types configuration parameter is now accessible.
 |
| CCSDS 922.2-W-0.11 | Cross Support Transfer Services - Tracking Data Service | November 2015 | * Updated in response to S. Gully’s email comments.
* Fixes additional typos
 |
| CCSDS 922.2-W-0.12 | Cross Support Transfer Services - Tracking Data Service | November 2015 | * Cleans up distinctions between procedure configuration parameters and service management parameters, and aligns their usage with the realignment in the SFW at the Darmstadt meeting. Adds “procedure configuration parameter” and “service management parameter” to the list of terms defined in the SFW in section 1.6.1.
* Removes serviceIdentifiers node from the CSTS OID tree, so that the trackingDataService node falls directly under the services node.
 |
| CCSDS 922.2-W-0.13 | Cross Support Transfer Services - Tracking Data Service | March 2016 | * Final consistency adjustments for SFW version dated 20160205
 |
| CCSDS 922.2-W-0.14 | Cross Support Transfer Services - Tracking Data Service | June 2016 | * Final consistency adjustments for SFW version dated 20160511,
* Minimized references to functional resources to limit their presence to be only whre they are absolutely pertinent to the specification.
 |

**CONTENTS**

Section Page

[1 introduction 1-1](#_Toc453570208)

[1.1 Purpose of thIS RECOMMENDATION 1-1](#_Toc453570209)

[1.2 SCOPE 1-1](#_Toc453570210)

[1.2.1 Scope of the TD-CSTS 1-1](#_Toc453570211)

[1.2.2 Scope of this recommended Standard 1-1](#_Toc453570212)

[1.3 APPLICABILITY 1-2](#_Toc453570213)

[1.4 rationale 1-2](#_Toc453570214)

[1.5 document strUcture 1-2](#_Toc453570215)

[1.5.1 document organization 1-2](#_Toc453570216)

[1.5.2 cross support Transfer services documentation 1-4](#_Toc453570217)

[1.6 definitions, nomenclature, and conventions 1-6](#_Toc453570218)

[1.6.1 Terms 1-6](#_Toc453570219)

[1.6.2 nomenclature 1-8](#_Toc453570220)

[1.6.3 conventions 1-8](#_Toc453570221)

[1.7 references 1-9](#_Toc453570222)

[2 Overview OF THE Tracking Data Cross Support Transfer SERVICE 2-1](#_Toc453570223)

[2.1 Service Summary 2-1](#_Toc453570224)

[2.2 Functional Description 2-1](#_Toc453570225)

[2.2.1 Service Production 2-2](#_Toc453570226)

[2.2.2 Service Provision 2-5](#_Toc453570227)

[2.3 Service ManagEment 2-6](#_Toc453570228)

[2.4 Cross Support View 2-7](#_Toc453570229)

[2.5 Operational scenario 2-10](#_Toc453570230)

[2.5.1 Service Management/Service Planning activities 2-10](#_Toc453570231)

[2.5.2 SLS Service Package Execution 2-11](#_Toc453570232)

[2.5.3 PosT-Space Link Session 2-15](#_Toc453570233)

[3 Tracking Data Cross Support Transfer Service Composition 3-1](#_Toc453570234)

[3.1 DISCUSSION 3-1](#_Toc453570235)

[3.2 Procedures of the Tracking Data Cross Support Transfer Service 3-1](#_Toc453570236)

[3.3 Tracking Data Cross Support Transfer Service State Machine 3-2](#_Toc453570237)

[4 Buffered Tracking Data Message Delivery Procedure 4-1](#_Toc453570238)

[4.1 Discussion 4-1](#_Toc453570239)

[4.1.1 Purpose 4-1](#_Toc453570240)

[4.1.2 Concept 4-1](#_Toc453570241)

[4.2 Procedure Type Identifier 4-1](#_Toc453570242)

[4.3 Extension 4-2](#_Toc453570243)

[4.4 Behavior 4-2](#_Toc453570244)

[4.4.1 DeriVED Behavior - STARTING 4-2](#_Toc453570245)

[4.4.2 DeriVED Behavior – Transferring Data and Notifications 4-2](#_Toc453570246)

[4.5 Required Operations 4-3](#_Toc453570247)

[4.5.6 START (confirmed) 4-3](#_Toc453570248)

[4.5.7 Transfer-Data (Unconfirmed) 4-4](#_Toc453570249)

[4.6 Configuration Parameters 4-4](#_Toc453570250)

[4.7 Procedure STATE TABLE (Provider Side) 4-5](#_Toc453570251)

[5 Setting of Configuration Parameters Inherited from Framewwork Procedures 5-1](#_Toc453570252)

[5.1 General 5-1](#_Toc453570253)

[5.2 Association Control Procedure Configuration Parameters 5-1](#_Toc453570254)

[5.3 Buffered Tracking Data Message Delivery Procedure Configuration Parameters 5-1](#_Toc453570255)

[5.4 Information Query Procedure Configuration Parameters 5-2](#_Toc453570256)

[6 Refinement of definitions of Framework Parameters, events, directives, and Diagnostic Values Used by the Tracking Data Service 7-1](#_Toc453570257)

[6.1 General 7-1](#_Toc453570258)

[6.2 Service-generic Event Definition Refinement 7-1](#_Toc453570259)

[6.2.1 Production Configured 7-1](#_Toc453570260)

[6.2.2 Production Interrupted 7-2](#_Toc453570261)

[6.2.3 Production Halted 7-2](#_Toc453570262)

[6.2.4 Production Operational 7-2](#_Toc453570263)

[ANNEX A SPecification of TD-CSTS TDM Components (normative) A-1](#_Toc436648918)

[ANNEX B Service Object Identifiers Module (Normative) B-1](#_Toc436648919)

[ANNEX C PRocedure – Buffered Tracking Data Message Data Delivery PDUs (Normative) C-1](#_Toc436648920)

[ANNEX D Tracking Data Service Procedure Parameters, Events, and Directives (Normative) D-1](#_Toc436648921)

[ANNEX E Tracking Data Production (Normative) E-1](#_Toc436648922)

[ANNEX F Implementation Conformance Statement Proforma (Normative) F-1](#_Toc436648923)

[ANNEX G Security, SANA, and Patent Considerations (Informative) G-1](#_Toc436648924)

[ANNEX H Acronyms (INFormative) H-1](#_Toc436648925)

[ANNEX I Informative references (INformative) I-1](#_Toc436648926)

[ANNEX J Cross References to Cross Support Transfer Service Specification Framework (Informative) J-1](#_Toc436648927)

Figure Page

[1-1 Cross Support Services Documentation 1-4](#_Toc436649727)

[2-1 Production and Provision of Tracking Data Cross Support Transfer Services 2-4](#_Toc436649728)

[2-2 Example of the Management and Provision of Tracking Data Service Instances for a Service Package 2-10](#_Toc436649729)

Table Page

[4-1 Buffered Tracking Data Message Delivery Required Operations 4-3](#_Toc436649894)

[4-2 START Extension Parameter 4-4](#_Toc436649895)

[4-3 Buffered Tracking Data Message Delivery Configuration Parameters 4-5](#_Toc436649896)

[F-1 Identification of PICS F-3](#_Toc436649897)

[F-2 Identification of Implementation Under Test F-4](#_Toc436649898)

[F-3 Identification of Supplier F-4](#_Toc436649899)

[F-4 Identification of Specification F-4](#_Toc436649900)

[F-5 Required Procedures F-5](#_Toc436649901)

[F-6 Required PDUs F-6](#_Toc436649902)

[F-7 BIND Invocation Parameters F-7](#_Toc436649903)

[F-8 BIND Return Parameters F-8](#_Toc436649904)

[F-9 PEER-ABORT Invocation Parameters F-9](#_Toc436649905)

[F-10 UNBIND Invocation Parameters F-9](#_Toc436649906)

[F-11 UNBIND Return Parameters F-10](#_Toc436649907)

[F-12 GET Invocation Parameters F-10](#_Toc436649908)

[F-13 GET Return Parameters F-11](#_Toc436649909)

[F-14 START Invocation Parameters F-12](#_Toc436649910)

[F-15 START Return Parameters F-12](#_Toc436649911)

[F-16 STOP Invocation Parameters F-14](#_Toc436649912)

[F-17 STOP Return Parameters F-15](#_Toc436649913)

[F-18 TRANSFER-DATA Invocation Parameters F-16](#_Toc436649914)

[F-19 NOTIFY Invocation Parameters F-17](#_Toc436649915)

[J-1 Cross Reference to CSTS SFW Sections and Paragraphs J-1](#_Toc436649916)

# introduction

## Purpose of thIS RECOMMENDATION

This Recommendation defines the Tracking Data Cross Support Transfer Service (TD-CSTS). The TD-CSTS is a transfer service that allows a spaceflight mission to receive periodic measurements of tracking data as soon as they are generated by a Cross Support Complex or anytime thereafter. The TD-CSTS delivers the tracking data as a periodic sequence of data segments formatted in accordance with components of a CCSDS *Tracking Data Message* (TDM), as specified in the TDM Recommended Standard [3]. The TD-CSTS is constructed using procedures and operations defined in the *Cross Support Transfer Service Specifications Framework* [1].

## SCOPE

### Scope of the TD-CSTS

The TD-CSTS transfers tracking data associated with the signal-related and angle-related TDM keyword categories [3].

The TD-CSTS does not deliver data using the time related keywords, media related keywords, or meteorological related keywords of the TDM, which are better suited to a file related transfer than to a real time transfer.

The TD-CSTS does not deliver VLBI/Delta-Differential One-way Ranging (DOR)-related data, which typically comprises a single small TDM file (a page or two in length) that is created as a result of substantial post-processing on the raw tracking data.

### Scope of this recommended Standard

This Recommended Standard defines the TD-CSTS in terms of:

a) the CSTS Framework procedures that comprise the service;

b) the extensions and refinements of the behavior of those CSTS procedures necessary to provide the transfer service;

c) the extensions and refinements of standard CSTS operations associated with each of the procedures;

d) the relationships among the procedures that comprise the service.

e) The requirements on tracking service production to enable the proper operation of the TD-CSTS.

This Recommended Standard does not specify:

a) individual implementations or products;

b) the implementation of entities or interfaces within real systems;

c) the methods or technologies required to measure the values of tracking service parameters;

d) the methods or technologies required for communication;

e) the management activities necessary to schedule, configure, and control the TD-CSTS.

## APPLICABILITY

The applicability and limits of applicability of Cross Support Transfer Services in general, as described in [1], pertain to the TD-CSTS.

The TD-CSTS is most applicable in situations where tracking data measurements are needed by the user within some relatively-short interval from the time at which they were taken (i.e., in “near-real-time”).

The TD-CSTS can also be used in situations where all tracking data measurements are not required to be delivered in near-real-time.

NOTE - However, the TDM that results from transfer via the TD-CSTS will likely be larger than a TDM that contains the same information but that is generated at a single time (i.e., after the conclusion of the tracking pass) because the transfer of TDM data via the TD-CSTS involves the repetition of metadata with every tracking data measurement in order to facilitate the incremental transfer of tracking data in near-real-time.

As specified in reference [3], TDMs can contain raw or processed tracking data. The TD-CSTS is therefore also capable of transferring processed as well as raw data.

## rationale

The goal of this Recommendation is to create a standard for interoperability for the exchange of tracking data between the cross support elements of various space Agencies and the users of the cross support services that they provide.

## document strUcture

### document organization

Section 2 describes the Tracking Data Cross Support Transfer service in terms of:

- the role of Service Management with respect to the TD-CSTS;

- the allocation of production and provision of the TD-CSTS to functional resources;

- the cross support view of the TD-CSTS;

- the functional description of the production and provision of the service; and

- an operational scenario that illustrates some of the more significant aspects of the service.

Section 3 specifies the top-level composition of the TD-CSTS. The service type identifier is declared, the procedures that comprise the service are identified, and the CSTS state machine that applies to the TD-CSTS is specified.

Section 4 defines the Buffered Tracking Data Message Delivery procedure as a derivation of the CSTS SFW Buffered Data Delivery procedure.

Section 5 specifies how the procedure configuration parameters are to be set for the TD-CSTS.

Section 6 specifies the Monitored Data Service-specific versions of the service-generic parameter and events that are defined in the CSTS SFW.Section 7 defines the refinement of CSTS SFW-defined parameters and events as they apply to the TD-CSTS.

ANNEX A specifies the structure and content of the data components that are used by the TD-CSTS to incrementally transfer TDMs. This annex also includes an example TDM Header and TDM Atomic Segments.

ANNEX B formally specifies the object identifiers for the Tracking Data transfer service, the TD-CSTS Provider Functional Resource Type, and the TDM Recording Buffer Functional Resource Type.

ANNEX C formally specifies the ASN.1 protocol data units (PDUs) for the Buffered Tracking Data Message Delivery procedure.

ANNEX D formally specifies the ASN.1 parameters, events, and directives for the Buffered Tracking Data Message Delivery procedure.

ANNEX E defines the tracking data production process. In particular, it specifies how tracking data measurements and associated metadata are to be generated and stored so that, when transferred by TD-CSTS instances, the result will be data structures that conform to the syntactic and semantic requirements for Tracking Data Messages as specified in [3].

ANNEX F documents the Implementation Conformance Statement (ICS) Proforma for the TD-CSTS.

ANNEX G addresses the security, Space Assigned Numbers Authority (SANA), and patent considerations associated with the TD-CSTS.

ANNEX H lists the acronyms used in this Recommended Standard.

ANNEX I lists the informative references cited in this Recommended Standard.

### cross support Transfer services documentation

The basic organization of the CSTS documentation and the relationship to CSTS documentation is shown inFigure 1‑1.



Figure 1‑1 : Cross Support Services Documentation

The Cross Support Services documents that are related to Cross Support Transfer Services are:

1. *Cross Support Concept—Part 1: Space Link Extension Services* (reference [I1]) A report introducing the concepts of cross support and the SLE services. Many of the concepts for the SLE transfer services have been adopted for the CSTSes (see h) below);
2. *Cross Support Reference Model—Part 1: Space Link Extension Services* (reference [2]): A Recommended Standard that defines the framework and terminology for the specification of SLE services. Much of the framework and terminology of this reference model has been adopted or adapted for CSTSes in [1];
3. *Space Communication Cross Support Service Management* suite (references [I7], [I8], and [I9]). Recommended Standards that specify the Service Management Information Entities that are used to configure and schedule CSTSes;
4. *The SLE Transfer Services suite*: The SLE Transfer Services are a suite of cross support transfer services that are used to transfer specific telecommand and telemetry protocol data units. The SLE Transfer Services are closely related to the CSTS suite in that they collectively define the set of operations that are the basis for the CSTS Specification Framework. However, due to history (the SLE Transfer Services were already specified and implemented prior to development of the CSTS Specification Framework) the SLE Transfer Services are separated from CSTSes;
5. *Space Link Extension - Internet Protocol for Transfer Services* (reference [I2]): A Recommended Standard that defines a protocol for transfer of Protocol Data Units (PDU) defined in the Cross Support Transfer Services. This Recommended Standard was originally developed to support SLE transfer services (hence the title), but it is also applicable to (and specified for) use by Cross Support Transfer Services.

The documents specific to Cross Support Transfer Services are:

1. *Cross Support Transfer Services Specification Framework* (reference [1]): A Recommended Standard that defines basic building blocks for the specification of Cross Support Transfer Service procedures;
2. *Guideline for Specification of Cross-Support Transfer Services*: A Recommended Practice that, when published, will define the guidelines for construction of a Cross Support Transfer Service based on the CSTS Specification Framework;

NOTE - As of the publication of this Recommended Standard, the Guidelines Magenta Book is in-progress. It is not required to understand this Recommended Standard.

1. *Cross Support Transfer Services Specification Framework Concepts* (reference [I3]): A report that provides tutorial material on the objectives and concepts of the CSTS Specification Framework;
2. Cross Support Transfer Services Suite: The set of specifications for actual CSTSes built from the procedures in the CSTS Specification Framework and in accordance with the CSTS Guidelines. The Cross Support Transfer Service Suite includes this Recommended Standard.

## definitions, nomenclature, and conventions

### Terms

#### Terms Defined in the Cross Support Transfer Services Specification Framework (CSTS Framework) [1]

1. Association Control procedure;
2. Blocking [operation];
3. Buffered Data Delivery procedure;
4. Complete [data delivery mode];
5. Cross Support complex;
6. Cross support service production;
7. Cross support transfer service provision;
8. Cross support transfer service;
9. delivery-mode;
10. discardable;
11. latency-limit;
12. Non-blocking [operation];
13. non-discardable;
14. procedure configuration parameter;
15. qualified parameter;
16. Real-time [data delivery mode];
17. Recording Buffer;
18. service instance provision period;
19. service management parameter;
20. Service Production Data Unit;
21. Service Production Event Notification;
22. service-user-responding-timer;
23. start-generation-time;
24. stop-generation-time;
25. transfer-buffer-size;
26. TransferDataInvocation.

#### Terms Defined in the Cross Support Reference Model [2]

1. Complex Management (CM);
2. Forward CLTU (F-CLTU);
3. Mission Data Operations System (MDOS);
4. Return All Frames (RAF);
5. Service package;
6. Space link session;
7. Utilization Management (UM).

#### Terms Defined as Keywords in the Tracking Data Message Specification [3]

1. Carrier power;
2. Carrier power to noise spectral density;
3. Differential One-way Ranging (DOR);
4. Doppler (instantaneous);
5. Doppler (integrated);
6. Header;
7. Keyword;
8. ORIGINATOR;
9. PARTICIPANT;
10. Range;
11. Ranging power to noise spectral density;
12. Receive frequency;
13. Segment;
14. timetag;
15. TDM Data section;
16. TDM Metadata section;
17. Tracking Data Message;
18. Tracking Data Record;
19. Transmit frequency;
20. Transmit frequency rate;
21. Very long baseline interferometry (VLBI).

#### Terms Defined in the Extensible Space Communication Cross Support Service Management Concept [I7]

a) Retrieval service package.

#### Terms Defined in this Specification

**TDM Atomic Segment** – A TDM Segment (i.e., Metadata Section and Data Section), constrained to contain a single tracking data measurement and its associated metadata.

### nomenclature

The following conventions apply throughout this Recommendation:

1. the words ‘shall’ and ‘must’ imply a binding and verifiable specification;
2. the word ‘should’ implies an optional, but desirable, specification;
3. the word ‘may’ implies an optional specification;
4. the words ‘is’, ‘are’, and ‘will’ imply statements of fact.

### conventions

This Recommendation uses the conventions defined in [1].

## references

[1] *Cross Support Transfer Service Specification Framework*. Draft Recommended Standard, CCSDS 921.1-R-2. Red Book. October 2015.

[2] *Cross Support Reference Model, Part 1: Space Link Extension*. Recommended Standard, CCSDS 910.4-B-2. Blue Book. October 2005.

[3] *Tracking Data Message*. Recommended Standard, CCSDS 503.0-B-1. Blue Book. November 2007.

[4] *Information Technology—ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*. International Standard, ISO/IEC 8825-1:2002. 3rd ed. Geneva: ISO, 2002.

# Overview OF THE Tracking Data Cross Support Transfer SERVICE

## Service Summary

The Tracking Data CSTS is a CCSDS Cross Support Transfer Service that provides a user with the capability to obtain tracking data in periodic measurements that are taken during the utilization phase of the cross support service Package.

The tracking data are formatted in accordance with the CCSDS *Tracking Data Message* Recommended Standard [3]. Tracking data measurement types supported by the Tracking Data Message (TDM) format include (among others) antenna angles, range, Doppler, and transmit and receive frequencies. The data is transferred by the TD-CSTS in such a way that reconstitution of a valid TDM is accomplished by direct extraction and concatenation of data fields from TD-CSTS protocol data units, with no format conversions required on the service user side.

The TD-CSTS supports delivery of periodic tracking data measurements in two data delivery modes: real-time and complete. The data delivery mode of an instance of the TD-CSTS is established by Service Management prior to the beginning of the service instance provision period.

In the real-time data delivery mode, the tracking data measurements are guaranteed to be delivered within a worst-case latency that is defined by the service user. However, in order to stay within this worst-case latency, a real-time TD-CSTS instance can discard some measurements if backpressure in the connection between the service user an service provider makes delivery of all measurements within the specified latency impossible (e.g., if the network that connects the service user and provider becomes congested).

In the complete data delivery mode, the TD-CSTS is guaranteed to deliver all periodic tracking data measurements, but without a guaranteed limit on the latency of their delivery. The complete delivery mode also allows the user to receive periodic tracking data measurements that were received before the TD-CSTS becomes active, either from an earlier time in the executing Service Package or even from a previously-executed Service Package.

NOTE - Even though the complete data delivery mode can be used to deliver the tracking data measurements a significant time after they were observed, the complete data delivery mode still delivers the measurements formatted as though they were to be transferred in real time.

## Functional Description

As defined in the Cross Support Reference Model [2], related cross support services are bundled into Service Packages for the purposes of ensuring that the required relationships among those cross support services are preserved during their production and provision. For example, multiple cross support transfer services might be related to the operation of the same RF link, and the return RF link might be related to the forward RF link. All of those transfer services, as well as the RF links themselves, are treated as a single package for the purposes of scheduling.

Service Packages include functions that produce tracking data measurements and the TD-CSTS instances by which those tracking data measurements are delivered to the user(s) of the tracking services (e.g., the spaceflight mission).

NOTE - There could be methods other than TD-CSTS by which tracking data are delivered to users, for example, via file transfer. Such other methods, if they exist, are outside the scope of this Recommended Standard.

A Service Package can contain multiple instances of the Tracking Data CSTS. Each instance of the TD-CSTS in a Service Package is capable of reporting all tracking data of the types supported by TD-CSTS for that Service Package. For example, if a Service Package includes functions that produce two-way Doppler and angle tracking measurements and two instances of TD-CSTS, each of the TD-CSTS instances has access to the two-way Doppler and angle tracking measurements.

### Service Production

The production of the Tracking Data CSTS instances associated with a given Service Package consists of:

* the functions that generate tracking-related data:
* the TDM Segment Generation function; and
* the TDM Recording Buffer function.

Figure 2‑1is a notional representation of the relationships among the tracking-data-generating functions, the TDM Segment Generation function, the TDM Recording Buffer function, and the Tracking Data CSTS instances (real-time and complete) in the production and provision of Tracking Data CSTS.

The tracking-data-generating functions depicted in Figure 2‑1are Forward Space Link Carrier Transmission, Forward Link Ranging, Return Space Link Carrier Reception, Range and Doppler Extraction, and the Antenna. Operating parameters of and measurements taken by these functions are the basis of the following types of tracking data (see reference [3]):

* Carrier power;
* Carrier power to noise spectral density;
* Doppler (instantaneous);
* Doppler (integrated);
* Range;
* Range power to noise spectral density;
* Receive frequency;
* Transmit frequency;
* Transmit frequency rate; and
* Antenna angles.

NOTE - The functionsillustrated in Figure 2‑1 are notional in the sense that they are abstractions, which in most cases are associated with specific space communication technologies.



Figure 2‑1 : Production and Provision of Tracking Data Cross Support Transfer Services

The TDM Segment Generation function combines the tracking measurements from the tracking-data-generating functions to produce a series of *atomic segments* that contain both the metadata that characterizes the tracking data as well as the tracking data measurements themselves, in conformance with the syntactic and semantic rules of the TDM (reference [3]).

Tracking-data-generating functions s could occur in different combinations and multiplicities within a Service Package. A Service Package that has both S-band and X-band return links operating simultaneously and both being used for Doppler tracking is an example of a Service Package with multiple concurrent sources of Doppler tracking data. The metadata associated with each tracking data measurement provides the information to discriminate among multiple occurrences of the same tracking data type.

NOTE - In the example above, the lines “TRANSMIT\_BAND = S” and RECEIVE\_BAND = S” in the metadata for an atomic segment containing integrated Doppler distinguishes that Doppler measurement from one reported in an atomic segment with the metadata lines “TRANSMIT\_BAND = S” and RECEIVE\_BAND = X”.

The TDM Recording Buffer stores the tracking data segments for subsequent retrieval by instances of the TD-CSTS operating in complete data delivery mode. The TDM Recording Buffer also stores ‘resource status change’ and resource configuration change’ event notifications regarding the status and configuration of the resources that perform the tracking data-related functions, as defined in 4.5.7 of reference [1]. These notifications are stored in the TDM Recording Buffer synchronously with respect to the time at which the atomic TDM segments are stored. The presence of these notifications may assist the analyst to troubleshoot and/or better understand the environment in which the atomic TDM segments were generated.

NOTE - While the TDM Recording Buffer is capable of receiving, storing, and providing ‘resource status change’ and resource configuration change’ event notifications concerning the resources that perform the tracking data-related functions, the availability of such notifications depends on the instrumentation of the real physical resources that provide those functions.

### Service Provision

The TD-CSTS operates in both real-time and complete data delivery modes, The data delivery mode of the TD-CSTS is equal to, and configured through, the data delivery mode of the Buffered Tracking Data Message Delivery procedure (see section 4), which is the prime procedure of the TD-CSTS. The Buffered Tracking Data Message Delivery procedure is derived from the CSTS Buffered Data Delivery Procedure (section 4.5 of reference [1]), from which it inherits the two data delivery modes.

In the real-time data delivery mode, the service transfers tracking data measurements as soon as possible after they are taken. In the complete data delivery mode, the service transfers tracking measurements retrieved from the associated TDM Recording Buffer for any specified period befeor and up until the end of the service instance provision period of the TD-CSTS instance.

Each instance of the TD-CSTS allows the user to select the start and stop times of the tracking data to be delivered.

Each TD-CSTS instance operating in the real-time mode transfers the atomic segments, each of which contains one measurement of one of the selected tracking data types, as they are made available by the TDM Segment Generation function. The user of the service can choose to receive all of the selected atomic segments that are generated between the specified start and stop times for that service instance, or any time-delimited subset thereof.

Each TD-CSTS instance operating in the complete mode retrieves from the TDM Recording Buffer and transfers all atomic segments for the selected tracking data types that have been (or will be) generated by the TDM Segment Generation function between the start and stop generation times specified for that service instance. The start and stop generation times may be any times prior to the end of that service instance provision period for that service instance.

Each TD-CSTS instance allows the user to query the production status of the service and the values of configuration parameters of the procedures that comprise the TD-CSTS.

Each TD-CSTS instance notifies the user whenever the production status changes during the service instance provision period of the TD-CSTS:

* The production status of a real-time TD-CSTS instance is defined by and equal to the resource status of TDM Segment Generation function that generates the TDM atomic segments that are transferred by that TD-CSTS instance.
* The production status of a complete TD-CSTS instance is defined by and equal to the resource status of the resource that performs the TDM Recording Buffer function that provides the TDM atomic segments and resource status and configuration change notifications that are transferred by that TD-CSTS instance.

Each TD-CSTS instance notifies the user whenever the production configuration changes during the service instance provision period of the TD-CSTS:

* The production configuration of a real-time TD-CSTS instance encompasses all resources that perform tracking-data-related functions that provide input that that TD-CSTS instance, including the TDM Segment Generation function that generates the TDM atomic segments that are transferred by that TD-CSTS instance. Thus a production configuration change notification is generated and transferred by the real-time TD-CSTS instance whenever a configuration change occurs in any of those resources.
* The production configuration of a complete TD-CSTS instance is defined by and equal to the resource configuration of the resource that performs the TDM Recording Buffer function that is associated with that complete TD-CSTS instance.

NOTE - As noted under 2.2.1, the TDM Recording Buffer records resource status change and resource configuration change notifications that are generated by (or on behalf of) the individual resources that perform tracking-data-related functions. These notifications are also transferred by a complete TD-CSTS instance, in generation-time-synchronous order with respect to the recorded atomic TDM segments. However, such resource status change and resource configuration change notifications are independent of the production status change and production configuration change notifications that are related to the complete TD-CSTS service instance itself.

## Service ManagEment

Cross support service management both establishes the constraints on the Service Packages to which a given spaceflight mission conforms (e.g., data rate and frequency ranges, types and numbers of cross support transfer service instances) and provides the mechanisms for instantiating conformant Service Packages (e.g., via scheduling).

With regard to the production of tracking data and provision of Tracking Data CSTS instances, cross support service management:

1. Schedules the Service Packages that specify the tracking activities that are to be performed and the TD-CSTS instances that transfer the tracking data to the users of those service instances, and
2. Establishes the types of tracking data that can be reported by each TD CSTS instance during the execution of a Service Package.

A Service Package identifies the various space communication and radiometric functions that are to be performed by a Cross Support Complex during a specified period of time. Typically, the Service Package corresponds to the functions performed at a single ground station for one pass/contact/track. The Service Package also defines – indirectly, through reference to configuration profiles – the configuration parameters that specify the initial configurations of the spce communication and radiometric functions, and the interrelationships among them (e.g., the frame length on each return link symbol stream). With respect to the Tracking Data Service, the Service Package identifies which resources perfrom the functions that produce the radiometric measurements thata are recoported by the Tracking data service, and pairs those resources to the TDM keyword identifiers (see [3]) used in the generation of the atomic segments

The means by which service management schedules the Service Packages is outside the scope of this Recommendation. The CCSDS SCCS-SM suite ([I8] and [I9]) defines a standard set of service management information entities used in the scheduling of Service Packages.

## Cross Support View

Figure 2‑2 shows an example configuration of a Cross Support Complex providing instances of Tracking Data CSTS to a Mission Data Operations System (MDOS). Consistent with Figure 2‑1, the TDM Segment Generation function receives tracking-related measurements from the Return Space Link Carrier Reception function, the Range and Doppler Extraction function, the Forward Space Link Carrier Transmission function, and the Antenna function.

NOTE 1 - The Forward Link Ranging function supplies its ranging timing information to the Range and Doppler Extraction function, which uses that information to compute the range and Doppler measurements that are in turn supplied to the TDM Segment Generation function.

The merged and TDM-segmented tracking data are supplied directly to the real-time TD-CSTS instances. For TD-CSTS instances operating in complete mode, the merged and segmented tracking data are supplied to the TDM Recording Buffer, which makes them available to the complete TD-CSTS instances.

NOTE 2 - For context, Figure 2‑2 also shows that the Return Space Link Carrier Reception and the Forward Space Link Carrier Transmission functions are also involved (along with the Antenna functionl in the production of the Return All Frames (RAF) (reference [I4]) and Forward CLTU (F-CLTU) (reference [I5]) SLE transfer service instances, respectively. However, there are other functions involved in the production of these SLE transfer services that are not illustrated in the figure.

One instance of the TDM Segment Generation function can supply tracking data to multiple instances of real-time TD-CSTS, and one instance of the TDM Recording Buffer can supply tracking data to multiple instances of complete TD-CSTS. One instance of the TDM Segment Generation function might monitor multiple instances of the same kind of functional resource. For example, if a mission spacecraft generates two return space links during a single space link session, (e.g., at S and X band) and one-way Doppler tracking is being performed on each of those links, two instances of the Return Space Link Carrier Reception function will be instantiated during the Service Package for that space link session (one for each of S and X band). The one-way measurements from both links are provided to the same instance of the TDM Segment Generation function. In such a case, one of the responsibilities of the TDM Segment Generation function is to generate appropriate metadata (see reference [3]) to distinguish between the S- and X-band Doppler measurements.



Figure 2‑2 : Example of the Management and Provision of Tracking Data Service Instances for a Service Package

NOTE - Some real Cross Support Complexes might not be implemented in a way that allows all tracking data to be filtered through a single TDM Segment Generation functional resource and/or TDM Recording Buffer. For example, in a particular Complex the antenna angle measurements might not be available for merging with space link frequency data. In such cases the Complex could constrain certain TD-CSTS instances to report only antenna angles and other TD-CSTS instances to report only space link frequency data.

## Operational scenario

This section presents an example operational scenario for the TD-CSTS. It does not cover all possibilities, but rather it attempts to provide an example that illustrates the main operational aspects of the TD-CSTS. This scenario is written in the context of service management operating in accordance with the Extensible SCCS-SM Concept (reference [I7]).

### Service Management/Service Planning activities

As part of Service Management activities that establish the relationship between the spaceflight mission and the Complex, Utilization Management (UM) and Complex Management (CM) negotiate the set of tracking services and associated measurements that will be available to the mission within the context of the Service Agreement. For the purpose of this scenario, the Service Agreement calls for the Complex to provide tracking and communication services on S-Band forward and return links, and tracking measurements consisting of integrated Doppler (range rate), range, and antenna angles. The Service Agreement also specifies how much tracking data can be stored in the TDM Recording Buffer for this mission at each ground station, and the conditions and mechanisms for the purging of that data.

As part of the negotiation process, several Space Link Session (SLS) configuration profiles and several Tracking Data Retrieval configuration profiles are created for use by the mission, Among other things, each SLS configuration profile identifies the resources that are involved in (a) the generation of the Doppler, range, and antenna angle measurements, (b) the generation of atomic TDM segments from those measurements, (c) the recording of those atomic TDM segments for subsequent transfer by complete-mode TD-CST instances, and (d) the tracking data types to be transferred by each real-time TD-CSTS (if any) that will operate during the course of the SLS (also known as a contact or pass). The SLS configuration profile also specifies the mapping between the identification of those resources and the TDM keywords (see [3]) that are used to refer to those resources in the atomic TDM segments.

Each Tracking Data Retrieval configuration profile identifies one or more complete-mode TD CSTS instances associated with one TDM Recording Buffer, and for each TD-CSTS instance specifies which tracking data types are to be transferred by that service instance.

At some time after the start time of the Service Agreement, UM causes CM to create two Trackings Data Retrieval Service Packages (see [I7]), each of which establishes one instance of complete TD-CSTS with access to the tracking data collected for the mission for the remaining lifetime of the Service Agreement. Complete TD-CSTS instance one (TD-1) is configured to transfer Doppler and range measurements. The user of TD-1 is a Mission flight operations function within theMDOS. Complete TD-CSTS instance two (TD-2) is configured to transfer Doppler, range, and antenna angle measurements.The user of TD-2 is the flight dynamics facility that maintains the precision orbit of the spacecraft on behalf of the Mission.

At some (different) time after the start time of the Service Agreement, UM causes CM to create an SLS Service Package (see reference [I7]) that specifies an SLS during which (among other things) Doppler and range measurements are to be taken using the S-Band forward and return links. The SLS Service Package includes an instance of real-time TD-CSTS (TD-3) (where the user of TD-3 is a Mission flight operations function. The SLS Service Package configures TD-3 to transfer Doppler, range, and antenna angle measurements and specifies the *latency-limit* value for TD-3.

### SLS Service Package Execution

#### Production of TDM Segments During the SLS

At the scheduled start time of the SLS Service Package, the Complex establishes the space links with the spacecraft and begins processing of the signals to and from the spacecraft. When each of the resources that generate or process tracking-related data (including the TDM Segment Generation Function) becomes operational, a time-stamped resource status change event notification, indicating the transition to ‘operational’, is placed into the TDM Recording Buffer for that resource. In addition, when the TDM Segment Generation Function) becomes operational, the real-time TD-CSTS instance TD-3 generates and transfers a ‘production status change’ event notification indication the transition of the production status to ‘operational’.

Thereafter, the antenna angle, Doppler, and range measurements are reported to the TDM Segment Generation function. For each single Doppler measurement, single range measurement, and antenna angle pair of measurements, the TDM Segment Generation function uses the mapping information in the applied configuration profile to generates an atomic segment that contains the tracking data measurement and the metadata that characterizes that tracking data in terms of the appropriate TDM keywords. The resultant atomic segments are made available to any and all real-time TD-CSTS instances that are active during the space link session. The atomic segments are also stored by the TDM Recording Buffer for retrieval and transfer by complete TD-CSTS instances.

#### Binding of Complete TD-CSTS Instance 1 (TD-1)

At some time before the scheduled start time of the SLS Service Package, the user of the complete TD-CSTS instance TD-1 invokes the BIND operation to bind to the service provider and transitions the service instance to the ‘bound.ready’ state.

#### TD-1 Buffered Tracking Data Message Delivery Procedure Prime Instance

Following the binding of the service instance, the user of TD-1 invokes the START operation of the prime instance of the Buffered Tracking Data Message Delivery (BTDMD) procedure. The START invocation has the **start-generation-time** set and **stop-generation-time** parameters set equal to the scheduled start and stop times (respectively) of the SLS Service Package, indicating that the service instance is to transfer all Doppler and range tracking data acquired during the execution of the SLS Service Package. TD-1 transitions to the ‘bound.active’ state and the Buffered Tracking Data Message Delivery instance activates, generates the TDM Header, and transfers the TDM Header to the user in the positive return of the START operation.

TD-1 then retrieves from the TDM Recording Buffer the first ‘resource status change’ event notification that was put into the TDM Recording Buffer after **start-generation-time**, places it in its Return Buffer, and initiates the release timer. The additional ‘resource status change’ notifications are subsequently retrieved and put into the Return Buffer. If any new atomic segments containing one of the selected tracking data types (Doppler and range for TD-1) is made available by the TDM Recording Buffer before the expiration of the release timer, TD-1 places them in its Return Buffer. When the Return Buffer fills or the release timer expires, TD-1 transfers the atomic segments and event norifications in the Return Buffer to the service user.

Subsequently, whenever a new atomic segment containing one of the selected tracking data types is made available by the TDM Recording Buffer, TD-1 places it in its Return Buffer and initiates the release timer if the atomic segment is the first one following transmission of the previous Return Buffer.

NOTE 1 - The user of TD-1 is not constrained as to when to start the service instance with respect to the execution of the SLS Service Package: as a complete service instance, TD-1 will deliver all designated tracking data for the period between the **start-generation-time** set and **stop-generation-time** even if the service instance is started after the start time of the SLS Service Package. However, starting the service instance at or before the start time of the SLS Service Package ensures that the tracking data measurements are transferred as soon as possible.

#### TD-1 Information Query Procedure

At some time following the activation of the BTDMD procedure, the user invokes the GET operation of the Information Query procedure to query the values of the parameters in the default parameter label list. The list-of-parameters parameter of the GET invocation is left empty to indicate the query of the default list. The default parameter list for the TD-CSTS is defined by the service specification to contain all of the configuration parameters of the Association Control and BTDMD procedures. The GET return contains the qualified parameters values for the following set of configuration parameter names:

* Service user responding timer (Association Control);
* Initiator ID (Association Control);
* Responder ID (Association Control);
* Responder Port ID (Association Control);
* Service Instance ID (Association Control);
* Return buffer size (Buffered Tracking Data Message Delivery);
* Delivery mode (Buffered Tracking Data Message Delivery).

#### Binding of Real-Time TD-CSTS Instance 3 (TD-3)

At the scheduled start of the service instance provision period of the real-time TD-CSTS instance (TD-3) within the SLS Service Package, the service instance exists in the ‘unbound’ state. Shortly after the scheduled start time of TD-3, the user of that service instance invokes the BIND operation to bind to the service provider and transitions the service instance to the ‘bound.ready’ state.

#### TD-3 Buffered Tracking Data Message Delivery Procedure Prime Instance

The user of TD-3 then invokes the START operation for the BTDMD procedure. The TD-3 START invocation has ‘null’ values for the **start-generation-time** and **stop-generation-time** parameters, indicating that the service instance is to begin sending atomic segments and service production change notifications as soon as any are available, and continue sending atomic segments and service production change notifications until the user invokes the STOP operation. TD-3 transitions to the ‘bound.active’ state and the BTDMD procedure activates, generates the TDM Header, and transfers the TDM Header to the user in the positive return of the START operation.

If the service instance provision period of TD-3 begins with sufficient lead time to allow TD-3 to be bound and started before the TDM Segment Generation function becomes operational, the TD-3 instance generates a ‘production status change’ event notification when the resource performing the TDM Segment Generation function becomes operational, places it in the Return Buffer, and initiates the release timer. If TD-3 is not started until after the resource performing the TDM Segment Generation function has become operational, whenever the first atomic segment containing one of the subscribed data types is made available by the TDM Segment Generation function, the BTDMD procedure places it in its Return Buffer and initiates the release timer. Subsequently, whenever a new atomic segment containing one of the subscribed data types is made available by the TDM Segment Generation function (or the production status changes), the BTDMD procedure places the atomic segment/event notification in its Return Buffer. When the Return Buffer fills or the release timer expires, TD-3 transfers the atomic segments and event notifications in the Return Buffer to the service user.

NOTE 2 - In order to minimize the latency of transfer of the atomic segments, the Return Buffer latency timer value will normally be configured to be on the order of the shortest sampling period of all of the tracking data sources for the TD-CSTS instance.

#### Backpressure in the Connection between TD Service Provider and User

Partially through the space link session, the communication service underlying TD-1 and TD-3 begins to experience congestion to the point that the resulting backpressure prohibits all of the tracking data from being transferred across the reliable connection in a timely manner. In order to maintain the timeliness of the tracking data measurements, TD-3 discards “stale” atomic segments and event notifications in order to ensure that the latest measurements are the ones that are delivered. After the backpressure clears, the transfer of timely atomic segments and event otifications resumes, along with a notification that some data has been discarded.

In contrast, TD-1, being a complete TD-CSTS instance, does not discard any atomic segments or notifications but rather continues to attempt to send all recorded segments and notifications regardless of the delay. After the backpressure clears, the backlogged atomic segments and notifications are transferred at the maximum rate permitted by the underlying communication service and the performance of the TD-1 user entity.

NOTE 3 - In many cases the nominal data rate of the underlying communication service will be at least several times that needed to transfer the tracking data. In these cases, even if backpressure temporarily slows the transfer of the atomic segments, a complete TD-CSTS instance is likely to quickly “catch up” once the backpressure has cleared.

#### Stopping and Unbinding of TD-3

Shortly before the scheduled stop time of TD-3, the user invokes the STOP operation of the BTDMD procedure to cease transfer of the tracking data measurements. The BTDMD procedure deactivates and TD-3 transitions to the ‘bound.ready’ state. The user then invokes the UNBIND operation, which causes TD-3 to transition to the ‘unbound’ state.

#### Stopping and Unbinding of TD-1

At the specified stop generation time for complete TD-CSTS instance TD-1, the BTDMD procedure of TD-1 generates and sends a notification that the requested data has reached its end. The user invokes the STOP operation of the BTDMD procedure to transition the procedure to the inactive state. The BTDMD procedure deactivates and TD-1 transitions to the ‘bound.ready’ state. The user then invokes the UNBIND operation, which causes TD-2 to transition to the ‘unbound’ state.

### PosT-Space Link Session

#### Retrieval of TDM Segments after Completion of the SLS

Following the execution of the SLS Service Package, the antenna angle, Doppler, and range measurements for the SLS have been processed by the TDM Segment Generation function and stored by the TDM Recording Buffer for retrieval and transfer by complete TD-CSTS instances.

#### Binding of Complete TD-CSTS Instance 2 (TD-2)

At some time after the stop time of the SLS Service Package, the user of complete TD-CSTS instance TD-2 invokes the BIND operation to bind to the service provider and transitions the service instance to the ‘bound.ready’ state.

#### TD-2 Buffered Tracking Data Message Delivery Procedure Prime Instance

The user of TD-2 then invokes the START operation of the BTDMD procedure. The START invocation has values for the **start-generation-time** and **stop-generation-time** parameters that are within the lifetime of the SLS Service Package described above. TD-2 transitions to the ‘bound.active’ state and the BTDMD procedure activates, generates the TDM Header and transfers the TDM Header to the user in the positive return of the START operation. The BTDMD procedure then retrieves (from the TDM Recording Buffer) and transfers (a) the atomic segments that contain the Doppler, range, and antenna angle measurements and that fall within the start-generation-time/stop-generation-time interval, and (b) recorded event notifications that were generated within the start-generation-time/stop-generation-time interval.

NOTE 4 - The setting of the start and stop generation times to be within the execution time of the SLS Service Package is arbitrary. A complete TD-CSTS instance has access to all tracking data in the TDM Recording Buffer with which it is associated, regardless of when (i.e., during which space link session) the data is captured.

#### Stopping and Unbinding of TD-2

At the specified stop generation time of TD-2, the BTDMD procedure of TD-2 generates and sends a notification that the requested data has reached its end. The user subsequently invokes the STOP operation to cease transfer of the tracking data measurements. The BTDMD procedure deactivates and TD-2 transitions to the ‘bound.ready’ state. The user then invokes the UNBIND operation, which causes TD-2 to transition to the ‘unbound’ state.

Because TD-2 is a complete delivery mode TD-CSTS instance, TD-2 delivers all recorded tracking data of the selected types and event notifications that were originally generated during the period bounded by the start and stop generation times, even if the connection between the provider and user of TD-2 experiences backpressure when TD-2 is active.

# Tracking Data Cross Support Transfer Service Composition

## DISCUSSION

The Tracking Data CSTS may be implemented as defined herein without need for further extension or refinement.

The service-level object identifiers for the Tracking Data CSTS are specified in ANNEX B.

## Procedures of the Tracking Data Cross Support Transfer Service

The Tracking Data transfer service shall be composed of the Association Control, Buffered Tracking Data Message Delivery, and Information Query procedures.

The Association Control procedure shall conform to the specification of the CSTS Specification Framework (CSTS SFW, reference [1]) Association Control procedure (section 4.3) without derivation.

There shall be one and only one instance of the Association Control procedure.

The Buffered Tracking Data Message Delivery procedure shall be derived from the CSTS SFW Buffered Data Delivery procedure as specified in 4.

The Buffered Tracking Data Message Delivery procedure shall be the primary procedure for the Tracking Data service.

There shall be one and only one instance of the Buffered Tracking Data Message Delivery procedure.

The version number of the Buffered Tracking Data Message Delivery procedure is 1.

The Information Query procedure shall be adopted directly from the Information Query procedure defined in section 4.9 of reference [1].

There shall be one secondary procedure instance of the Information Query procedure.

NOTE - Table 3‑1 summarizes the procedures that comprise the Tracking Data transfer service, where (a) the “[P]” designate Buffered Tracking Data Message Delivery as the primary procedure; (b) *Version* = “-” indicates that the version of the service procedure is the same as that of the CSTS SFW procedure for the procedure that is directly adopted (Association Control), and *Version* = “1” indicates the version of the refined and/or extended service procedures (Buffered Tracking Data Message Delivery and Information Query); (c) *No. of Instances* indicates the minimum and maximum number of allowed instances of each procedure type; (d) *Specification Approach* indicates which procedures are directly adopted or refined and extended; and I *Source* indicates the CSTS SFW procedure from which the service procedure is adopted, refined and/or extended.

Table 3‑1: Procedures of the Tracking Data CSTS

|  |  |  |  |
| --- | --- | --- | --- |
| Procedure | Association Control | Buffered Tracking Data Message Delivery [P] | Information Query |
| Version | - | 1 | - |
| No. of Instances | 1..1 | 1..1 | 1..1 |
| Specification Approach | adopted | refined-and-extended | adopted |
| Source | CSTS SFW [1]: Association Control (section 4.3) | CSTS SFW [1]: Buffered Data Delivery (section 4.5) | CSTS SFW [1]: Information Query (section 4.9) |

## Tracking Data Cross Support Transfer Service State Machine

The Tracking Data Cross Support Transfer Service state machine conforms to the state machine for a CSTS with a stateful prime procedure instance, as defined in the CSTS Framework (F3 of reference [1]).

# Buffered Tracking Data Message Delivery Procedure

## Discussion

### Purpose

The Buffered Tracking Data Message Delivery procedure supports transfer of CCSDS Tracking Data Messages [3], structured into data units from the Provider to the User using one of the real-time or complete delivery modes, as defined in [1].

### Concept

The Buffered Tracking Data Message Delivery (BTDMD) procedure is derived from the CSTS SFW Buffered Data Delivery procedure so that it conveys the tracking data in a way that satisfies the syntactic and semantic requirements for valid CCSDS Tracking Data Messages.

a) The contents of the data parameter of the TransferDataInvocations are refined to be TDM Atomic Segments, as defined in Annex A of this specification;

b) The START positive return is extended to include a TDM Header, as defined in Annex A of this specification.

A BTDMD procedure instance can deliver one or more tracking data types, as configured by Service Management. For each tracking data type that is specified for a BTDMD instance, the BTDMD delivers the TDM Atomic Segments for all instances of that tracking data type that are generated by the tracking-related resources that are associated with the TD-CSTS instance that executes the BTDMD procedure. The association of TD-CSTS instances with specific tracking-related resources is a function of Service Management.

NOTE - The parent CSTS SFW Buffered Data Delivery procedure specifies that the START operation includes selection criteria for the selection of the TransferDataInvocations to be delivered. For the case of the BTDMD procedure, the selection criteria applied in the START operation are limited to the start and stop generation time parameters specified in the parent CSTS SFW Buffered Data Delivery procedure. Selection of the types of tracking data to be delivered is configured in the Service Package prior to the start of the TD-CSTS service instance provision period and applies for the duration of the service instance provision period.

## Procedure Type Identifier

The procedure identifier buffTrkDataMsgDel, as specified in ANNEX B, shall be used for this procedure.

## Extension

The Buffered Tracking Data Message Delivery extends the Buffered Data Delivery procedure by modification of the behavior of the procedure, addition of a parameter to the START operation, and refinement of a parameter of the TRANSFER-DATA operation.

## Behavior

The overall activities of the Buffered Tracking Data Message Delivery procedure are the same as those of the standard Buffered Data Delivery procedure as defined in the CSTS Framework (reference [1]). The detailed behavior of the Buffered Tracking Data Message Delivery procedure with respect to starting, transferring data and notifications, stopping, and aborting is the same as that of the standard Buffered Data Delivery procedure as defined in [1], with the exceptions to the Starting behavior specified in 4.4.1 and the Transferring Data and Notifications behavior defined in 4.4.2.

### DeriVED Behavior - STARTING

In successfully performing the START operation, the service provider shall return a result that contains a TDM Header (see annex A) in addition to the data contained in the Buffered Data Delivery START positive return.

### DeriVED Behavior – Transferring Data and Notifications

In addition to the start and end generation times selection criteria specified in the START invocation, the selection of data to be delivered by the TRANSFER-DATA invocations shall be further limited to the Atomic Segments (see 4.5.7.1.1.1) that contain tracking data that conform to the tracking-data-types configuration parameter.

The tracking-data-types configuration parameter shall specify one or more of the following types:

* Doppler (instantaneous);
* Doppler (integrated);
* Range;
* Carrier power;
* Carrier power to noise spectral density;
* Ranging power to noise spectral density;
* Receive frequency;
* Transmit frequency;
* Transmit frequency rate;
* a pair of antenna angle Tracking Data Records.

NOTE - The capabilities of individual tracking service providers may restrict the types of tracking data that are available from those providers.

## Required Operations

The Buffered Tracking Data Message Delivery procedure shall use the STOP and NOTIFY operations of the CSTS SFW Buffered Data Delivery procedure without extension or refinement.

The Buffered Tracking Data Message Delivery procedure shall extend the START operation of the CSTS SFW Buffered Data Delivery procedure as specified in 4.5.6.

The Buffered Tracking Data Message Delivery procedure shall refine the TRANSFER-DATA operation of the CSTS SFW Buffered Data Delivery procedure as specified in 4.5.7.

The START and STOP operations of the Buffered Tracking Data Message Delivery procedure shall be Blocking, as defined in for their parent operations in the CSTS SFW Buffered Data Delivery procedure.

The TRANSFER-DATA and NOTIFY operations of the Buffered Tracking Data Message Delivery procedure shall be Non-Blocking, as defined for their parent operations in the CSTS SFW Buffered Data Delivery procedure.

NOTE - Table 4‑1 summarizes the operations of the Buffered Tracking Data Message Delivery procedure.

Table 4‑1 : Buffered Tracking Data Message Delivery Required Operations

|  |  |  |  |
| --- | --- | --- | --- |
| **Operations** | **Extended** | **Refined** | **Procedure Blocking/Non-Blocking** |
| START | Y | N | Blocking |
| STOP | N | N | Blocking |
| TRANSFER-DATA | N | Y | Non Blocking |
| NOTIFY | N | N | Non Blocking |

### START (confirmed)

#### Invocation, Return and Parameters

In addition to the parameters of the START invocation and return for the Buffered Data Delivery procedure as defined in [1], the extension parameter specified in Table 4‑2shall be present in the START return of the Buffered Tracking Data Message Delivery procedure.

Table 4‑2 : START Extension Parameter

|  |  |  |
| --- | --- | --- |
| **Extension Parameters** | **Invocation** | **Return** |
| tdm-header |  | C |

##### Extension parameter syntax

The type BuffTrkDataDelStartPosReturnExt, as defined in ANNEX C, shall define the syntax of the extension parameter of the START positive return.

##### tdm-header

The tdm-header parameter shall contain a TDM Header, as specified in Annex A.

The tdm-header parameter shall be present in the return if and only if the value of the **result** parameter is ‘positive result’ (see the result parameter of the Standard Operation Header in [1]).

### Transfer-Data (Unconfirmed)

#### Invocation and Parameters

##### data parameter resolution

The **data** parameter shall be of type octet string, formatted as an Atomic Segment as defined in annex A.

## Configuration Parameters

Table 4‑3 defines the configuration parameters of the Buffered Tracking Data Message Delivery procedure that need to be configured in the context of this procedure. For each configuration parameter, the table provides the engineering unit (if applicable), a cross reference to the use of the parameter in the specification of the procedure, identifies whether the parameter may be read and/or dynamically modified, and also identifies the Parameter Identifier to be used in reporting the value of the parameter.

Table 4‑3 : Buffered Tracking Data Message Delivery Configuration Parameters

| Parameters | Cross-Reference | Readable | Dynamically modifiable | ConfigurationParameter Identifier and Type  |
| --- | --- | --- | --- | --- |
| return-buffer-size (in number of TRANSFER-DATA and/or NOTIFY invocations the buffer will accommodate) | CSTS SFW 4.5.3.2.7.3 | Yes | Yes | pBDDreturnBufferSize PBDDreturnBufferSizeType(CSTS FSW E3.16) |
| delivery-latency-limit (in seconds) | CSTS SFW 4.5.3.2.7.2 | Yes | Yes | pBDDdeliveryLatencyLimit PBDDdeliveryLatencyLimitType(CSTS FSW E3.16) |
| delivery-mode | CSTS SFW 4.5.2.2.2.1, 4.5.3.2.6 | Yes | No | pBDDdeliveryMode PBDDdeliveryModeType (CSTS FSW E3.16) |
| tracking-data-types | 4.4.2.2 | Yes | No | PBTDMDtrackingDataTypes(ANNEX D) |

## Procedure STATE TABLE (Provider Side)

The state table for the Buffered Tracking Data Message Delivery procedure is the same as that for the Buffered Data Delivery procedure as specified in reference [1]).

# Setting of Configuration Parameters Inherited from Framewwork Procedures

## General

The procedures of the CSTS SFW define configuration parameters for the Framework procedures, but defers to the derived services the specification of the method by which each of those configuration parameters is to be set. This section specifies the method by which each of the Framework procedure configuration parameters is to be set for the TD-CSTS.

## Association Control Procedure Configuration Parameters

The service-user-responding-timer (reference [1], table 4-2) shall be configured by a service management parameter with the classifier tdServiceUserRespondingTimer.

The initiator-identifier (reference [1], table 4-2) shall be configured by a service management parameter with the classifier tdInitiatorId.

The responder-identifier (reference [1], table 4-2) shall be configured by a service management parameter with the classifier tdResponderId.

The responder-port-identifier (reference [1], table 4-2) shall be configured by a service management parameter with the classifier tdResponderPortId.

The service-instance-identifier (reference [1], table 4-2) shall beconfigured by a service management parameter with the classifier tdServiceInstanceId.

## Buffered Tracking Data Message Delivery Procedure Configuration Parameters

NOTE - The CSTS SFW defers the setting of the delivery-mode, delivery-latency-limit, and transfer-buffer-size configuration parameters to the service that uses or derives the Buffered Tracking Data Message Delivery procedure.

The delivery-mode (see reference [1], table 4-16) shall be configured by a service managementparameter with the classifier tdDeliveryMode

The treturn-buffer-size (see reference [1], table 4-16) shall be configured by a service management parameter with the classifier tdReturnBufferSize.

The delivery-latency-limit (see reference [1], table 4-16) shall be configured by a service management parameter with the classifier tdDeliveryLatencyLimit.

## Information Query Procedure Configuration Parameters

NOTE - The CSTS SFW defers the setting of the default list of parameters and named label lists configuration parameters to the service that uses the Information Query procedure. The Tracking Data CSTS defines the contents and name of the default list of parameter labels.

The default list of parameters (reference [1], table 4-52) shall be named “defaultParameterList” and contain the following Parameter Labels:

* [associationControl : pACserviceUserRespTimer], where associationControl is the Procedure Identifier of the Framework Association Control procedure as defined in the CCSDS-CSTS-OBJECT-IDENTIFIERS module (E3.1 in reference [1]), and pACserviceUserRespTimer is the Parameter Identifier of the service-user-responding-timer configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]);
* [associationControl : pACinitiatorId], where associationControl is the Procedure Identifier of the Framework Association Control procedure as defined in the CCSDS-CSTS-OBJECT-IDENTIFIERS module in (E3.1 reference [1]), and pACinitiatorId is the Parameter Identifier of the initiator-identifier configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]);
* [associationControl : pACresponderId], where associationControl is the Procedure Identifier of the Framework Association Control procedure as defined in the CCSDS-CSTS-OBJECT-IDENTIFIERS module (E3.1 in reference [1]), and pACresponderId is the Parameter Identifier of the responder-identifier configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]);
* [associationControl : pACresponderPortId], where associationControl is the Procedure Identifier of the Framework Association Control procedure as defined in the CCSDS-CSTS-OBJECT-IDENTIFIERS module (E3.1 in reference [1]), and pACresponderPortId is the Parameter Identifier of the responder-port-identifier configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]);
* [associationControl : pACserviceInstanceId], where associationControl is the Procedure Identifier of the Framework Association Control procedure as defined in the CCSDS-CSTS-OBJECT-IDENTIFIERS module (E3.1 in reference [1]), and pACserviceInstanceId is the Parameter Identifier of the service-instance-identifier configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]);
* [buffTrkDataMsgDel : pBDDreturnBufferSize], where buffTrkDataMsgDel is the Procedure Identifier of the TD-CSTS Buffered Tracking Data Message Delivery procedure as defined in ANNEX B, and pBDDreturnBufferSize is the Parameter Identifier of the return-buffer-size configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]);
* [buffTrkDataMsgDel : pBDDrecordingBufferSize], where buffTrkDataMsgDel is the Procedure Identifier of the TD-CSTS Buffered Tracking Data Message Delivery procedure as defined in ANNEX B, and pBDDrecordingBufferSize is the Parameter Identifier of the recording-buffer-size configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]); and
* [buffTrkDataMsgDel : pBDDdeliveryMode], where buffTrkDataMsgDel is the Procedure Identifier of the TD-CSTS Buffered Tracking Data Message Delivery procedure as defined in ANNEX B, and pBDDdeliveryMode is the Parameter Identifier of the delivery-mode configuration parameter as defined in the CCSDS-CSTS-FW-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES module (E3.16 in reference [1]);

The named label lists (table 5-52 in reference [1]) shall be a service management parameter.

# Tracking Data Service-Specific Versions of Service-Generic Parameter and Events

## General

Annex A of the CSTS SFW specifies the following service-generic parameters and events for use by any CSTS:

1. A production status that can be monitored. The OID to be use for the parameter that contains the production status for every CSTS is specified in Annex E3.17 of the CSTS SFW with the classifier svcProductionStatusVersion1.
2. A production status change event that is to be emitted when the production status changes, as specified in 3.11.2.2.3.2 (a) of the CSTS SFW. The OID to be use for the production status change event for every CSTS is specified in Annex E3.17 of the CSTS SFW with the classifier svcProductionStatusChangeVersion1.
3. A production configuration change event that is to be emitted when any functional resource in the production experiences a configuration change, as specified in 3.11.2.2.3.2 (b) of the CSTS SFW. The OID to be use for the production configuration change event for every CSTS is specified in Annex E3.17 of the CSTS SFW with the classifier svcProductionConfigurationChangeVersion1.

Each CSTS is to provide its own label for the production status parameter, production status change event, and production configuration.

The Tracking Data service supports the production status parameter and the production status change and production configuration change events.

## tdSvcProductionStatus Parameter

The tdSvcProductionStatus parameter shall contain the production status, with the Published Identifier svcProductionStatusVersion1 as specified in the CCSDS-CSTS-GENERIC-SERVICE-OBJECT-IDENTIFIERS module specified in the CSTS SFW (E3.17 in reference [1]).

## tdSvcProductionStatusChange Parameter

The tdSvcProductionStatusChange event shall report production status changes, with the Published Identifier svcProductionStatusChangeVersion1 as specified in the CCSDS-CSTS-GENERIC-SERVICE-OBJECT-IDENTIFIERS module specified in the CSTS SFW (E3.17 in reference [1]).

## tdSvcProductionConfigurationChange Parameter

The tdSvcProductionConfigurationEvent event shall report production configuration changes, with the Published Identifier svcProductionConfigurationChangeVersion1 as specified in the CCSDS-CSTS-GENERIC-SERVICE-OBJECT-IDENTIFIERS module specified in the CSTS SFW (E3.17 in reference [1]).

# Refinement of definitions of Framework Parameters, events, directives, and Diagnostic Values Used by the Tracking Data Service

## General

Except where explicitly refined in this section, the definitions of the parameters, events, directives, and diagnostic values of the operations of the Framework procedures that are used by the Tracking Data service are the same as their definitions in the CSTS SFW.

## tdSvcProductionStatus Parameter Definition Refinement

NOTE - This refined definition applies to the tdSvcProductionStatus parameter, which has the Published Identifier svcProductionStatusVersion1 in the CCSDS-CSTS-GENERIC-SERVICE-OBJECT-IDENTIFIERS module specified in the CSTS SFW (E3.17 in reference [1]).

### Production Configured

For a TD-CSTS operating in real-time mode, the definition of the ‘production configured’ value of the tdSvcProductionStatus parameter shall be refined to mean that configuration of the resource performing the TDM Segment Generation function has been completed.

For a TD-CSTS operating in complete mode, the definition of the ‘production configured’ value of the tdSvcProductionStatus parameter shall be refined to mean that configuration of the resource performing the TDM Recording Buffer function has been completed.

### Production Interrupted

For a TD-CSTS operating in real-time mode, the definition of the ‘production interrupted’ value of the tdSvcProductionStatus parameter shall be refined to mean that the resource performing the TDM Segment Generation function has been stopped because of a condition that may be temporary.

For a TD-CSTS operating in complete mode, the definition of the ‘production interrupted’ value of the tdSvcProductionStatus parameter shall be refined to mean that the resource performing the TDM Recording Buffer function has been stopped because of a condition that may be temporary.

### Production Halted

For a TD-CSTS operating in real-time mode, the definition of the ‘production halted’ value of the tdSvcProductionStatus parameter shall be refined to mean that the resource performing the TDM Segment Generation function has been stopped by management action.

For a TD-CSTS operating in real-time mode, the definition of the ‘production halted’ value of the tdSvcProductionStatus parameter shall be refined to mean that the resource performing the TDM Recording Buffer function has been stopped by management action.

### Production Operational

For a TD-CSTS operating in real-time mode, the definition of the ‘production operational’ value of the tdSvcProductionStatus parameter shall be refined to mean that the resource performing the TDM Segment Generation function has changed to ‘operational’.

For a TD-CSTS operating in real-time mode, the definition of the ‘production operational’ value of the tdSvcProductionStatus parameter shall be refined to mean that the resource performing the TDM Recording Buffer function has changed to ‘operational’.

## tdSvcProductionStatusChange Event Definition Refinement

NOTE - This refined definition applies to the tdSvcProductionStatusChange event, which has the Published Identifier svcProductionStatusChangeVersion1 in the CCSDS-CSTS-GENERIC-SERVICE-OBJECT-IDENTIFIERS module specified in the CSTS SFW (E3.17 in reference [1]).

For a TD-CSTS operating in real-time mode, the definition of the event-value of the tdSvcProductionStatusChange event shall be refined to mean that the change refers to the resource performing the TDM Segment Generation function.

 For a TD-CSTS operating in complete mode, the definition of the event-value of the tdSvcProductionStatusChange event shall be refined to mean that the change refers to the resource performing the TDM Recording Buffer function.

1. SPecification of TD-CSTS TDM Components

(normative)
	1. General

This section defines the contents of the TDM Header and TDM Atomic Segment.

* 1. TDM Header

The format and semantic content of the TDM Header shall conform to the TDM Header as specified in the TDM Header section of [3]. Figure A‑1 is an example of the TDM Header. See reference [3] for the definitions of the TDM Header keywords.

|  |
| --- |
| CCSDS\_TDM\_VERS = 0.12.tmurCREATION\_DATE = 2007-075T11:42:28.000ORIGINATOR = JPL |

**Figure A‑1: Example TDM Header**

* 1. TDM Atomic Segment

The TDM Atomic Segment shall comprise a TDM Metadata Section, as specified in the TDM Metadata section of [3], followed by a TDM Data Section, as defined in the TDM Data Section (General Specification) and TDM Section Keywords sections of [3], with the constraint that a TDM Data Section shall contain either:

1. A single Tracking Data Record for one of the following tracking data types:
2. Doppler (instantaneous);
3. Doppler (integrated);
4. Range;
5. Carrier power;
6. Carrier power to noise spectral density;
7. Ranging power to noise spectral density;
8. Receive frequency;
9. Transmit frequency;
10. Transmit frequency rate;

or

1. a pair of antenna angle Tracking Data Records.

The format and semantic content of the TDM Metadata Section of the TDM Metadata Partition shall conform to those specified in the TDM Metadata section of [3].

The format and semantic content of the Tracking Data Records of the TDM Data Partition shall conform to those specified in the TDM Data Section (General Specification) and TDM Section Keywords sections of [3].

Figure A‑2 contains two example TDM Atomic Segments for the TDM with the TDM Header presented in Figure A‑1. Note that the line separating the two segments is not part of either segment. See reference [3] for the definitions of the TDM Metadata and TDM Data keywords.

|  |
| --- |
| META\_STARTTIME\_SYSTEM = UTCSTART\_TIME = 2007-075T11:50:43.000STOP\_TIME = 2007-075T11:50:43.000PARTICIPANT\_1 = ‘DSS 26’MODE = N/ATRANSMIT\_BAND = XMETA\_STOPDATA\_STARTTRANSMIT\_FREQ\_1 = 2007-075T11:50:43.000 7175510611.700343DATA\_STOPMETA\_STARTTIME\_SYSTEM = UTCSTART\_TIME = 2007-075T11:50:43.000STOP\_TIME = 2007-075T11:50:43.000PARTICIPANT\_1 = ‘DSS 26’MODE = N/ATRANSMIT\_BAND = XMETA\_STOPDATA\_STARTTRANSMIT\_FREQ\_RATE\_1 = 2007-075T11:50:43.000 0.0DATA\_STOP |

**Figure A‑2: Example TDM Atomic Segments**

* 1. Discussion: IllustrationS of nominal TD-CSTS data transformations

This subsection provides illustrated examples of TD-CSTS data transformations in three cases: (1) a TD-CSTS instance operating in real-time delivery mode, with user-specified start and stop generation times, (2) a TD-CSTS instance operating in real-time delivery mode, with no specified start and stop generation times, and (3) a TD-CSTS instance operating in complete data mode.

* + 1. Real-time Delivery MoDe TD-CSTS with User-SPecified Start and Stop Generation TimeS

Figure A‑3 illustrates some of the data transformations from the TDM Segment Generation Function, through the TD-CSTS operations to the resulting Tracking Data Message document for this case.

The production of atomic segments by the TDM Segment Generation function starts at the beginning of tracking services, which, as illustrated in the figure, may precede the START of a TD-CSTS instance. In this example, the START invocation specifies a start-generation-time that occurs sometime after the START operation is invoked. Therefore, the TD-CSTS instance sends the START return and then waits until the specified start-generation-time before beginning to transfer atomic segments produced as of the START operation and ignores any that occurred before.

NOTE - In this example case, explicit values are provided in the START invocation for both the start-generation-time and the stop-generation-time, but in general whether or not a parameter contains a specified time value is independent of the setting of the other.

The TD-CSTS instance transfers only those atomic segments that each contain one of the tracking data types selected as part of the managed configuration of the TD-CSTS instance. As shown in the figure, atomic segments that do not meet the selection criteria are ignored by the TD-CSTS instance and are not included in the TDM that is ultimately created from the atomic segments that are transferred by the TD-CSTS instance.

The TD-CSTS instance may also generate tdSvcProductionStatusChange event notifications if the production status – which for real-time TD service instances is defined as the status of the TDM Segment Generation function - changes. Figure A‑3 illustrates the generation of two such notifications, a tdSvcProductionStatusChange event with value ‘interrupted’ followed (after some time) by a tdSvcProductionStatusChange notification with value ‘operational’. As defined in reference [1], these notifications are *discardable*, meaning that a TD-CSTS in real-time data delivery mode may not deliver these notifications if the connection experiences backpressure (a TD-CSTS in complete delivery mode delivers all notifications, whether they are discardable or non-discardable). However, even when these notifications are transferred, they are not included in the resulting TDM – they are provided to the TD-CSTS User entity for purposes of monitoring the execution of the TD-CSTS. Note that in real-time mode, TDM Segment Generation generates atomic segments only when it is operational – the gap in Figure A‑3 between segments N+& and M represents lost tracking data.

A TD-CSTS provider instance may also inject notifications regarding the execution of the transfer service itself (e.g., ‘data discarded due to excess backlog’) into the transfer. The example shows the discarding of TRANSFER-DATA invocations due to backpressure. Once the backpressure clears tracking data begins to flow again, and the notification ‘data discarded due to excess backlog’ is also sent. As with the production-generated notifications, these provider-generated notifications do not affect the content of the resultant TDM.

Finally, when the user-specified stop-generation-time is reached, the TD service provider injects an end-of-data notification. The ‘end of data’ notification (which is non-discardable) is used by the TD-CSTS user to identify the end of the TDM.

The net result is a TDM with atomic segments for the time from the start-generation-time through the stop-generation-time, minus segments corresponding to the period of time where the production of tracking data was interrupted and the period of time during which network backpressure cause the TD service to discard segments.

* + 1. Real-time Delivery Mode TD-CSTS with UnSPecified Stop-Generation-Time

Figure A‑4 illustrates some of the data transformations from Tracking Data Production, through the TD-CSTS operations to the resulting Tracking Data Message document for the case where both the start-generation-time and the stop-generation-time are left unspecified in the START invocation. Figure A‑4 is identical to Figure A‑3 except for two differences. The first difference is that at the beginning of the TD service, the transfer of atomic segments begins as soon as possible after the START operation is performed. The second difference is that the user signals the end of the TDM by sending the STOP invocation.

The net result is a TDM with atomic segments for the time from the performance of the START operation through the performance of the STOP operation, minus the segments corresponding to the period of time where the production of tracking data was interrupted and the period of time during which network backpressure cause the TD service to discard segments.

****

**Figure A‑3: TDM Segment Generation, TD-CSTS Operations, and Tracking Data Message Document for the Real-Time Delivery Mode with User-Specified Start and Stop Generation Times**

****

**Figure A‑4: TDM Segment Generation, TD-CSTS Operations, and Tracking Data Message Document for the Real-Time Delivery Mode with Unspecified Start and Stop Generation Times**

* + 1. Complete Delivery Mode TD-CSTS

Figure A‑5 illustrates some of the data transformations from Tracking Data Production, through the TD-CSTS operations to the resulting Tracking Data Message document for the complete delivery mode case. In the complete delivery mode, both the start-generation-time and the stop-generation-time must be specified by the user in the START invocation.

The first difference between Figure A‑5 and the previous two figures is that the Tracking Data Production functionality that is “seen” by the TD service instance is the Tracking Data Recording Buffer (instead of TDM Segment Generation), which may hold up to several weeks or more worth of tracking data. The start-generation-time parameter of the TD-CSTS START invocation can contain any time before the end of the service instance provision period of the TD-CSTS instance. If the start-generation-time is before the time that the START operation is performed, then once the START return is sent the selected tracking data types and any production-related notifications that are stored in the Recording Buffer for the requested period are transferred by the service.

The second difference is that in complete mode, the service production status is determined by the status of the TDM Recording Buffer. In this example, when the status of the TDM Recording Buffer is temporarily halted, transfer of atomic segments is suspended. However, as soon as the TDM Recording Buffer becomes operational again, the transfer of atomic segments resumes with the segment immediately following the one that was previously sent.

The third, and most important, difference is that unlike the real-time delivery mode, if network backpressure is encountered the service continues to attempt to send all requested tracking data. When a tracking data segment or notification with a timestamp later than the specified stop-generation-time is encountered in the Recording Buffer, the TD service provider transfers an ‘end of data’ notification, which signals the TD-CSTS user that there are no more data in the requested timespan, and therefore the TDM can be closed.

The net result is a TDM with atomic segments for the time from the start-generation-time through the stop-generation-time, minus segments corresponding to the period of time where the production of tracking data was interrupted.

****

**Figure A‑5: TDM Recording Buffer, TD-CSTS Operations, and Tracking Data Message Document for the Complete Delivery Mode**

1. Service Object Identifiers Module

(Normative)

CCSDS-TRACKING-DATA-OBJECT-IDENTIFIERS

{ iso identified-organization (3) standards-producing-organization(112) ccsds(4) css(4) csts(1) services(2) trackingDataService (2) trackingDataServiceModules (4) object-identifiers (1)

}

DEFINITIONS

IMPLICIT TAGS

::= BEGIN

EXPORTS

trackingDataDerivedServices

, trackingDataExtServiceParameters

, trackingDataServiceProcedures

**;**

IMPORTS services

, crossSupportFunctionalities

 FROM CCSDS-CSTS-OBJECT-IDENTIFIERS

 BufferSize

 FROM CCSDS-CSTS-COMMON-TYPES

;

-- The CCSDS-CSTS-OBJECT-IDENTIFIERS and CCSDS-CSTS-COMMON-TYPES modules

-- are defined in E3.1 and E3.3 of the CSTS Framework [1]

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Root Object Identifiers of the Service

trackingDataService OBJECT IDENTIFIER ::= {services 2}

trackingDataServiceDerivedServices OBJECT IDENTIFIER ::= {trackingDataService 1}

trackingDataServiceExtServiceParameters OBJECT IDENTIFIER ::=
{trackingDataService 2}

trackingDataServiceProcedures OBJECT IDENTIFIER ::=
{trackingDataService 3}

trackingDataServiceModules OBJECT IDENTIFIER ::= {trackingDataService 4}

pBTDMDtrackingDataTypes OBJECT IDENTIFIER ∷=
{trackingDataServiceExtServiceParameters 1}

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Procedure Type Identifier:

buffTrkDataMsgDel OBJECT IDENTIFIER ::=
 {trackingDataServiceProcedures 1}

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Root Object Identifiers of the TD-CSTS Provider Functional

-- Resource Type

tdCstsProvider OBJECT IDENTIFIER ::=

 {crossSupportFunctionalities 19}

tdCstsProviderParametersId OBJECT IDENTIFIER ::= {tdCstsProvider 1}

tdCstsProviderEventsId OBJECT IDENTIFIER ::= {tdCstsProvider 2}

tdCstsProviderDirectivesId OBJECT IDENTIFIER ::= {tdCstsProvider 3}

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Framework Procedure Object Identifiers used by the
-- Tracking Data service

-- The Information Query procedures may be used to request the current
-- values of procedure configuration parameters with the OIDs
-- - pACserviceUserRespTimer
-- - pACinitiatorId
-- - pACresponderId
-- - pACresponderPortId
-- - pACserviceInstanceId
-- - pBDDreturnBufferSize
-- - pBDDdeliveryMode
-- - pIQlistNames

-- as defined in the CSTS Framework [1] in the

-- CCSDS-CSTS-FW- PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES

-- module

END

1. PRocedure – Buffered Tracking Data Message Data Delivery PDUs

(Normative)
	1. Tracking Data BUFFERED-TRACKING-DATA-MESSAGE\_DELIVERY\_PDU module

CCSDS-TRACKING-DATA-BUFFERED-TRACKING-DATA-MESSAGE-DELIVERY-PDUS

{ iso identified-organization (3) standards-producing-organization(112) ccsds(4) css(4) csts(1) services (2) trackingDataService (2) trackingDataServiceModules (4) extensions (2) buffTrkDataMsgDelPdus (1)

}

DEFINITIONS

IMPLICIT TAGS

::= BEGIN

IMPORTS BufferedDataDeliveryPdu

 FROM CCSDS-CSTS-BUFFERED-DATA-DELIVERY-PDUS

-- CCSDS-CSTS-BUFFERED-DATA-DELIVERY-PDUS is defined in E3.7 of the CSTS
-- SFW [1]

 trackingDataServiceExtServiceParameters

FROM CCSDS-TRACKING-DATA-OBJECT-IDENTIFIERS

;

-- =============================================================

-- The Buffered Tracking Data Message Delivery procedure is derived from

-- the Buffered Data Delivery procedure. It reuses the PDU defined in the

-- Buffered Data Delivery procedure: BufferedDataDeliveryPdu type defined

-- in the CCSDS-CSTS-BUFFERED-DATA-DELIVERY-PDUS module of the CSTS

-- Framework [1].

-- =============================================================

-- \*\*\*\*\*

-- START Invocation extension parameters

-- No extension parameters added to the START Invocation of the Buffered

-- Data Delivery procedure. Therefore,

-- ‘StartInvocation’: ‘startInvocationExtension’: ‘bddStartInvocExt’:

-- ‘BuffDataDelStartInvocExt’: ‘buffDataDelStartInvocExtExtension’ (see

-- CCSDS-CSTS-BUFFERED-DATA-DELIVERY-PDUS module in E3.7 of [1]) shall be set to

-- ‘notUsed’

-- START positive return parameters

-- START positive return is extended with tdm-header parameter. This

-- extension is defined by ‘StartReturn’: ‘StandardReturnHeader’: ‘result’:

-- ‘positive’: ‘BuffTrkDataDelStartPosReturnExt’. No further parameters are

-- added, i.e., ‘StartReturn’: ‘StandardReturnHeader’: ‘result’:

-- ‘positive’: ‘BuffTrkDataDelStartPosReturnExt’:

-- ‘buffTrkDataDelStartPosReturnExtExtension’ shall be set to ‘notUsed’

BuffTrkDataDelStartPosReturnExt ::= SEQUENCE

{ tdmHeader VisibleString

 buffTrkDataDelStartPosReturnExtExtension Extended

}

buffTrkDataDelStartPosReturnExt OBJECT IDENTIFIER ::=

{trackingDataServiceExtServiceParameters 1}

-- START negative return extension parameters

-- No extension parameters added to the START negative return of the

-- BufferedDataDelivery procedure. Therefore

-- ‘StartReturn’: ‘StandardReturnHeader’: ‘result’: ‘negative’:

-- ‘negExtension’ (see CCSDS-CSTS-COMMON-OPERATIONS-PDUS module in E3.4 – -- [1]) shall be set to ‘notUsed’

-- START negative return extension diagnostics

-- No extension diagnostics added to the START negative return of the

-- BufferedDataDelivery procedure. Therefore the negative StartReturn is

-- returned using one of the common diagnostics

-- of ‘StandardReturnHeader’: ‘result’: ‘negative’: ‘diagnostic’:

-- ‘Diagnostic’ (see CSTS SFW 3.3.2.7) or one of the additional diagnostics

-- defined by ‘StartReturn’: ‘StandardReturnHeader’: ‘result’: ‘negative’:

-- ‘diagnostic’: ‘Diagnostic’: ‘diagnosticExtension’: ‘startDiagnosticExt’:

-- ‘StartDiagnosticExt’ or any of the additional values defined

-- by ‘StartReturn’: ‘StandardReturnHeader’: ‘result’: ‘negative’:

-- ‘diagnostic’: ‘Diagnostic’: ‘diagnosticExtension’: ‘startDiagnosticExt’:

-- ‘StartDiagnosticExt’: ‘startDiagnosticExtExtension’: ‘bddStartDiagExt’:

-- ‘BuffDataDelStartDiagnosticExt’ except

-- buffDataDelStartDiagnosticExtExtension.

-- \*\*\*\*\*

-- STOP Invocation extension parameters

-- No extension parameters added to the STOP Invocation of the

-- BufferedDataDelivery procedure. Therefore,

-- ‘StopInvocation’: ‘stopInvocationExtension’(see

-- CCSDS-CSTS-COMMON-OPERATIONS-PDUS module in E3.4 of [1])shall be set to

-- ‘notUsed’

-- STOP positive return extension parameters

-- No extension parameters added to the STOP positive return of the

-- BufferedDataDelivery procedure. Therefore

-- ‘StopReturn’: ‘StandardReturnHeader’: ‘result’: ‘positive’: (see

-- CCSDS-CSTS-COMMON-OPERATIONS-PDUS module in E3.4 of [1]) shall be set to

-- ‘notUsed’.

-- STOP negative return extension parameters

-- No extension parameters added to the STOP negative return of the.

-- BufferedDataDelivery procedure. Therefore

-- ‘StopReturn’: ‘StandardReturnHeader’: ‘result’: ‘negative’:

-- ’negExtension’ (see CCSDS-CSTS-COMMON-OPERATIONS-PDUS module in E3.4 of [1])

-- shall be set to ‘notUsed’.

-- STOP negative return extension diagnostics

-- The negative StopReturn is returned using one of the common diagnostics

-- of ‘StandardReturnHeader’: ‘result’: ‘negative’: ‘diagnostic’:

-- ‘Diagnostic’ (see CSTS SFW 3.3.2.7). No additional diagnostics are

-- specified, i.e., ‘StopReturn’: ‘StandardReturnHeader’: ‘result’:

-- ‘negative’: ‘diagnostic’: ‘Diagnostic’ must not be set to

-- ‘diagnosticExtension’.

-- \*\*\*\*\*

-- TRANSFER-DATA Invocation extension parameters

-- No extension parameters added to the TRANSFER-DATA Invocation of the

-- BufferedDataDelivery procedure. Therefore,

-- ‘TransferDataInvocation’: transferDataInvocationExtension (see

-- CCSDS-CSTS-COMMON- OPERATIONS-PDUS module in E3.4 of [1])

-- shall be set to ‘notUsed’

-- TRANSFER-DATA Invocation data parameter resolution

-- The data parameter of the TRANSFER-DATA Invocation is resolved as an

-- octet string. Therefore,

-- ‘TransferDataInvocation’: data’: ‘AbstractChoice’

-- (see CCSDS-CSTS-COMMON-OPERATIONS-PDUS module in E3.4 of [1]) shall be set to

-- ‘opaqueString’

-- \*\*\*\*\*

-- NOTIFY Invocation extension

-- No extension parameters are added to the NOTIFY Invocation of the

-- BufferedDataDelivery Therefore,

-- ‘NotifyInvocation’: ‘notifyInvocationExtension’ (see

-- CCSDS-CSTS-COMMON-OPERATIONS-PDUS module in E3.4 of [1]) shall be set to

-– ‘notUsed’

--

END

* 1. Transfer Syntax

The BuffTrkDataDelStartPosReturnExt type specified in this module shall be encoded for transfer using the Basic Encoding Rules specified in [4].

1. Tracking Data Service Procedure Parameters, Events, and Directives

(Normative)

CSDS-TRACKING-DATA- SERVICE-PROCEDURE-PARAMETERS-EVENTS-DIRECTIVES

{ iso identified-organization (3) standards-producing-organization(112) ccsds(4) css(4) csts(1) services (2) serviceIdentifiers(2) trackingDataService (2) trackingDataServiceModules (4) procedureParamEventDirective (3)

}

DEFINITIONS

IMPLICIT TAGS

::= BEGIN

IMPORTS Enumerated

 FROM CCSDS-CSTS-COMMON-TYPES

-- CCSDS-CSTS-COMMON-TYPES is defined in E3.3 of the CSTS
-- Framework [1]

 trackingDataServiceExtServiceParameters

FROM CCSDS-TRACKING-DATA-OBJECT-IDENTIFIERS

;

-- ============================================================
-- BUFFERED TRACKING DATA MESSAGE DELIVERY

pBTDMDtrackingDataTypes OBJECT IDENTIFIER ∷-
 {trackingDataServiceExtServiceParameters 2}

PBTDMDtrackingDataTypes ::= SET OF TrackingDataType

TrackingDataType ::= Enumerated

{ dopplerInstantaneous [0]

, dopplerIntegrated [1]

, range [2]

, carrierPower [3]

, carrierPowerToNoiseSpectralDensity [4]

, rangingPowerToNoiseSpectralDensity1 [5]

, receiveFrequency [6]

, transmitFrequency [7]

, transmitFrequencyRate [8]

, antennaAngles [9]

}

END

1. Tracking Data Production

(Normative)
	1. General

The production of tracking data for transfer via the TD-CSTS is performed by two functions: the TDM Segment Generation function and the TDM Recording Buffer function,

The TDM Segment Generation function generates a time-ordered sequence of Atomic Segments as defined in A3, which correspond to Segments of the CCSDS Tracking Data Message [3] with the additional constraint that each segment may contain only one tracking data measurement (except in the case of antenna angles, in which case the pair of angle is contained in the Atomic Segment).

The TDM Recording Buffer function stores the Atomic Segments generated by the TDM Segment Generation function, for subsequent retrieval by TD-CSTS instances operating in the Complete delivery mode.

* 1. Specification of the TDM Segment Generation Function
		1. TDM Segment Generation Function Behavior

The TDM Segment Generation function shall collect from the radiometric data-generating functional resources of the Service Package the following tracking data measurements (as appropriate to the tracking activities being performed in the context of the Service Package):

* Doppler (instantaneous);
* Doppler (integrated);
* Range;
* Carrier power;
* Carrier power to noise spectral density;
* Ranging power to noise spectral density;
* Receive frequency;
* Transmit frequency;
* Transmit frequency rate; and
* Antenna angles.

See [3] for the definitions of these terms.

Each tracking data measurement shall be accompanied by the time at which it was measured at its respective functional resource.

NOTE - The periodicity at which the tracking data measurements are generated, and the degree to which Service Management may control that periodicity, may vary by functional resource instance and individual implementation and is outside the scope of this Recommended Standard.

The TDM Segment Generation function shall format the tracking measurements into TDM Tracking Data Records.

The TDM Segment Generation function shall combine the Tracking Data Records and their associated metadata into Atomic Segments, as defined in annex A.

Each Atomic Segment shall constitute a Service Production Data Unit (see the Buffered Data Delivery procedure of [1]).

The TDM Segment Generation function shall make its current status available through the tdmSegmentGenStatus parameter, with values ‘configured’, ‘operational’, interrupted’, and ‘halted’.

The TDM Segment Generation function shall emit a tdmSegmentGenStatusChange event for each change of status, with the event-value equal to the value of the tdmSegmentGenStatus parameter after the status change.

* + 1. TDM Segment Generation function Service MANAGEment INFORMATION

For each TDM Segment Generation function instance, Service Management shall establish the relationships among each tracking signal path (including the names of the PARTICIPANTs (see reference [3]) involved with that signal path), the radiometric data-generating functional resources of the Service Package that constitute each signal path, and the TDM Metadata Keywords specified in section 3.3 of reference [3], as necessary to form the TDM Metadata Sections of the Atomic Segments.

* + 1. TDM Segment Generation Functional resource type

The OID for the Functional Resource type that represents the TDM Segment Generation function shall be as specified in the SANA Functional Resource Registry, using the FR classifier tdmSegmentGen.

The tdmSegmentGenStatus parameter shall be registered under the tdmSegmentGenParametersId ({tdmSegmentGen 1} node of the SANA FR registry.

The tdmSegmentGenStatusChange event shall be registered under the tdmSegmentGeneventsId ({tdmSegmentGen 2} node of the SANA FR registry.

Any other monitored parameters, notifiable events, or directives that may be defined for the TDM Segment Generation FR type shall be registered with SANA under the tdmSegmentGenParametersId, tdmSegGenEventsId, or tdmSegGenDirectivesId ({tdmSegmentGen 3})nodes, respectively, of the SANA FR registry.

* 1. Specification of the TDM Recording Buffer Function
		1. TDM Recording Buffer Function Behavior

The functionality of the TDM Recording Buffer shall be as that specified for Recording Buffer in 4.5.7 of the CSTS SFW [1].

The Service Production Data Units recorded by the TDM Recording Buffer function shall be the Atomic Segments generated by the TDM Segment Generation function (E2).

The TDM Segment Generation function shall make its current status available through the tdmRecordingBufferStatus parameter, with values ‘configured’, ‘operational’, interrupted’, and ‘halted’.

The TDM Segment Generation function shall generate a tdmRecordingBufferStatusChange event for each change of status, with the event-value equal to the value of the tdmRecordingBufferStatus parameter after the status change.

The TDM Recording Buffer shall make the size of the recording buffer available through the tdmRecordingBufferSize parameter, which conforms to the specification of the queriable recording-buffer-size parameter defined in CSTS SFW section 4.5.7.10).

In accordance with CSTS SFW section 4.5.7.5 (b), if the recording buffer overflows, the TDM Recording Buffer FR type shall emit a tdmFrRecordingBufferOverflow event which conforms to the specification of the ‘fr recording buffer overflow’ event. The event-value for the tdmFrRecordingBufferOverflow event shall be empty.

* + 1. TDM Recording Buffer function Service MANAGEment INFORMATION

NOTE - The CSTS SFW states in section 4.5.7.5 that the “time span over which data is retained in the recording buffer, the policy for deleting data from the recording buffer, and the conditions under which the recording buffer begins to accept data following an overflow condition are outside the scope of” the CSTS SFW. They are also outside the scope of this Recommended Standard.

* + 1. TDM Recording Buffer functional resource type

The OID for the Functional Resource type that represents the TDM Recording Buffer function shall be as specified in the SANA Functional Resource Registry, using the FR classifier tdmRecordingBuffer.

The tdmRecordingBufferStatus parameter shall be registered under the tdmRecordingBufferParametersId ({tdmRecordingBuffer 1} node of the SANA FR registry.

The tdmRecordingBufferSize parameter shall be registered under the tdmRecordingBufferParametersId node of the SANA FR registry.

The tdmSegmentGenStatusChange event shall be registered under the tdmSegmentGeneventsId ({tdmSegmentGen 2} node of the SANA FR registry.

The tdmFrRecordingBufferOverflow event shall be registered under the tdmSegmentGeneventsId node of the SANA FR registry.

Any other monitored parameters, notifiable events or directives that may be defined for the TDM Recording Buffer FR type shall be registered under the tdmRecordingBufferParametersId, tdmRecordingBufferEventsId, or tdmRecordingBufferDirectivesId ({tdmRecordingBuffer 3})nodes, respectively, of the SANA FR registry.

1. Implementation Conformance Statement Proforma

(Normative)
	1. INTRODUCTION
		1. OVERVIEW

This annex provides the Implementation Conformance Statement (ICS) Requirements List (RL) for an implementation of the *Cross Support Transfer Services – Tracking Data Service*, CCSDS 922.2-W-0.14, June 2016. CCSDS 922.2 specifies the requirements on the provider of the Tracking Data Cross Support Transfer Service.

The ICS for an implementation is generated by completing the RL in accordance with the instructions below. An implemenation shall satisfy the mandatory conformance requirements reference in the RL.

The RL support column in this annex is blank. An implementation’s completed RL is called the PICS. The PICS states which capabilities and options have been implemented. The following can use the PICS:

1. the implementer, as a checklist to reduce the risk of failure to conform to the standard through oversight;
2. a supplier or potential acquirer of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma;
3. a user or potential user of the implementation, as a basis for initially checking the possibility of interworking with another implementation (it should be noted that, while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICSes);
4. a tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.
	* 1. ABBREVIATIONS AND CONVENTIONS

The RL consists of information in tabular form. The status of features is indicated using the abbreviations and conventions described below.

Item Column

The item column contains a prefix identifying the element the given table is referring to and a sequential numbers for items in the table.

Feature Column

The feature column contains a brief descriptive name for a feature. It implicitly means ‘Is this feature supported by the implementation?’

Status Column

The status column uses the following notations:

|  |  |  |
| --- | --- | --- |
|  | M | mandatory; |
|  | O | optional; |
|  | O.<n> | optional, but support of at least one of the group of options labeled by the same numeral <n> is required; |
|  | C<n> | conditional as defined in corresponding expression below table; |
|  | X | prohibited; |
| 1.
 | N/A | not applicable. |

Support Column Symbols

The support column is to be used by the implementer to state whether a feature is supported by entering Y, N, or N/A, indicating:

1. Y Yes, supported by the implementation;
2. N No, not supported by the implementation;
3. N/A Not applicable.

The support column should also be used, when appropriate, to enter values supported for a given capability.

Allowed Values Column

All PDU parameter types are specified in annex E of the CSTS SFW (reference [[1]].using ASN.1. The ASN.1 data type specifications constrain among others the permissible value range and therefore such constraints are not repeated in the Allowed Values column in the tables contained in this ICS annex. However, if a parameter is constrained for all instances of the given PDU to a subset of the range or set specified for that parameter in annex E, then the subset is identified in the tables that contain PDU parameters.

Allowed Values Column Symbols

If the allowed values are too large to fit in the Allowed Values cell, the Allowed Values column uses the notation “AV<n>” not indication that the allowed values are specified below the table.

Supported Values Column

The Supported Values column is to be used by the implementer to state whether the specified range or set of values for the parameter is supported by entering Y or SV<n>, indicating:

|  |  |  |
| --- | --- | --- |
|  | Y | Yes, the range/set defined in the Recommended Specification is fully supported by the implementation; |
|  | SV<n> | The range/set defined in the Recommended Specification is not fully supported by the implementation. The supported subset is documented below the table. |

* + 1. INSTRUCTIONS FOR COMPLETING THE RL

An implementer shows the extent of compliance to the Recommended Standard by completing the RL; that is, the state of compliance with all mandatory requirements and the options supported are shown. The resulting completed RL is called a PICS. The implementer shall complete the RL by entering appropriate responses in the support or values supported column, using the notation described in F1.2. If a conditional requirement is inapplicable, N/A should be used. If a mandatory requirement is not satisfied, exception information must be supplied by entering a reference X*i*, where *I* is a unique identifier, to an accompanying rationale for the noncompliance.

* 1. PICS PROFORMA FOR the Tracking Data CSTS PROTOCOL (CCSDS 922.2-B-1)
		1. GENERAL INFORMATION

The PICS for a TD-CSTS implementation shall encompass the filled in Table F‑1 to Table F‑4.

Table F‑1 : Identification of PICS

|  |  |
| --- | --- |
| Date of Statement (DD/MM/YYYY) |  |
| PICS serial number |  |
| System Conformance statement cross-reference |  |

Table F‑2 : Identification of Implementation Under Test

|  |  |
| --- | --- |
| Implementation name |  |
| Implementation version |  |
| Special Configuration |  |
| Other Information |  |

Table F‑3 : Identification of Supplier

|  |  |
| --- | --- |
| Supplier |  |
| Contact Point for Queries |  |
| Implementation Name(s) and Versions |  |
| Other information necessary for full identification, e.g., name(s) and version(s) for machines and/or operating systems;System Name(s) |  |

Table F‑4 : Identification of Specification

|  |
| --- |
| CCSDS 922.2-B-1  |
| Have any exceptions been required?NOTE – A YES answer means that the implementation does not conform to the Recommended Standard. Non-supported mandatory capabilities are to be identified in the PICS, with an explanation of why the implementation is nonconforming. | Yes [ ] No [ ] |

* + 1. REQUIREMENTS LIST

This section provides the Requirement Lists for the elements specified in this Recommended Standard.

Table F‑5 : Required Procedures

| Procedures |
| --- |
| Item | Description | Reference | Status | Support |
|  | Association Control | CSTS SFW 4.3 | M |  |
|  | Buffered Data Delivery | CSTS SFW 4.5 | M |  |
| 1.
 | Buffered Tracking Data Message Delivery | TD-CSTS 4 | M |  |
| 1.
 | Information Query | CSTS SFW 4.9 | M |  |

The Buffered Data Delivery procedure is mandatory in the sense that the Buffered Tracking Data Message Delivery procedure (which is mandatory) is derived from the Buffered Data Delivery procedure. In this TD-CSTS ICS, all requirements for the Buffered Data Delivery procedure are covered by the requirements for the Buffered Data Delivery procedure.

Table F‑6 : Required PDUs

| Item | PDU | Ref. | Service-Provider-System | Service-User-System |
| --- | --- | --- | --- | --- |
|  | Status | Support | Status | Support |
|  | BindInvocation | CSTS SFW E3.5 | M |  | M |  |
|  | BindReturn | CSTS SFW E3.5 | M |  | M |  |
|  | PeerAbortInvocation | CSTS SFW E3.5 | M |  | M |  |
|  | UnbindInvocation | CSTS SFW E3.5 | M |  | M |  |
|  | UnbindReturn | CSTS SFW E3.5 | M |  | M |  |
|  | GetInvocation | CSTS SFW E3.4 | M |  | M |  |
|  | GetReturn | CSTS SFW E3.4 | M |  | M |  |
|  | NotifyInvocation | CSTS SFW E3.4 | M |  | M |  |
|  | StartInvocation | CSTS SFW E3.4 | M |  | M |  |
|  | StartReturn | CSTS SFW E3.4 | M |  | M |  |
|  | StopInvocation | CSTS SFW E3.4 | M |  | M |  |
|  | StopReturn | CSTS SFW E3.4 | M |  | M |  |
|  | TransferData-Invocation | CSTS SFW E3.4 | M |  | M |  |

Table F‑7 : BIND Invocation Parameters

| Parameters of the BindInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | invokerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | procedureInstanceId | CSTS SFW E3.3 | M |  | AV1 |  |
|  | initiatorIdentifier | CSTS SFW E3.5 | M |  |  |  |
|  | responder-PortIdentifier | CSTS SFW E3.5 | M |  |  |  |
|  | serviceType | CSTS SFW E3.5 | M |  |  |  |
|  | versionNumber | CSTS SFW E3.5 | M |  |  |  |
|  | serviceInstance-Identifier | CSTS SFW E3.5 | M |  |  |  |
|  | bindInvocation-Extensior | CSTS SFW E3.5 | M |  | ‘not-Used’ |  |

AV1 For the BIND invocation the procedureRole element of the parameter bindInv-3 must be set to ‘associationControl’.

The parameters bindInv-1, bindInv-2 and bindInv-3 are contained in the complex parameter

invocationHeader shown in CSTS SFW E3.5. This parameter is of the type StandardInvocationHeader that is specified in CSTS SFW E3.3.

Table F‑8 : BIND Return Parameters

| Parameters of the BindReturn PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | performerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | result | CSTS SFW E3.3 | M |  |  |  |
|  | positive | CSTS SFW E3.3 | C1 |  | ‘not-Used’ |  |
|  | diagnostics | CSTS SFW E3.3 | C2 |  | AV2 |  |
|  | negExtension | CSTS SFW E3.3 | C2 |  | ‘not-Used’ |  |
|  | responderIdentifier | CSTS SFW E3.5 | M |  |  |  |

C1 IF bindRet-3 = ‘positive’ THEN M ELSE X

C2 IF bindRet-3 = ‘negative’ THEN M ELSE X

AV2 For the negative BIND return the parameter bindRet-5 is extended by the type AssocBindDiagnosticExt defined in CSTS SFW E3.5. Therefore the parameter bindRet-5 may have any value (a) defined for the Diagnostic type in CSTS SFW E3.3 except ‘diagnosticExtension’; or (b) any value defined by ‘diagnosticExtension’: ‘acBindDiagExt’: ‘AssocBindDiagnosticExt’ defined in CSTS SFW E3.5 except ‘assocBindDiagnosticExtExtension’.

All parameters of the BIND return PDU except bindRet-7 are contained the complex parameter of the type StandardReturnHeader that is specified in CSTS SFW E3.3. Specific extensions are, however, specified in CSTS SFW E3.5.

‑9 : PEER-ABORT Invocation Parameters

| Parameters of the PeerAbortInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | diagnostic | CSTS SFW E3.5 | M |  | 40..126 |  |

Table ‑10 : UNBIND Invocation Parameters

| Parameters of the UnbindInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | invokerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | procedureInstanceId | CSTS SFW E3.3 | M |  | AV3 |  |
|  | unbindInvocation-Extension | CSTS SFW E3.5 | M |  | ‘not-Used’ |  |

AV3 For the UNBIND invocation the procedureRole element of the parameter unbindInv-3 must be set to ‘associationControl’.

The parameters unbindInv-1, unbindInv-2 and unbindInv-3 are contained in the complex parameter invocationHeader shown in CSTS SFW E3.5. This parameter is of the type StandardInvocationHeader that is specified in CSTS SFW E3.3.

Table F‑11 : UNBIND Return Parameters

| Parameters of the UnbindReturn PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | performerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | result | CSTS SFW E3.3 | M |  | AV4 |  |

AV4 The value of the result parameter of the UNBIND return PDU shaIways be set to the value ‘positive’: ‘notUsed’; i.e., the reIis always positive and not extended.

All parameters of the UNBIND return PDU are contained the complex parameter of the type StandardReturnHeader that is specified in CSTS SFW E3.3.

Table F‑12 : GET Invocation Parameters

| Parameters of the GetInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | invokerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | procedureInstanceId | CSTS SFW E3.3 | M |  | AV4 |  |
|  | listOfParameters | CSTS SFW E3.4 | M |  |  |  |
|  | getInvocation-Extension | CSTS SFW E3.4 | M |  | ‘not-Used’ |  |

AV4 The value of the procedureRole element of the parameter getInv-3 must be set to ‘secondary procedure’.

The parameters getInv-1, getInv-2, and getInv-3 are contained in the complex parameter standardInvocationHeader shown in CSTS SFW E3.4. This parameter is of the type StandardInvocationHeader that is specified in CSTS SFW E3.3.

Table F‑13 : GET Return Parameters

| Parameters of the GetReturn PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | performerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | result | CSTS SFW E3.3 | M |  |  |  |
|  | positive | CSTS SFW E3.3 | C3 |  | AV5 |  |
|  | qualifiedParameters | CSTS SFW E3.4 | C3 |  | AV6 |  |
|  | getPosReturnExt-Extension | CSTS SFW E3.4 | C3 |  | ‘not-Used’ |  |
|  | diagnostics | CSTS SFW E3.3 | C4 |  | AV7 |  |
|  | negExtension | CSTS SFW E3.3 | C4 |  | ‘not-Used’ |  |

C3 IF getRet-3 = ‘positive’ THEN M ELSE X

C4 IF getRet-3 = ‘negative’ THEN M ELSE X

AV5 For the positive GET return the parametergetRet-4 is set to ‘getPosReturnExt’: ‘GetPosReturnExt’ defined inCSTS SFW E3.4.

AV6 For the positive GET return the parametergetRet-5 is specified by ‘qualifiedparameters’: ‘QualifiedParametersSequence’. The type QualifiedParametersSequence is defined inCSTS SFW E3.4.

AV7 For the negative GET return the parameter getRet-7 is extended by the type GetDiagnosticExt defined in CSTS SFW E3.4. Therefore the parameter getRet-7 may have (a) any standard value defined for the Diagnostic type in CSTS SFW E3.3 except ‘diagnosticExtension’; or (b) any value defined by the extension ‘diagnosticExtension’: ‘getDiagnosticExt’: ‘GetDiagnosticExt’ defined in CSTS SFW E3.4 except ‘getDiagnosticExtExtension’.

All parameters of the GET return PDU are contained the complex parameter of the type StandardReturnHeader that is specified in CSTS SFW E3.3. Specific extensions are, however, specified in CSTS SFW E3.4.

.**Table** **F‑14** **: START Invocation Parameters**

| Parameters of the StartInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | invokerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | procedureInstanceId | CSTS SFW E3.3 | M |  | AV8 |  |
|  | startInvocation-Extension | CSTS SFW E3.4 | M |  | AV9 |  |
|  | startGenerationTime | CSTS SFW E3.7 | M |  |  |  |
|  | stopGenerationTime | CSTS SFW E3.7 | M |  |  |  |
|  | buffDataDelStartInvocExtExtension | CSTS SFW E3.7  | M |  | ‘not used’ |  |

AV8 The value of the procedureRole element of the parameter startInv-3 must be set to ‘prime procedure’.

AV9 The startInv-4 parameter shall be set to the value ‘bddStartInvocExt’: 'BuffDataDelStartInvocExt'.

The parameters startInv-1, startInv-2 and startInv-3 are contained in the complex parameter standardInvocationHeader shown in CSTS SFW E3.4. This parameter is of the type StandardInvocationHeader that is specified in CSTS SFW E3.3.

Table F‑15 : START Return Parameters

| Parameters of the StartReturn PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | performerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | result | CSTS SFW E3.3 | M |  |  |  |
|  | positive | CSTS SFW E3.3 | C5 |  | AV10 |  |
|  | diagnostics | CSTS SFW E3.3  | C6 |  | AV11 |  |
|  | negExtension | CSTS SFW E3.3  | C6 |  | ‘not used’ |  |
|  | buffTrkDataDelStart-PosReturnExt-Extension | ANNEX C | C5 |  | ‘not used’ |  |

C5 IF startRet-3 = ‘positive’ THEN M ELSE X

C6 IF startRet-3 = ‘negative’ THEN M ELSE X

AV10 The parameter startRet-4 has the value ‘buffTrkDataDelStartPosReturnExt’: 'BuffTrkDataDelStartPosReturnExt', as defined in ANNEX C

AV11 For the START return PDU the parameter startRet-5 is extended by the type StartDiagnosticExt defined in CSTS SFW E3.4 and BuffDataDelStartDiagnosticExt defined in CSTS E3.7. Therefore the parameter startRet-5 may have (a) any standard value defined for the Diagnostic type in CSTS SFW E3.3 except ‘diagnosticExtension’; (b) any value defined by the extension ‘diagnosticExtension’: ‘startDiagnosticExt’: ‘StartDiagnosticExt’ in CSTS SFW E3.4 except ‘startDiagnosticExtExtension’; or (c) any value defined by the extension ‘diagnosticExtension’: ‘startDiagnosticExt’: ‘StartDiagnosticExt’: ‘startDiagnosticExtExtension’: bddStartDiagExt’: ‘BuffDataDelStartDiagnosticExt’ in CST SFW E3.7 except ‘buffDataDelStartDiagnosticExtExtension’.

Table F‑16 : STOP Invocation Parameters

| Parameters of the StopInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | invokerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | procedureInstanceId | CSTS SFW E3.3 | M |  | AV12 |  |
|  | stopInvocation-Extension | CSTS SFW E3.4 | M |  | ‘not used’ |  |

AV12 The value of the procedureRole element of the parameter stopInv-3 must be set to ‘prime procedure’.

The parameters stopInv-1, stopInv-2 and stopInv-3 are contained in the complex parameter standardInvocationHeader shown in CSTS SFW E3.4. This parameter is of the type StandardInvocationHeader that is specified in CSTS SFW E3.3.

Table F‑17 : STOP Return Parameters

| Parameters of the StopReturn PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | performerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | result | CSTS SFW E3.3 | M |  |  |  |
|  | positive | CSTS SFW E3.3 | C7 |  | ‘not used’ |  |
|  | diagnostics | CSTS SFW E3.3 | C8 |  | AV13 |  |
|  | negExtension | CSTS SFW E3.3 | M |  | ‘not used’ |  |

C7 IF stopRet-3 = ‘positive’ THEN M ELSE X

C8 IF stopRet-3 = ‘negative’ THEN M ELSE X

AV13 The parameter stopRet-5 may have any standard value defined for the Diagnostic type in CSTS SFW E3.3 except ‘diagnosticExtension’.

All parameters of the STOP return PDU are contained the complex parameter of the type StandardReturnHeader that is specified in CSTS SFW E3.3.

Table F‑18 : TRANSFER-DATA Invocation Parameters

| Parameters of the TransferDataInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
|  | invokerCredentials | CSTS SFW E3.3 | M |  |  |  |
|  | invokeId | CSTS SFW E3.3 | M |  |  |  |
|  | procedureInstanceId | CSTS SFW E3.3 | M |  | AV14 |  |
|  | generationTime | CSTS SFW E3.4 | M |  |  |  |
|  | sequenceCounter | CSTS SFW E3.4 | M |  |  |  |
|  | data | CSTS SFW E3.4 | M |  | A15 |  |
|  | transferData-InvocationExtension | CSTS SFW E3.4 | M |  | ‘not used’ |  |

AV14 The value of the procedureRole element of the parameter transferDataInv-3 must be set to ‘prime procedure’.

AV15 The value of transferDataInv-6 is refined to be an octet string formatted as an Atomic Segment as defined in A3.

The parameters transferDataInv-1, transferDataInv-2 and transferDataInv-3 are contained in the complex parameter standardInvocationHeader shown in CSTS SFW E3.4. This parameter is of the type StandardInvocationHeader that is specified in CSTS SFW E3.3.

Table F‑19 : NOTIFY Invocation Parameters

| Parameters of the NotifyInvocation PDU |
| --- |
| Item | Parameter | Ref. | Status | Support | Values |
| Allowed | Supported |
| 1.
 | invokerCredentials | CSTS SFW E3.3 | M |  |  |  |
| 1.
 | invokeId | CSTS SFW E3.3 | M |  |  |  |
| 1.
 | procedureInstanceId | CSTS SFW E3.3 | M |  | AV16 |  |
| 1.
 | eventTime | CSTS SFW E3.4 | M |  |  |  |
| 1.
 | eventName | CSTS SFW E3.4 | M |  |  |  |
| 1.
 | eventValue | CSTS SFW E3.4 | M |  | AV17 |  |
| 1.
 | notifyInvocationExtension | CSTS SFW E3.4 | M |  | ‘not-Used’ |  |

AV16 The value of the procedureRole element of the parameter notifyInv-3 must be set to ‘prime procedure’.

AV17 The value of the notifyInv-6 parameter can be any value that can be expressed using the type SequenceOfQualifiedValues defined in CSTS SFW E3.3 or ‘empty’. The value of ‘eventValue’ must not be set to ‘eventValueExtension’.

The parameters notifyInv-1, notifyInv-2, and notifyInv-3 are contained in the complex parameter standardInvocationHeader in the NotifyInvocation type shown in CSTS SFW E3.4. This parameter is of the type StandardInvocationHeader that is specified in CSTS SFW E3.3.

1. Security, SANA, and Patent Considerations

(Informative)
	1. Security Considerations
		1. Introduction

This subsection describes security aspects of the Tracking Data service.

The CSTS Specification Framework (reference [[1]]) explicitly provides authentication and access control for CSTSes. As one of the suite of CSTSes, the Tracking Data service inherits the authentication and access control capabilities defined in the CSTS Specification Framework. The Tracking Data service provides no service-specific security capabilities. As specified in the CSTS Specification Framework, additional security capabilities, if required, are levied on the underlying communications services that support the TD-CSTS. Specification of the various underlying communications technologies, and in particular their associated security provisions, are outside the scope of this Recommended Standard.

* + 1. Security Concerns with Respect to the Tracking Data Service

The Statements of Security Concerns subsection of the CSTS Specification Framework identifies the support for capabilities that respond to security concerns in the areas of data privacy (also known as confidentiality), data integrity, authentication, access control, availability of resources, and auditing.

* + 1. Potential Threats and Attack Scenarios

As a member of the suite of CSTSes, the Tracking Data service depends on unspecified mechanisms operating in the underlying communications service, or on privacy-ensuring capabilities in the service-specific application processes that interoperate through the Framework procedures, to ensure data privacy (confidentiality). If no such mechanisms are actually implemented, or the mechanisms selected are inadequate or inappropriate to the network environment in which the mission is operating, an attacker could read the data contained in the TD-CSTS protocol data units as they traverse the WAN between service user and service provider.

The CSTS Specification Framework constrains the ability of a third party to seize control of an active CSTS instance, but it does not specify mechanisms that would prevent an attacker from intercepting the protocol data units. Interception of tracking data could assist an attacker in establishing the orbit/trajectory of the Mission spacecraft, which could assist the attacker in subsequent attempts to acquire return link data or jam the forward link. The prevention of such interception attacks depends on unspecified mechanisms in the underlying communications service. If no such mechanisms are actually implemented, or the mechanisms selected are inadequate or inappropriate to the network environment in which the mission is operating, an attacker could intercept data transferred between the service user and the service provider without detection.

If the CSTS authentication capability is not used and if authentication is not ensured by the underlying communications service, attackers could somehow obtain valid initiator-identifier values and use them to initiate TD-CSTS instances by which they could gain access to the tracking data transferred via the service.

The TD-CSTS depends on unspecified mechanisms operating in the underlying communications service to ensure that the supporting network has sufficient resources to provide sufficient support to legitimate service users. If no such mechanisms are actually implemented, or the mechanisms selected are inadequate or inappropriate to the network environment in which the mission is operating, an attacker could prevent legitimate service users from using the TD-CSTS.

If the service provider of the TD-CSTS provides no security auditing capabilities, or if a service user chooses not to employ auditing capabilities that do exist, then attackers may delay or escape detection while stealing data exchanged via the service.

* + 1. Consequences of not Applying Security to the Technology

The consequences of not applying security to the TD-CSTS are possible degradation and loss of ability to use the service, or the interception of tracking data that could aid in the determination of the orbit/trajectory of the spacecraft, acquisition of the space link, and/or jamming of the space link. Note that the ability to actually acquire the space link (that is, either transmit to the spacecraft or receive and interpret data from the spacecraft) depends on the security supplied by the protocols used on that space link.

* 1. SANA Considerations

The TD-CSTS relies on SANA registries to register the object identifiers (OIDs) for the TD-CSTS

* 1. Patent Considerations

There are no patents that are known to apply to the technology used in the Tracking Data service.

1. Acronyms

(INFormative)

|  |  |
| --- | --- |
|  |  |
| CCSDS | Consultative Committee for Space Data Systems |
| CLTU | Communication Link Transmission Unit |
| CM | Complex Management |
| CSTS | Cross Support Transfer Service |
| CSTS SFW | CSTS Specification Framework  |
| DOR | Differential One-way Ranging |
| F-CLTU | Forward CLTU |
| ISO | International Organization for Standardization |
| MDOS | Mission Data Operation System |
| PDU | Protocol Data Unit |
| RAF | Return All Frames |
| RF | Radio Frequency |
| SANA | Space Assigned Numbers Authority |
| SCCS-SM | Space Communication Cross Support Service Management |
| SM | Service Management |
| SLE | Space Link Extension |
| TD-CSTS | Tracking Data Cross Support Transfer Service |
| TDM | Tracking Data Message |
| UM | Utilization Management |
| VLBI | Very Long Baseline Interferometry |

1. Informative references

(INformative)

[I1] *Cross Support Concept—Part 1: Space Link Extension Services*. Report Concerning Space Data Systems Standards, CCSDS 910.3-G-2. Green Book. Issue 2. Washington, D.C.: CCSDS, July 2002.

[I2] *Space Link Extension – Internet Protocol for Transfer Services.* Recommended Standard, CCSDS 913.1-B-1. Blue Book. September 2008.

[I3] *Cross Support Transfer Services Specification Framework Concepts.* Report Concerning Space Data Systems Standards, CCSDS 920.0-G-0.14. Draft Green Book. Issue 0.14. Washington, D.C.: CCSDS, to be published,

[I4] *Space Link Extension – Return All Frames Service Specification*. Recommended Standard, CCSDS 911.1-B-2. Blue Book. November 2004.

[I5] *Space Link Extension – Forward CLTU Service Specification*. Recommended Standard, CCSDS 912.1-B-2. Blue Book. November 2004.

[I6] *Cross Support Transfer Services – Monitored Data Service*. Draft Recommended Standard, CCSDS 922.1-W-0.17. Red Book. June 2016.

[I7] *Extensible Space Communication Cross Support Service Management—Concept*. Report Concerning for Space Data System Standards, CCSDS 902.0-G-1. Green Book. Issue 1. Washington, D.C.: CCSDS, April 2014.

[I8] *Space Communication Cross Support Service Management—Service Management Utilization Request Formats*. Recommendation for Space Data System Standards, CCSDS 902.9-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS. (Forthcoming).

[I9] *Space Communication Cross Support Service Management—Simple Schedule Format Specification*. Recommendation for Space Data System Standards, CCSDS 902.1-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS. (Forthcoming).

1. Cross References to Cross Support Transfer Service Specification Framework

(Informative)

Table J‑1 lists the specific sections and paragraphs of the Cross Support Transfer Service Specification Framework (reference [1]) that are referenced by this Recommended Standard, and identifies the sections and paragraphs of this Recommended Standard that make specific reference to each of those CSTS SFW sections/paragraphs.

Table ‑ : Cross Reference to CSTS SFW Sections and Paragraphs

| **CSTS SFW Section/ Paragraph** | **Referencing Sections/Paragraphs of MD-CSTS** |
| --- | --- |
|  |  |
|  |  |
| 3.3.2.7 | C1 (2 occurrences) |
| 3.11.2.2.3.2 (a) | 6.1 (b) |
| 3.11.2.2.3.2 (b) | 6.1 (c) |
| 4.3 | 3.2.2, Table 3‑1, Table F‑5 |
| 4.3.5, Table 4-2 | 5.2.1, 5.2.2, 5.2.3, 5.2.4, 5.2.5 |
| 4.5 | 2.2.2, Table 3‑1, Table F‑5 |
|  |  |
| 4.5.2.2.2.1 | Table 4‑3 |
|  |  |
|  |  |
|  |  |
| 4.5.3.2.6 | Table 4‑3 |
| 4.5.3.2.7.2 | Table 4‑3 |
| 4.5.3.2.7.3 | Table 4‑3 |
| 4.5.5, Table 4-16 | 5.3.1, 5.3.2, 5.3.3  |
|  |  |
| 4.5.7 | 2.2.1, E3.1 |
| 4.5.7.5 | E3.1, A1.1.1 |
| 4.5.7.9 | A1.1.1 |
| 4.9 | 3.2.8, Table 3‑1, Table F‑5 |
| 4.9.5, Table 4-52 | 5.4.1, 5.4.2 |
|  |  |
| E3.1 | 5.4.1, ANNEX B  |
| E3.3 | ANNEX B, ANNEX C, Table F‑7, Table F‑8, Table F‑10, Table F‑11, Table F‑12, Table F‑13, .Table F‑14, Table F‑15, Table F‑16, Table F‑17, Table F‑18, Table F‑19 |
| E3.4 | C1 (8 occurrences), Table F‑6 (multiple occurrences), Table F‑12, Table F‑13, .Table F‑14, Table F‑15, Table F‑16, Table F‑18, Table F‑19 |
| E3.5 | Table F‑6 (multiple occurrences), Table F‑7, Table F‑8, F‑9, Table F‑10 |
|  |  |
| E3.7 | C1, .Table F‑14, Table F‑15  |
| E3.16 | Table 4‑3, 5.4.1 |
| E3.17 | 6.1, 6.2.1, 6.3.1, 6.4.17.2.1, 7.2.2, 7.2.3, 7.2.4 |
| F3 | 3.3 |

1. bookmarks

(This annex is an artifact of constructing this Recommended Standard and will be removed prior to publication)

[1] Ref\_CSTS\_SFW

[2] Ref\_CSRM

[3] Ref\_TDM

[4] Ref\_IOS\_BER

[I1] IRef\_CrossSupportConcept

[I2] Ref\_SLE\_ISP1

[I3] Ref\_CSTS\_SFW\_Concepts

[I4] Ref\_SLE\_RAF

[I5] Ref\_SLE\_FCLTU

[I6] Ref\_MD\_CSTS

[I7] Ref\_ESCCSSM\_Concept

[I8] Ref\_SCCSSM\_SvcReqAndSvcPkg

[I9] Ref\_SCCSSM\_SvcAgrAndConfigProfile